ERNW RESEARCH
pursuing knowledge.

BMBF UNCOVER

KURZBERICHT ZUM TEILVORHABEN

Version:	1.0
Datum:	20.06.2024
Klassifikation:	Intern
Verbundprojekt:	Datenbasierte EntwurfsUNterstützung durch kontinuierliches MOnitoring Von SichERheitsvorfällen (UNCOVER)
Verbundpartner:	ERNW Research GmbH
Förderkennzeichen:	16KIS1410K
Projektlaufzeit:	01.06.2021–31.12.2023

uncover

ERNW RESEARCH
pursuing knowledge.

Inhaltsverzeichnis

1 Informationen zur Handhabung 4

1.1 Dokumenten-Status und -Besitzer 4

1.2 Erläuterung der Klassifizierung 4

1.3 Versionshistorie 5

2 Kurzbericht zum Teilvorhaben 6

2.1 Ursprüngliche Aufgabenstellung 6

2.2 Wissenschaftlicher und technischer Stand, an den angeknüpft wurde 6

2.3 Ablauf des Vorhabens 6

2.4 Wesentliche Ergebnisse 7

2.5 Zusammenarbeit mit anderen Forschungseinrichtungen 7

ERNW RESEARCH
pursuing knowledge.

Tabellenverzeichnis

Tabelle 1: Dokumenteneigenschaften 4
Tabelle 2: Dokumentenklassifizierungen 4
Tabelle 3: Versionshistorie 5

1 Informationen zur Handhabung

Das vorliegende Dokument ist als *Intern* klassifiziert. Jeglicher Weitergabe oder Offenlegung des Dokuments MUSS die ausdrückliche Freigabe durch den in Abschnitt *Dokumenten-Status und -Besitzer* benannten Dokumentenbesitzer vorausgehen.

1.1 Dokumenten-Status und -Besitzer

Der Besitzer ist als Inhaber des vorliegenden Dokuments ausschließlich entscheidungsbefugt über die Weitergabe dieses Dokuments und verantwortlich für die Distribution der jeweils gültigen Fassung.

Mögliche Einträge für den Status des Dokumentes sind *Vor-Entwurf*, *Entwurf*, *aktuell gültig* und *veraltet*.

Dokumenteneigenschaften	
Titel:	BMBF UNCOVER - Kurzbericht zum Teilvorhaben
Dokumentenbesitzer:	ERNW Research GmbH
Version:	1.0
Status:	Effective
Klassifikation:	Intern
Autor(en):	Andreas Dewald, Matthias Hamann, Patrick Bidinger

Tabelle 1: Dokumenteneigenschaften

1.2 Erläuterung der Klassifizierung

Klassifizierung	Empfänger
Öffentlich:	Jeder
Intern:	Alle Mitarbeiter und Kunden
Vertraulich:	Nur Mitarbeiter
Geheim:	Nur dedizierte Mitarbeiter

Tabelle 2: Dokumentenklassifizierungen

1.3 Versionshistorie

Version	Datum	Details
1.0	20.06.2024	Initiale Version.

Tabelle 3: Versionshistorie

ERNW RESEARCH
pursuing knowledge.

2 Kurzbericht zum Teilvorhaben

Das Ziel des Teilvorhabens der ERNW Research GmbH im BMBF-Projekt UNCOVER war die Beratung zu und Untersuchung von IT-Sicherheit hinsichtlich Konzeption, Design und Umsetzung von Entwicklungsmethoden und Werkzeugunterstützung zur Analyse von Security-Vorfällen im Automotive-Umfeld.

2.1 Ursprüngliche Aufgabenstellung

Für die Realisierung und Einführung autonom agierender Fahrzeuge ist das sichere Zusammenspiel von Funktionen, Systemen und Diensten sowie deren Überwachung über den kompletten Produktlebenszyklus unerlässlich. Eine ausschließliche entwicklungsbegleitende Unterstützung im Rahmen eines Security-by-Design ist nicht mehr ausreichend und muss durch die Rückführung von Erfahrungen aus dem Betrieb kontinuierlich unterstützt werden.

Ein essentieller Baustein des Gesamtvorhabens war die Entwicklung einer flexiblen Monitoring-Plattform zur Erkennung sicherheitsrelevanter Ereignisse im laufenden Betrieb autonom agierender Fahrzeuge. Beitrag der ERNW Research GmbH war hierbei einerseits, bzgl. der Erfassung der Daten zu beraten und die Erfahrungen aus praktischen Sicherheitsuntersuchungen einfließen zulassen, und andererseits, durch Beratung beim Design und Durchführung eines Assessments die Sicherheit der Plattform selbst zu gewährleisten.

2.2 Wissenschaftlicher und technischer Stand, an den angeknüpft wurde

Während des Projekts wurden insbesondere die folgenden Standards berücksichtigt:

- ISO/SAE 21434 "Road vehicles – Cybersecurity engineering"
- ISO 21448 "Road vehicles – Safety of the intended functionality" (SOTIF)

Die im Zuge der Absicherung der Monitoring-Plattform entwickelte Stromchiffre DRACO fußt auf dem in [HKMZ18] vorgeschlagenen Design-Ansatz, die Verwundbarkeit von Stromchiffren wie Sprout [AM15] und Plantlet [MAM16] gegen generische Unterscheidungsangriffe dadurch zu beheben, dass neben dem geheimen symmetrischen Schlüssel auch der öffentliche Initialisierungsvektor (IV) kontinuierlich in das Zustandsupdate mit einfließt.

2.3 Ablauf des Vorhabens

Das Gesamtprojekt war in die folgenden Arbeitspakete unterteilt:

- AP 1: Use Case Definition und Anforderungen an die Daten und Sicherheitsanalysen
- AP 2: Methoden und Werkzeug für kontinuierliche Security- und Safety-Analysen basierend auf den Erkenntnissen aus Security- und Safety-Vorfällen

- AP 3: Monitoring-Plattform zur Erkennung von Security- und Safety-Vorfällen
- AP 4: Methoden für datengetriebene Unterstützung der Entwurfsphase
- AP 5: Datenschutzrechtliche Betrachtung
- AP 6: Evaluation und Demonstration

Die ERNW Research GmbH hat zu den Arbeitspaketen AP 1, AP 3, AP 4 und AP 6 beigetragen. Das Arbeitspaket AP 3 wurde darüber hinaus von der ERNW Research GmbH geleitet. An den Arbeitspaketen AP 2 und AP 5 hatte die ERNW Research GmbH keine direkte Beteiligung.

Wegen Verzögerungen auf Verbundebene, unter anderem bedingt durch die Ausläufer der COVID-19-Pandemie, kam es zu einer Projektverzögerung von einem Monat. Das Projekt wurde entsprechend kostenneutral bis zum 31. Dezember 2023 verlängert.

2.4 Wesentliche Ergebnisse

Zentrale Punkte der Beratungs- und Unterstützungstätigkeit durch die ERNW Research GmbH waren

- die Erarbeitung von Use Cases für die datenbasierte Analyse von Sicherheitsvorfällen im Kontext des autonomen Fahrens (mit Fokus auf IT-Sicherheit und entsprechende Angriffsszenarien),
- die Analyse der zu erfassenden Daten hinsichtlich ihrer Eignung zur Erkennung von Sicherheitsvorfällen auf Basis erarbeiteter Angriffsszenarien,
- die Beratung und Unterstützung beim sicheren Design der Monitoring-Plattform und die Beratung hinsichtlich des Deployment-Prozesses,
- die Sicherheitsüberprüfung der Monitoring-Plattform (insbesondere des Demonstrators) durch etablierte Analysetechniken wie Fuzzing und Source Code Audits.

Im Zuge der kryptographischen Absicherung der Monitoring-Plattform hinsichtlich Datenspeicherung, Datenaustausch und Deployment-Prozess wurde die leichtgewichtige Stromchiffre DRACO entworfen, kontinuierlich weiterentwickelt und in einer Authenticated-Encryption-Variante für die Monitoring-Plattform selbst sowie für deren Backend implementiert.

2.5 Zusammenarbeit mit anderen Forschungseinrichtungen

Während des gesamten Projekts wurde in stetigem Austausch mit den Konsortialpartnern zusammengearbeitet. In bilateraler Zusammenarbeit mit dem Projektpartner KIT entstandenen die Publikationen [SHB23] und [SHTB23]. In Zusammenarbeit mit sämtlichen Projektpartnern entstand die Publikation [SLS+24]. Die Stromchiffre DRACO [HMKM22, HMKM23] entstand in Zusammenarbeit mit der Arbeitsgruppe Theoretische Informatik und IT-Sicherheit der Universität Mannheim und unter Beteiligung der Universität Siegen.

ERNW RESEARCH
pursuing knowledge.

Literaturverzeichnis

[AM15] Frederik Armknecht and Vasily Mikhalev. *On Lightweight Stream Ciphers with Shorter Internal States*. In FSE 2015, pages 451–470. Springer, 2015.

[HKMZ18] Matthias Hamann, Matthias Krause, Willi Meier, and Bin Zhang. *Design and Analysis of Small-state Grain-like Stream Ciphers*. Cryptography and Communications, 10(5):803–834, 2018.

[HMKM22] Matthias Hamann, Alexander Moch, Matthias Krause, and Vasily Mikhalev. *The DRACO Stream Cipher: A Power-efficient Small-state Stream Cipher with Full Provable Security against TMDTO Attacks*. IACR Transactions on Symmetric Cryptology, 2022(2), 1–42.

[HMKM23] Matthias Hamann, Alexander Moch, Matthias Krause, and Vasily Mikhalev. *The DRACO Stream Cipher – FSE 2023 Presentation*. *https://iacr.org/submit/files/slides/2023/fse/fse2023/tosc2022_2_14/slides.pdf*. FSE 2023. 2023-03-20, Kobe, Japan.

[MAM16] Vasily Mikhalev, Frederik Armknecht, and Christian Müller. *On Ciphers that Continuously Access the Non-volatile Key*. IACR Transactions on Symmetric Cryptology, 2016(2), 52–79.

[SHB23] Matthias Stammler, Matthias Hamann, and Jürgen Becker. *Multilevel Security Model for Secure Information Flow Inside Software Components Employing Automated Code Generation*. 2023 12th Mediterranean Conference on Embedded Computing (MECO). IEEE, 2023.

[SHTB23] Matthias Stammler, Matthias Hamann, Tanja Harbaum, and Jürgen Becker. *Mitigating Masking in Automotive Communication Systems: Modeling and Hardware Generation*. 2023 26th Euromicro Conference on Digital System Design (DSD). IEEE, 2023.

[SLS+24] Matthias Stammler, Julian Lorenz, Eric Sax, Jürgen Becker, Matthias Hamann, Patrick Bidinger, Andreas Dewald, Paraskevi Georgouti, Alexios Camarinopoulos, Günter Becker, Klaus Finsterbusch, Maximilian Kirschner, Laurenz Adolph, Carl Philipp Hohl, Maria Rill, Daniel Vonderau, Victor Pazmino. *UNCOVER: Data-Driven Design Support through Continuous Monitoring of Security Incidents*. 2024 Design, Automation and Test in Europe Conference and Exhibition (DATE).

ERNW RESEARCH
pursuing knowledge.

BMBF UNCOVER

ABSCHLUSSBERICHT ZUM TEILVORHABEN

Version:	1.0
Datum:	20.06.2024
Klassifikation:	Intern
Verbundprojekt:	Datenbasierte EntwurfsUNterstützung durch kontinuierliches MOnitoring Von SichERheitsvorfällen (UNCOVER)
Verbundpartner:	ERNW Research GmbH
Förderkennzeichen:	16KIS1410K
Projektlaufzeit:	01.06.2021–31.12.2023

ERNW RESEARCH
pursuing knowledge.

Inhaltsverzeichnis

1 Informationen zur Handhabung 6
1.1 Dokumenten-Status und -Besitzer . . . 6
1.2 Erläuterung der Klassifizierung . . . 6
1.3 Versionshistorie . . . 7

2 Aufgabenstellung und Einordnung des Teilvorhabens 8
2.1 Voraussetzungen . . . 9
2.2 Planung und Ablauf . . . 9
2.3 Ausgangssituation, verwendete Standards und Quellen . . . 10
2.4 Zusammenarbeit mit anderen Stellen . . . 10

3 Ergebnisübersicht des Teilvorhabens 11
3.1 Ergebnisse AP 1 . . . 11
3.2 Ergebnisse AP 2 . . . 13
3.3 Ergebnisse AP 3 . . . 13
3.4 Ergebnisse AP 4 . . . 13
3.5 Ergebnisse AP 5 . . . 13
3.6 Ergebnisse AP 6 . . . 14

4 Detaildarstellung der Ergebnisse von AP 3 15
4.1 Designkonzept der Monitoring-Plattform . . . 15
4.2 Kryptographische Absicherung der Monitoring-Plattform . . . 16
4.2.1 Stromchiffren und deren typische Einsatzbereiche . . . 17
4.2.2 Time-Memory-Data Tradeoff Angriffe . . . 18
4.2.3 Neue Ansätze zur Entwicklung leichtgewichtiger Stromchiffren . . . 18
4.2.4 DRACO . . . 19
4.2.5 Aktuelle Weiterentwicklungen: DRACO v2 and DRACO-PQ . . . 20
4.3 Sicherheitsprüfung der Monitoring-Plattform . . . 21
4.3.1 Penetration-Tests . . . 21
4.3.2 Source Code Audit . . . 30
4.3.3 Untersuchung der Update/Deployment-Mechanismen . . . 31

5 Verwertung und Veröffentlichungen 33
5.1 Wissenschaftliche Verwertbarkeit und Anschlussfähigkeit der Arbeiten . . . 33
5.2 Verwertbarkeit im Projektanschluss . . . 33

5.3 Bekanntgewordener Fortschritt an anderen Stellen 34
5.4 Bezug zum zahlenmäßigen Nachweis 34
5.5 Veröffentlichungen 34

ERNW RESEARCH
pursuing knowledge.

Abbildungsverzeichnis

Abbildung 1: Fahrzeugsetup zur Diskussion von Angriffsszenarien in ISO 21434 (Anhang G). (Abbildung aus [DNR21]) . . . 11
Abbildung 2: Mögliche Monitoring-Positionen im Fahrzeug. (Abbildung aus [DNR21]) . . . 15
Abbildung 3: DRACO in der Phase der Schlüsselstromerzeugung. . . . 19
Abbildung 4: Demonstrator der UNCOVER Monitoring-Plattform. . . . 23
Abbildung 5: Beispielhafte Busmaster Log-Datei. . . . 24
Abbildung 6: Fuzzing-Analyse der Monitoring-Plattform. . . . 27
Abbildung 7: "Degradation of Service"-Finding der Fuzzing-Tests. . . . 28

ERNW RESEARCH
pursuing knowledge.

Tabellenverzeichnis

Tabelle 1: Dokumenteneigenschaften 6
Tabelle 2: Dokumentenklassifizierungen 6
Tabelle 3: Versionshistorie 7

1 Informationen zur Handhabung

Das vorliegende Dokument ist als *Intern* klassifiziert. Jeglicher Weitergabe oder Offenlegung des Dokuments MUSS die ausdrückliche Freigabe durch den in Abschnitt *Dokumenten-Status und -Besitzer* benannten Dokumentenbesitzer vorausgehen.

1.1 Dokumenten-Status und -Besitzer

Der Besitzer ist als Inhaber des vorliegenden Dokuments ausschließlich entscheidungsbefugt über die Weitergabe dieses Dokuments und verantwortlich für die Distribution der jeweils gültigen Fassung.

Mögliche Einträge für den Status des Dokumentes sind *Vor-Entwurf*, *Entwurf*, *aktuell gültig* und *veraltet*.

Dokumenteneigenschaften	
Titel:	BMBF UNCOVER - Abschlussbericht zum Teilvorhaben
Dokumentenbesitzer:	ERNW Research GmbH
Version:	1.0
Status:	Effective
Klassifikation:	Intern
Autor(en):	Andreas Dewald, Matthias Hamann, Patrick Bidinger

Tabelle 1: Dokumenteneigenschaften

1.2 Erläuterung der Klassifizierung

Klassifizierung	Empfänger
Öffentlich:	Jeder
Intern:	Alle Mitarbeiter und Kunden
Vertraulich:	Nur Mitarbeiter
Geheim:	Nur dedizierte Mitarbeiter

Tabelle 2: Dokumentenklassifizierungen

1.3 Versionshistorie

Version	Datum	Details
1.0	20.06.2024	Initiale Version.

Tabelle 3: Versionshistorie

ERNW RESEARCH
pursuing knowledge.

2 Aufgabenstellung und Einordnung des Teilvorhabens

Das Ziel des Teilvorhabens der ERNW Research GmbH im BMBF-Projekt UNCOVER war die Beratung zu und Untersuchung von IT-Sicherheit hinsichtlich Konzeption, Design und Umsetzung von Entwicklungsmethoden und Werkzeugunterstützung zur Analyse von Security-Vorfällen im Automotive-Umfeld.

Für die Realisierung und Einführung autonom agierender Fahrzeuge ist das sichere Zusammenspiel von Funktionen, Systemen und Diensten sowie deren Überwachung über den kompletten Produktlebenszyklus unerlässlich. Diese Kooperation findet sowohl innerhalb des Fahrzeugs, in Form von Kommunikation zwischen Steuergeräten, als auch mit der Umwelt im Sinne von mobilen Endgeräten und Infrastruktur statt. Dabei spielen neben interagierenden Fahrzeugen auch übergreifende Informationsdienste wie Navigations- oder Wetterdienste eine zentrale Rolle. Diese Informationen fließen im autonom agierenden Fahrzeug zusammen, um Entscheidungen über Navigation, Spurführung und Stabilisierung auf unterschiedlichen Funktionsebenen im Fahrzeug treffen zu können. Da diese Informationen bei Fehlern oder Manipulation zu Fehlverhalten des Fahrzeugs bis hin zu Gefahrensituationen führen können, müssen sie besonders geschützt werden. Dadurch gewinnt die IT-Sicherheit in den vernetzten Systemen und Anwendungen sowie der Einsatz von sicheren und vertrauenswürdigen Informations- und Kommunikationstechnologien (IKT) immer mehr an Bedeutung bei der Entwicklung und dem Betrieb solcher Systeme.

Zur sicheren Entwicklung und Auslegung der Fahrzeuge werden im Rahmen eines Security-by-Design Maßnahmen getroffen, um möglichst viele Risiken bereits zur Entwurfszeit abzudecken. Jedoch können sich während der Betriebszeit zusätzliche Schwachstellen und Sicherheitslücken ergeben, die zur Entwurfszeit nicht bekannt waren. Insbesondere bei autonomen Systemen sind Änderungen während des Betriebs zu erwarten. Diese Änderungen können sowohl das System selbst als auch seine Umwelt betreffen:

- Selbstlernende Systeme basierend auf KI-Anteilen, die sich durch Lernen eigenständig weiterentwickeln.
- Änderungen an der Umwelt (neue Kommunikationsmöglichkeiten, neue Verkehrsteilnehmer wie z. B. E-Scooter, veränderte Rahmenbedingungen, Gesetzgebung, ...).
- Änderungen/Updates von bestehenden Funktionen im Fahrzeug, durch die neue Kommunikationspfade entstehen.

Die kontinuierlichen Veränderungen bergen potenzielle Risiken in Bezug auf die Cyber-Sicherheit, die zugleich durch ihre Auswirkung auf die funktionale Sicherheit und insbesondere auf die Erfüllung der Sollfunktion (SOTIF) zu erheblichen Gefährdungen für Verkehrsteilnehmer führen können. So besteht erhebliches Potenzial für wirtschaftliche Schäden, die bspw. durch Datenmissbrauch und Datenmanipulation entstehen können. Zudem ist es durch die Veränderungen der Umwelt sowie des Systems zwingend notwendig, über den gesamten Produktlebenszyklus der Fahrzeuge und Systeme bis hin zur Außerbetriebnahme kontinuierlich die Sicherheit zu gewährleisten. Eine ausschließliche entwicklungsbegleitende Unterstützung im Rahmen eines Security-by-Design ist nicht mehr ausreichend und muss durch die Rückführung von Erfahrungen aus dem Betrieb kontinuierlich unterstützt werden.

Ein essentieller Baustein des Gesamtvorhabens war die Entwicklung einer flexiblen Monitoring-Plattform zur Erkennung sicherheitsrelevanter Ereignisse im laufenden Betrieb. Sie wurde in Form eines Frameworks entworfen und stellt die technische Grundlage zur Integration konfigurierbarer Monitoring-Mechanismen (sog. Templates) in eine E/E-Architektur dar. Bei diesen Templates handelt es sich um Hardware- und Softwarebausteine, die eine Erkennung bezüglich safety- und securityrelevanter Ereignisse erlauben. Sie sollen während des Entwicklungsprozesses parametriert und flexibel in verschiedene Komponenten wie ECUs oder Fahrzeugbusse der E/E-Architektur integriert werden können. Beitrag der ERNW Research GmbH war hierbei einerseits, bzgl. der Erfassung der Daten zu beraten und die Erfahrungen aus praktischen Sicherheitsuntersuchungen einfließen zulassen, und andererseits, durch Beratung beim Design und Durchführung eines Assessments die Sicherheit der Plattform selbst zu gewährleisten.

Zentrale Punkte dieser Beratungs- und Unterstützungstätigkeit durch die ERNW Research GmbH waren dabei:

- Erarbeitung von Use Cases für die datenbasierte Analyse von Sicherheitsvorfällen im Kontext des autonomen Fahrens (mit Fokus auf IT-Sicherheit und entsprechende Angriffsszenarien).
- Analyse der zu erfassenden Daten hinsichtlich ihrer Eignung zur Erkennung von Sicherheitsvorfällen auf Basis erarbeiteter Angriffsszenarien.
- Beratung und Unterstützung beim sicheren Design der Monitoring-Plattform (bspw. hinsichtlich des Einsatzes geeigneter kryptographischer Verfahren) und Beratung hinsichtlich des Deployment-Prozesses.
- Sicherheitsüberprüfung der Monitoring-Plattform (insbesondere des Demonstrators) durch etablierte Analysetechniken wie Fuzzing und Source Code Audits.

2.1 Voraussetzungen

Das Projekt fand im Rahmen der Fördermaßnahme "KMU-innovativ: Informations- und Kommunikationstechnologie (IKT)" im Technologiebereich "Kommunikationssysteme, IT-Sicherheit (KIS)" statt.

Dem Projekt standen dabei keine realen Felddaten autonom fahrender Fahrzeuge zur Verfügung, auf Basis derer eine entsprechende Konzeption und anschließende Tests der zu entwickelnden Monitoring-Plattform hätten stattfinden können. Dadurch war es unter anderem notwendig, sinnvolle Fahr- und Fahrzeugszenarien (sowohl für die störungsfreie Nutzung als auch, im Kontrast dazu, im Kontext von Angriffen) zu erarbeiten und diese im Projektverlauf kontinuierlich auf Plausibilität und Eignung zu prüfen.

2.2 Planung und Ablauf

Das Gesamtprojekt war in die folgenden Arbeitspakete unterteilt:

- AP 1: Use Case Definition und Anforderungen an die Daten und Sicherheitsanalysen

- AP 2: Methoden und Werkzeug für kontinuierliche Security- und Safety-Analysen basierend auf den Erkenntnissen aus Security- und Safety-Vorfällen
- AP 3: Monitoring-Plattform zur Erkennung von Security- und Safety-Vorfällen
- AP 4: Methoden für datengetriebene Unterstützung der Entwurfsphase
- AP 5: Datenschutzrechtliche Betrachtung
- AP 6: Evaluation und Demonstration

Die ERNW Research GmbH hat zu den Arbeitspaketen AP 1, AP 3, AP 4 und AP 6 beigetragen. Das Arbeitspaket AP 3 wurde darüber hinaus von der ERNW Research GmbH geleitet. An den Arbeitspaketen AP 2 und AP 5 hatte die ERNW Research GmbH keine direkte Beteiligung.

Wegen Verzögerungen auf Verbundebene, unter anderem bedingt durch die Ausläufer der COVID-19-Pandemie, kam es zu einer Projektverzögerung von einem Monat. Das Projekt wurde entsprechend kostenneutral bis zum 31. Dezember 2023 verlängert.

2.3 Ausgangssituation, verwendete Standards und Quellen

Während des Projekts wurden insbesondere die folgenden Standards berücksichtigt:

- ISO/SAE 21434 "Road vehicles – Cybersecurity engineering"
- ISO 21448 "Road vehicles – Safety of the intended functionality" (SOTIF)

Weitere verwendete Quellen werden im Anhang genannt und in den jeweiligen Berichtsabschnitten referenziert.

Die umfassende wissenschaftliche Ausgangssituation wird im Gesamtbericht des Vorhabens detailliert beschrieben. Für dieses Teilvorhaben relevante Grundlagen werden nachfolgend direkt als Teil der jeweiligen Kapitel erläutert (siehe bspw. Kapitel 4.2.2 und 4.2.3).

2.4 Zusammenarbeit mit anderen Stellen

Während des gesamten Projekts wurde in stetigem Austausch mit den Konsortialpartnern zusammengearbeitet. Die in Kapitel 4.2 beschriebene Stromchiffre DRACO wurde in Zusammenarbeit mit der Arbeitsgruppe Theoretische Informatik und IT-Sicherheit der Universität Mannheim und unter Beteiligung der Universität Siegen entwickelt.

3 Ergebnisübersicht des Teilvorhabens

In diesem Kapitel wird eine Übersicht über die Ergebnisse des Teilvorhabens gegeben. Eine detaillierte Darstellung besonders wichtiger Ergebnisse aus AP 3 findet separat in Kapitel 4 statt.

3.1 Ergebnisse AP 1

Im Rahmen von AP1 wurden zusammen mit den Projektpartnern die Use Cases und Anforderungen des autonomen Fahrens für die datenbasierte Analyse von Sicherheitsvorfällen erarbeitet. Hier wurde durch die ERNW Research GmbH eine intensive Recherche hinsichtlich bestehender Standards und Publikationen durchgeführt, welche insbesondere ergaben, dass sich die in ISO 21434 (Anhang G) genannten Angriffsszenarien sehr gut als Ausgangspunkt für die IT-sicherheitsbezogenen Design- und Prüfaspekte des UNCOVER-Projekts eignen. Die betreffenden Angriffsszenarien fußen auf dem in Abbildung 1 dargestellten Fahrzeugsetup.

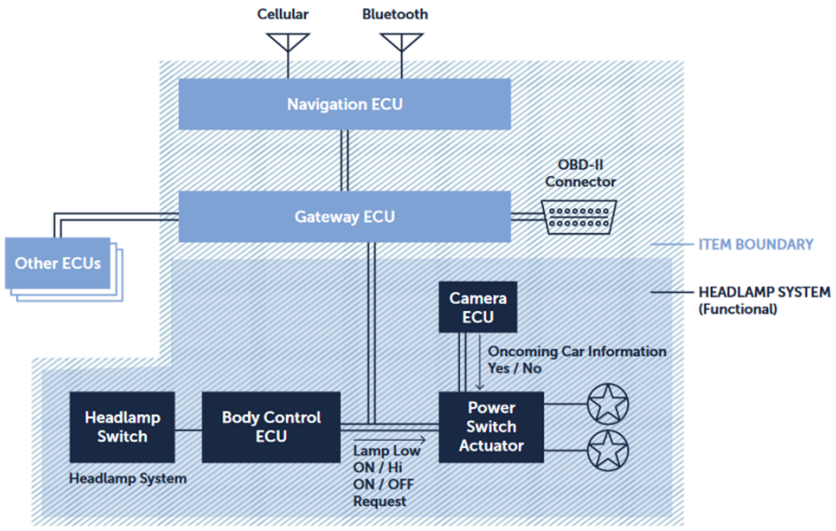

Abbildung 1: Fahrzeugsetup zur Diskussion von Angriffsszenarien in ISO 21434 (Anhang G). (Abbildung aus [DNR21])

Wie in [DNR21] beschrieben, geht es hierbei um ein Scheinwerfersystem, welches bspw. bei auftretendem Gegenverkehr automatisch von Fern- auf Abblendlicht wechselt. Als Gefahren werden u. a. Angriffe auf die Integrität (bspw. via Spoofing) und die Verfügbarkeit (bspw. via Flooding) des CAN-Busses betrachtet, welche zu einem unerwünschten Verhalten des Scheinwerfersystems und damit verbundenen Unfallszenarien (bspw. *Ausfall des Lichts → Abkommen von der Straße* oder *plötzliches Aufblenden in den Gegenverkehr → Kollision*) führen. Als mögliche 'Einfallstore' können hier, wie in Abbildung 1 dargestellt, beispielsweise die Mobilfunk-/Internetverbindung, eine Bluetooth-Verbindung oder ein bösartiges CAN-Bus-Dongle dienen.

Zudem ergeben sich aus dem möglichen Einsatz von KI im Rahmen des Monitorings neben klassischen Angriffsarten (wie bspw. dem Versuch, über den CAN-Bus in das Monitoring-Modul 'einzudringen') auch neue Gefahren wie bspw. der Versuch eines Angreifers, die KI 'falsch zu trainieren' und dadurch Sicherheitsvorfälle (bspw. über False Positives/Negatives des Monitorings → unerwünschte/keine Reaktion) zu provozieren. Dies beträfe insbesondere AP 2 ("Methoden und Werkzeug für kontinuierliche Security- und Safety-Analysen basierend auf den Erkenntnissen aus Security- und Safety-Vorfällen").

Weiterhin könnte die im Rahmen des Projekts betrachtete Rückführung der Monitoring-Ergebnisse in den Entwicklungsprozess durch einen Angreifer ausgenutzt werden, um mittels Manipulation von Monitoring-Daten Einfluss auf genau diesen Entwicklungsprozess zu nehmen. Dies beträfe insbesondere AP 4 ("Methoden für datengetriebene Unterstützung der Entwurfsphase"). Während einer entsprechenden Manipulation von Monitoring-Daten sowohl im Speicher der Monitoring-Plattform als auch beim Transfer zwischen Monitoring-Plattform und Backend durch den Einsatz von Authenticated Encryption entgegengewirkt wird (vgl. Kapitel 4.2), wurde die Erkennung von bspw. bereits während der Entstehung im Fahrzeug (mit dem Ziel einer Beeinflussung der KI des Backends bzw. des Entwicklungsprozesses) manipulierten Daten durch entsprechende Analysen im Backend nach entsprechenden Diskussionen im Konsortium als außerhalb des Projektumfangs eingeordnet.

Neben der Auswertung relevanter Standards wie ISO 21434 (Anhang G) wurden durch die ERNW Research GmbH in AP 1 auch Informationen aus tatsächlichen Sicherheitsvorfällen beigetragen, wie beispielsweise dem Hack eines Jeep Cherokees, durch welchen Angreifer per Mobilfunk dem tatsächlichen Fahrer weitgehend die Kontrolle entziehen und zentrale Fahrfunktionen selbst steuern konnten [MV15]. Darüber hinaus flossen umfangreiche Erfahrungen der ERNW Research GmbH aus der praktischen Durchführung von Sicherheitsuntersuchungen in die Erarbeitung der Use Cases mit ein. Alle entsprechenden Informationen wurden mit Blick auf ihre Anwendbarkeit und Übertragbarkeit im Projekt analysiert und fanden zudem Eingang in die Erarbeitung und Definition von Schnittstellen zwischen den APs.

Kernpunkt des dem Projekt zugrundeliegenden Angreifermodells ist, dass sich eine ECU des Fahrzeugs bereits vollständig unter der Kontrolle eines Angreifers befindet. Wie es zu dieser initialen Kompromittierung gekommen ist (bspw. über einen Hack des Kommunikations- oder Entertainmentsystems), hat im Projektkontext eine untergeordnete Bedeutung. Entscheidend ist vielmehr, dass der Angreifer mittels der kompromittierten ECU in der Lage ist, beliebige Nachrichten auf dem CAN-Bus des Fahrzeugs zu versenden und zu empfangen bzw. mitzuhören. Dies eröffnet ihm eine Vielzahl potentiell hochkritischer weiterer Angriffspfade, bspw. über das Versenden falscher GPS-/Radar-/Lidar-Daten an Steuerungssysteme des Fahrzeugs. Ziel des Projekts war es entsprechend, durch Beobachtung des CAN-Busses selbst in diesem mächtigen Angreifermodell bösartige Aktionen detektieren und in Zukunft verhindern zu können.

Auch nach Abschluss von AP 1 wurde die Unterstützung und Beratung der Projektpartner in Sicherheitsfragen durch die ERNW Research GmbH kontinuierlich fortgesetzt. Insbesondere wurde das Verständnis hinsichtlich der im Rahmen von AP1 erarbeiteten Use Cases und Anforderungen des autonomen Fahrens im Hinblick auf IT-Sicherheit weiter vertieft.

Dazu fand durch die ERNW Research GmbH eine weitergehende Sichtung von aktuellen Standards und Publikationen statt, die IT-sicherheitsbezogene Design- und Prüfaspekte des UNCOVER-Projekts berührten.

3.2 Ergebnisse AP 2

Keine Beteiligung der ERNW Research GmbH.

3.3 Ergebnisse AP 3

Die Tätigkeiten der ERNW Research GmbH im Rahmen von AP 3 umfassten insbesondere die Beratung und Unterstützung beim sicheren Design der Monitoring-Plattform (bspw. hinsichtlich des Einsatzes geeigneter kryptographischer Verfahren). Dazu fand über den gesamten Projektzeitraum hinweg ein intensiver Austausch mit dem für die konkrete Implementierung der Monitoring-Plattform verantwortlichen Projektpartner KIT statt. Dieser Austausch resultierte in mehreren gemeinsamen Publikationen (vgl. Kapitel 5.5), welche unter anderem die Grundlagen für das Designkonzept der Monitoring-Plattform legen. Details dieses Designkonzepts werden in Kapitel 4.1 vorgestellt.

Zur kryptographischen Absicherung der Monitoring-Plattform im Hinblick auf Datenspeicherung, Datenaustausch und Deployment-Prozess wurde durch die ERNW Research GmbH als Teil von AP 3 eine leichtgewichtige Stromchiffre, welche Authenticated Encryption bietet, entwickelt und für die Monitoring-Plattform selbst sowie für deren Backend implementiert. Entsprechende Details werden in Kapitel 4.2 beschrieben.

Teil von AP 3 war zudem eine Sicherheitsüberprüfung der Monitoring-Plattform (insbesondere des Demonstrators) mittels etablierter Analysetechniken wie Fuzzing und Source Code Audits. Eine Beschreibung des jeweiligen Vorgehens und eine exemplarische Vorstellung entsprechender Ergebnisse findet sich in Kapitel 4.3.

3.4 Ergebnisse AP 4

Im Rahmen dieses APs wurden durch die ERNW Research GmbH die von den Projektpartnern betrachteten Daten auf deren Eignung hin analysiert, einen Sicherheitsvorfall auf dieser Basis detektieren zu können. Entsprechende Einschätzungen und Verbesserungsvorschläge wurden kontinuierlich an die beteiligten Partner rückgemeldet.

3.5 Ergebnisse AP 5

Keine Beteiligung der ERNW Research GmbH.

3.6 Ergebnisse AP 6

Die Bereits im Rahmen von AP 3 über den Projektzeitraum hinweg kontinuierlich durch die ERNW Research GmbH geleistete Unterstützung der Projektpartner beim Design und der Absicherung der Monitoring-Plattform (vgl. Kapitel 4) wurde hier mit Fokus auf den entwickelten Demonstrator fortgesetzt. Ein weiterer Fokus der als Teil von AP 6 durchgeführten Beratung durch die ERNW Research GmbH lag darauf zu gewährleisten, dass der Demonstrator die IT-sicherheitsbezogenen Fähigkeiten der Monitoring-Plattform adäquat verkörpert.

4 Detaildarstellung der Ergebnisse von AP 3

Im Folgenden werden die wichtigsten unter Beteiligung der ERNW Research GmbH erzielten Ergebnisse von AP 3 dargestellt.

4.1 Designkonzept der Monitoring-Plattform

Im Rahmen der in Kapitel 3.1 skizzierten Use Cases werden in [DNR21] auch die in Abbildung 2 dargestellten Monitor-Varianten zur Gefahrenerkennung aufgeführt.

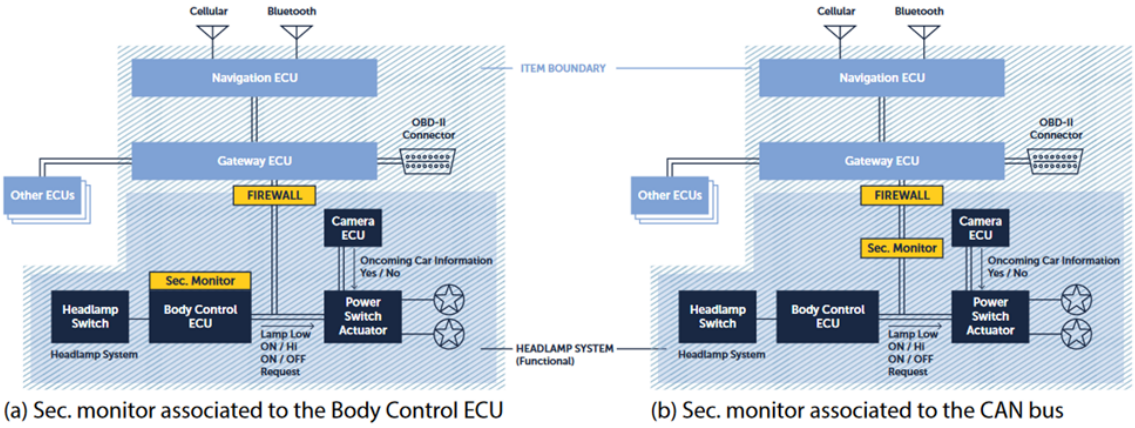

Abbildung 2: Mögliche Monitoring-Positionen im Fahrzeug. (Abbildung aus [DNR21])

Diese passen zu den initial durch den Projektpartner KIT vorgeschlagenen Beispielarchitekturen, sodass die dargestellten Use Cases (gerade auch, weil sie direkt aus ISO 21434 (Anhang G) stammen) als für den Projektkontext geeignet identifiziert wurden. Auch eine Erweiterung des Angriffsszenarios ist denkbar, bspw. hinsichtlich der bösartigen Täuschung der Camera ECU mittels Laser (→ Fahrzeug 'denkt' fälschlich, es sei Tag, und schaltet die Scheinwerfer ganz aus, was mangels Sicht zum Abkommen von der Fahrbahn führt). Hierdurch würde auch der Aspekt eines Angriffs abgedeckt, welcher nicht auf die Integrität oder die Verfügbarkeit des CAN-Busses abzielt.

Für die Umsetzung des Demonstrators in AP 6 kam die als Variante (b) in Abbildung 2 dargestellte Positionierung der Monitoring-Plattform als separate, auf dem CAN-Bus 'lauschende' Komponente zum Einsatz. Konzeptionell ist die im Rahmen von AP 3 entwickelte Monitoring-Plattform aber auch unmittelbar zur Umsetzung von Variante (a) in Abbildung 2 geeignet. Dazu müssten nicht einmal separate Instanzen der Monitoring-Plattform in den einzelnen ECUs platziert werden. Vielmehr würde für jede zu überwachende ECU ein entsprechender, leichtgewichtiger Sensor ausreichen, welcher die Daten der betreffenden ECU (bspw. auch drahtlos) an die zentrale Monitoring-Plattform des Fahrzeugs wei-

terleitet. Auch hier spielt die in Kapitel 4.2 beschriebene, leichtgewichtige Stromchiffre mit Authenticated Encryption eine entscheidende Rolle beim sicheren, ressourceneffizienten Datenaustausch.

Unter Berücksichtigung der Ergebnisse der in den Kapitel 3.1 und 3.4 erwähnten Recherche- und Analysetätigkeiten wurde eine Menge von Anforderungen an die Fähigkeiten der Monitoring-Plattform identifiziert, um die im Projekt angestrebte datenbasierte Erkennung und Analyse von Sicherheitsvorfällen umsetzen zu können. Zu diesen entsprechend in die Monitoring-Plattform eingebrachten, notwendigen Fähigkeiten zählen bspw. die Erfassung von Daten auf dem überwachten Bussystem samt anschließender Verarbeitung auf Basis von

- Sender und Empfänger,
- Verletzung von Wertegrenzen bestimmter Daten,
- zeitlichen Zusammenhängen zwischen bestimmten Nachrichten.

Besonders hervorzuheben hinsichtlich der Erarbeitung des Designkonzepts der Monitoring-Plattform sind zwei im Rahmen von AP 3 in Zusammenarbeit mit dem Projektpartner KIT entstandene Publikationen, welche die Modellierung und Hardwaregenerierung von Monitoring-Systemen im Automotive-Kontext behandeln:

- Matthias Stammler, Matthias Hamann, Jürgen Becker. *Multilevel Security Model for Secure Information Flow Inside Software Components Employing Automated Code Generation*. 2023 12th Mediterranean Conference on Embedded Computing (MECO). IEEE, 2023. [SHB23]
- Matthias Stammler, Matthias Hamann, Tanja Harbaum, Jürgen Becker. *Mitigating Masking in Automotive Communication Systems: Modeling and Hardware Generation*. 2023 26th Euromicro Conference on Digital System Design (DSD). IEEE, 2023. [SHTB23]

Für Details zu diesen Arbeiten wird auf den Abschlussbericht zum Teilvorhaben des Projektpartners KIT verwiesen.

4.2 Kryptographische Absicherung der Monitoring-Plattform

Ein weiterer wichtiger Teil von AP 3 war die Entwicklung der neuen, hochgradig effizienten Stromchiffre DRACO für ressourcenbeschränkte Geräte. Deren Vorstellung fand auf der FSE 2023[1] in Kobe, Japan, statt, welche als international führende Konferenz im Bereich der symmetrischen Kryptographie gilt:

- Matthias Hamann, Alexander Moch, Matthias Krause, Vasily Mikhalev. *The DRACO Stream Cipher: A Power-efficient Small-state Stream Cipher with Full Provable Security against TMDTO Attacks*. IACR Transactions on Symmetric Cryptology (ToSC), 2022(2): 1–42. [HMKM22]
- Matthias Hamann, Alexander Moch, Matthias Krause, Vasily Mikhalev. *The DRACO Stream Cipher – FSE 2023 Presentation* (mit zusätzlichen, neuen Vorschlägen für ein Update des Original DRACO Key Schedules). *https://iacr.org/submit/files/slides/2023/fse/fse2023/tosc2022_2_14/slides.pdf*. FSE 2023. 2023-03-20, Kobe, Japan. [HMKM23]

[1] *https://fse.iacr.org/2023/*

DRACO wurde durch die ERNW Research GmbH in enger Zusammenarbeit mit der Arbeitsgruppe Theoretische Informatik und IT-Sicherheit der Universität Mannheim und unter Beteiligung der Universität Siegen entwickelt.

Die Chiffre erlaubt, bei angestrebt unverändert hohem Sicherheitsniveau, eine Reduktion der Chip Area um über 20 Prozent und eine Reduktion der Power Consumption um über 30 Prozent verglichen mit bestehenden, für solche Szenarien eingesetzten Stromchiffren wie bspw. Grain-128a. Dadurch ist DRACO im Kontext von UNCOVER nicht nur sehr gut zur Absicherung des Datenaustauschs der FPGA-basierten, ressourcenbeschränkten Monitoring-Plattform mit dem Backend geeignet, sondern eröffnet dieser auch weitreichendere Anwendungsmöglichkeiten. Konkret ist so etwa eine zukünftige Erweiterung der Datenerfassung für das Monitoring durch Platzierung von Sensoren an kritischen Stellen des Fahrzeugs denkbar (vgl. Abbildung 2 in Kapitel 4.1). Während solche Sensoren klassischerweise hochgradig ressourcenbeschränkt sind, muss angesichts der potentiell sensiblen Daten, die von ihnen an die Monitoring-Plattform übertragen würden, trotzdem ein hohes Sicherheitsniveau gewährleistet werden. DRACO bietet hierfür die entsprechenden Möglichkeiten. Da neben Vertraulichkeit auch Authentizität und Integrität der zwischen der Monitoring-Plattform und dem Backend ausgetauschten Daten gewährleistet werden müssen, wurde DRACO im Jahr 2023 um entsprechende Funktionalitäten erweitert und diese FPGA-basiert in die Monitoring-Plattform integriert (vgl. Kapitel 4.2.5).

Wichtig zu erwähnen ist hierbei auch, dass die durch die Monitoring-Plattform zur Übertragung an das Backend aufgezeichneten Daten nicht erst im Rahmen der Datenübertragung, sondern vielmehr bereits direkt bei ihrer zunächst lokalen Speicherung im Fahrzeug per Authenticated Encryption geschützt werden. Dadurch wird ein unbefugtes Auslesen bzw. Manipulieren dieser sensiblen Daten auch im Falle eines Angreifers mit physischen Zugriff auf das Fahrzeug abgewehrt, solange der verwendete symmetrische Schlüssel (bspw. durch Einsatz eines Hardware-Sicherheitsmoduls) geheim bleibt.

Eine zentrale Rolle kommt der für UNCOVER entwickelten, leichtgewichtigen Authenticated-Encryption-Lösung auf Basis von DRACO auch im Rahmen des Update/Deployment-Prozesses zu. Kapitel 4.3.3 enthält eine detaillierte Beschreibung entsprechender Szenarien und Vorgehensweisen.

4.2.1 Stromchiffren und deren typische Einsatzbereiche

Im Rahmen der symmetrischen Kryptographie unterscheidet man typischerweise zwei Arten von Verschlüsselungsschemata: Blockchiffren und Stromchiffren. Blockchiffren unterteilen einen Klartext in Blöcke fester Größe (z. B. 64 oder 128 Bit) und verschlüsseln einen solchen Datenblock als Ganzes. Stromchiffren hingegen betrachten den Klartext als einen kontinuierlichen Datenstrom. Stromchiffren arbeiten taktweise auf Basis eines geheimen internen Zustands, welcher sich klassischerweise aus einem oder mehreren rückgekoppelten Schieberegistern zusammensetzt. In jedem Schritt werden dabei ein oder mehrere sogenannte Schlüsselstrombits auf Basis des internen Zustands berechnet und ausgegeben. Anschließend werden die Schieberegister des internen Zustands mittels ihrer Feedback-Funktionen aktualisiert. Der geheime Schlüsselstrom wird schließlich mit dem Klartext kombiniert, normalerweise unter Verwendung

der XOR-Operation. Ein Vorteil von Stromchiffren besteht darin, dass ihr Ressourcenbedarf in vielen Anwendungsszenarien, insbesondere bei Einsatz von Field-Programmable Gate Arrays (FPGAs) oder Application-Specific Integrated Circuits (ASICs), deutlich geringer als der von Blockchiffren ist. Dies macht sie besonders geeignet für leichtgewichtige Kryptografie, welche auf ressourcenbeschränkte Geräte wie kostengünstige RFID-Tags oder auch in einem Fahrzeug verteilte Sensoren abzielt.

Als Teil des UNCOVER-Projekts hat die ERNW Research GmbH (Matthias Hamann) daher in Zusammenarbeit mit der Universität Mannheim (Alexander Moch, Matthias Krause) und der Universität Siegen (Vasily Mikhalev) die neue leichtgewichtige Stromchiffre DRACO entwickelt.

4.2.2 Time-Memory-Data Tradeoff Angriffe

Stromchiffren sind anfällig gegenüber sogenannten Time-Memory-Data (TMD-TO) Tradeoff Angriffen [Bab95, Gol96, BS00]. Solche Angriffe nutzen das Geburtstagsparadoxon, um einen der im Rahmen der Schlüsselstromgenerierung auftretenden geheimen internen Zustände aufzudecken. Dieser kann dann (durch simples Weitertakten) zur Entschlüsselung des restlichen Chiffretexts und häufig (durch Rückwärtstakten) sogar zur einfachen Wiederherstellung des geheimen Schlüssels verwendet werden. Aufgrund des Geburtstagsparadoxons ist die Sicherheit solcher Chiffren daher typischerweise auf die Hälfte der Größe des internen Zustands begrenzt. Dies hat das Design von Stromchiffren in der Vergangenheit dahingehend beeinflusst, dass die interne Zustandsgröße klassischerweise mindestens doppelt so groß wie die gewünschte Sicherheitsstufe gewählt wurde (also bspw. ein interner Zustand der Größe 160 Bit bei angestrebtem 80-Bit-Sicherheitsniveau). Solche klassischen Chiffren stehen jedoch im fundamentalen Gegensatz zum Designprinzip moderner leichtgewichtiger Stromchiffren, da ein größerer innerer Zustand zwangsläufig den Ressourcenbedarf erhöht. Klassische Stromchiffren, die einen großen internen Zustand nutzen, sind beispielsweise die eSTREAM-Portfolio-Mitglieder[2] Grain [HJM06] und Trivium [CP05].

4.2.3 Neue Ansätze zur Entwicklung leichtgewichtiger Stromchiffren

In jüngster Zeit wurden Anstrengungen unternommen, die Größe des internen Zustands zu reduzieren und gleichzeitig ein angemessenes Sicherheitsniveau beizubehalten. Die 2017 vorgestellte Stromchiffre LIZARD [HKM17b] erhöht die Sicherheit gegen sogenannte Key-Recovery-Angriffe über die klassische Grenze des Geburtstagsparadoxons hinaus und erreicht ein Sicherheitsniveau von $2n/3$, wobei n die Größe des internen Zustands bezeichnet (vgl. [HK18] für einen entsprechenden Sicherheitsbeweis der zugrundeliegenden LIZARD-Construction). Dies wird durch bitweise Addition des geheimen Schlüssels auf den internen Zustand im letzten Schritt der Initialisierungsphase der Chiffre erreicht. Die Sicherheit gegen sogenannte Unterscheidungsangriffe, bei denen ein Angreifer einen durch die betreffende Chiffre generierten Schlüsselstrom von einem echt zufällig generierten Schlüsselstrom unterscheiden will, bleibt jedoch auch bei LIZARD auf dem durch das Geburtstagsparadoxon bestimmten Niveau $n/2$.

[2] *https://www.ecrypt.eu.org/stream/*

Zusätzlich zu dem volatilen internen Zustand, der durch die Feedback-Schieberegister klassischer Stromchiffren realisiert wird, verwenden die seit 2015 vorgestellten Stromchiffren Sprout [AM15] und Plantlet [MAM16] im Rahmen ihrer Zustandsaktualisierung kontinuierlich den im nichtvolatilen Speicher (bspw. einem EEPROM) abgelegten geheimen Schlüssel. Die diesem Designansatz zugrundeliegende Hoffnung bestand darin, dass die zusätzliche kontinuierliche Einbeziehung der Schlüsselbits die Sicherheit über die Grenze des Geburtstagsparadoxons hinsichtlich der volatilen internen Zustandsbits hinaus erhöhen würde. Allerdings wurden diese Chiffre-Designs nicht durch entsprechende Sicherheitsbeweise begleitet und schließlich erfolgreich angegriffen [HKMZ18]. Auch die im Jahr 2021 vorgestellte Stromchiffre Atom [BCI+21] verwendet den geheimen Schlüssel kontinuierlich. Jedoch bietet auch diese keinen über die klassische Grenze des Geburtstagsparadoxons hinausgehenden Schutz gegen Unterscheidungsangriffe, da auch hier der in [HKMZ18] vorgestellte generische Angriff greift.

Ein dritter Vorschlag wurde in [HKM17a] gemacht. Anstatt den nichtvolatilen geheimen Schlüssel kontinuierlich zu verwenden, wird stattdessen der nichtvolatile öffentliche Initialisierungsvektor (Initial Value, IV) zur kontinuierlichen Zustandsaktualisierung verwendet. Ein entsprechender Sicherheitsbeweis wurde in [HKM19] veröffentlicht.

4.2.4 DRACO

Auf der FSE 2023 in Kobe, Japan, stellten die ERNW Research GmbH (Matthias Hamann), die Universität Mannheim (Alexander Moch, Matthias Krause) und die Universität Siegen (Vasily Mikhalev) ein neues Stromchiffre-Design namens DRACO [HMKM22] vor. Die Schlüsselgröße der Chiffre beträgt 128 Bit und die IV-Größe beträgt 96 Bit. DRACO verwendet einen volatilen 128-Bit-Zustand und einen nichtvolatilen 128-Bit-Zustand. Der volatile Zustand wird durch zwei nichtlineare rückgekoppelte Schieberegister (Nonlinear Feedback Shift Registers, NFSRs) mit einer Gesamtgröße von 128 Bit realisiert. Der nichtvolatile Zustand besteht aus dem öffentlichen 96-Bit-IV und einem 32-Bit-Präfix des geheimen Schlüssels. Abbildung 3 zeigt DRACO in der Phase der Schlüsselstromerzeugung.

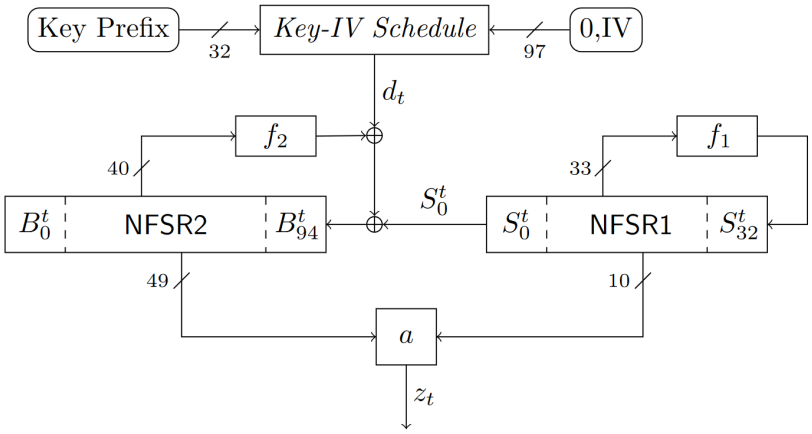

Abbildung 3: DRACO in der Phase der Schlüsselstromerzeugung.

Für das zugrundeliegende neue generische Schema stellen wir in [HMKM22] eine Sicherheitsanalyse im Random-Oracle-Modell bereit und beweisen volle n-Bit-Sicherheit gegen generische TMD-TO-Angriffe für einen volatilen n-Bit-Zustand. Im Fall von DRACO bedeutet dies konkret, dass jeder generische TMD-TO-Unterscheidungsangriff (und damit auch jeder generische TMD-TO-Key-Recovery-Angriff) gegen die Chiffre eine Zeitkomplexität von mindestens 2^{128} Schritten aufweist. Unserer Kenntnis nach ist DRACO damit die erste leichtgewichtige Stromchiffre, die dies mit lediglich 128 volatilen und 128 nichtvolatilen Zustandsbits erreicht.

Die Hauptvariante von DRACO speichert das Schlüsselpräfix und den IV extern. In einem ultraleichtgewichtigen Szenario wie einem kostengünstigen RFID-Tag, bei dem der geheime Schlüssel in die anwendungsspezifische integrierte Schaltung (ASIC) 'eingebrannt' oder in einem EEPROM gespeichert ist und bei dem der Frame Counter als IV verwendet wird, benötigt DRACO 23 Prozent weniger Chip Area und 31 Prozent weniger Power als die etablierte Stromchiffre Grain-128a [ÅHJM11] bei 10 MHz. Die drastische Einsparung an Power resultiert aus der geringeren Area, aber insbesondere auch aus der Tatsache, dass im Gegensatz zu früheren Chiffren wie Grain-128a nur die Hälfte der Zustandsbits ständig aktualisiert wird, was die kostspielige, sogenannte Dynamic Power Consumption deutlich reduziert.

4.2.5 Aktuelle Weiterentwicklungen: DRACO v2 and DRACO-PQ

Da es sich um die erste Chiffre-Instanziierung für das neu eingeführte Designparadigma handelt, muss DRACO offensichtlich einer strengen und umfassenden Kryptoanalyse durch die Community (und möglicherweise daraus resultierenden Änderungen) unterzogen werden. Seit der Veröffentlichung der initialen DRACO-Spezifikation im Juni 2022 (die FSE-Konferenz bündelt klassischerweise die Veröffentlichungen der vier Ausgaben von *IACR Transactions on Symmetric Cryptology (ToSC)* aus dem Vorjahr) wurde bisher lediglich ein Schwachpunkt identifiziert. In [Ban22] wird ein Unterscheidungsangriff vorgestellt, der es ermöglicht, einen von DRACO generierten Schlüsselstrom von einem Strom echt zufälliger Bits mit einer Zeit- und Speicherkomplexität von jeweils 2^{107} auf der Grundlage von 2^{40} durch den Angreifer gewählten IVs zu unterscheiden. Auch wenn man dies noch als tolerierbar ansehen könnte (wie zuvor erwähnt, haben die Designer des 'konkurrierenden' Verschlüsselungsverfahrens Atom [BCI+21] bereits im entsprechenden Designdokument anerkannt, dass für ihre Chiffre sogar Unterscheidungsangriffe mit einer Komplexität von 'nur' 2^{80} möglich sind), möchten wir das gesamte Potenzial des neuen Continuous-Key-IV-Designprinzips ausschöpfen und eine volle 128-Bit-Sicherheit gegen Schlüsselwiederherstellungs- und Unterscheidungsangriffe erreichen. Daher haben wir bereits auf der FSE 2023 entsprechende Vorschläge für einen modifizierten Key-IV-Schedule vorgestellt [HMKM23].

Eine aktualisierte Variante von DRACO ist bereits in die Monitoring-Plattform des UNCOVER-Projekts integriert. Diese weist eine etwas schlechtere Hardwareeffizienz als die initiale Version von DRACO auf, bietet jedoch noch immer deutliche Vorteile gegenüber den zuvor erwähnten Stromchiffren Atom und Grain-128a (vgl. [HMKM23]). Zudem läuft aktuell eine Zusammenarbeit mit dem Autor des besagten Unterscheidungsangriffs aus [Ban22], im Rahmen derer der Key-Schedule von DRACO gegenüber den neuen Vorschlägen aus [HMKM23] nochmals überarbeitet wird. Hier existiert auch bereits eine neue Version, welche als DRACO v2 veröffentlicht werden soll und nach aktuellem Stand mindes-

tens wieder die Hardwareeffizienz der initialen DRACO-Version besitzen wird, bei gleichzeitiger Resistenz gegen den Unterscheidungsangriff aus [Ban22].

Zusätzlich wurde die Verschlüsselung per DRACO im Hinblick auf die Anforderungen der Monitoring-Plattform um einen Mechanismus zur Nachrichtenauthentifizierung ergänzt. Diese sogenannte Authenticated Encryption (AE) gewährleistet nicht nur die Vertraulichkeit der betreffenden Daten, sondern ermöglicht bei Kenntnis des symmetrischen Schlüssels auch, deren Authentizität und Integrität zu verifizieren. Die entsprechenden Einsatzszenarien im Rahmen der Monitoring-Plattform sind vielfältig und erstrecken sich, wie in der Kapiteleinleitung beschrieben, beispielsweise auf die sichere Speicherung und Übertragung aufgezeichneter Daten (zwischen Monitoring-Plattform und Backend bzw. im Fahrzeug verteilten Sensoren) sowie auf die Absicherung des Deployment-Prozesses. Die Verifikation von Authentizität und Integrität einer entschlüsselten Nachricht erfolgt mittels eines Authentication Tags der Länge 64 Bit, welches zuvor im Rahmen des Verschlüsselungsvorgangs auf Basis der Klartext-Nachricht und jedes zweiten Schlüsselstrombits erzeugt wurde. Das konkrete Schema zur Erzeugung dieses Authentication Tags entspricht jenem von Grain-128AEADv2 [HJM+2021], einem Finalisten des *NIST Lightweight Cryptography Standardization Process*[3].

Neben dem unmittelbaren Einsatz der neuen Chiffre im Rahmen von UNCOVER hat die Entwicklung von DRACO zudem eine Reihe internationaler Forschungskooperation zwischen der ERNW Research GmbH und verschiedenen Partnern hervorgebracht, welche in Kapitel 5.1 detaillierter beschrieben werden.

4.3 Sicherheitsprüfung der Monitoring-Plattform

Für die IT-sicherheitstechnische Prüfung der Monitoring-Plattform wurde eine Kombination der im Folgenden vorgestellten Ansätze als geeignet identifiziert.

4.3.1 Penetration-Tests

Grundlage eines solchen Penetration-Tests ist die Erfassung aller interner und externer Schnittstellen (CAN etc.) des Geräts. Dabei muss initial definiert werden, wie invasiv die Prüfung aufgebaut sein soll; bspw. kann der Fokus nur auf frei erreichbaren Schnittstellen liegen oder auch invasive Methoden wie das irreversible Öffnen des Gehäuses oder gar anlöten zusätzlicher Teile umfassen. Auf Basis dieser Informationen wird anschließend ein grundlegender Hardening Check durchgeführt. Dieser umfasst typischerweise das Feststellen des Status (deaktiviert, erreichbar etc.) kritischer Schnittstellen wie ISP/ICP, SWIM, JTAG, UART und sonstiger Debugging-Schnittstellen. Auch eine Prüfung von UDS-Diagnoseschnittstellen sowie vergleichbare Prüfungen fallen, sofern anwendbar, üblicherweise in diesen Bereich eines entsprechenden Penetration-Tests. Im Rahmen von UNCOVER stand ausschließlich der Demonstrator für entsprechende Untersuchungen zur Verfügung. In seiner gleichzeitigen Rolle als Entwicklungsplattform basiert dieser auf einem *AMD Zynq UltraScale+ MPSoC ZCU102 Evaluation Kit*, dessen Schnittstellen im entsprechenden Gehäuse

[3] *https://csrc.nist.gov/projects/lightweight-cryptography*

leicht/frei zugänglich sind (vgl. Abbildung 4 in Kapitel 4.3.1.2). Insbesondere liegt es auch gerade in der Natur eines solchen Entwicklungsboards, möglichst viele Schnittstellen bereitzustellen. Diese Eigenschaften des Demonstrators lassen jedoch keinerlei Rückschlüsse auf mögliche Hardening-Probleme zukünftiger, praktischer Instanziierungen der Monitoring-Plattform zu. Entsprechend wurde der Fokus der Penetration-Tests im Rahmen von UNCOVER vielmehr auf den Punkt Vulnerability Research / Schwachstellenanalyse gelegt, welcher in den nachfolgenden Kapiteln 4.3.1.1 und 4.3.1.2 detaillierter beschrieben wird.

4.3.1.1 Kommunikationsanalyse

Bei der Kommunikationsanalyse wird versucht, durch Spoofing oder einen Man-in-the-Middle Angriff in den Kommunikationskanal einzudringen und damit die Vertraulichkeit und Integrität der Daten, die über diesen Kanal ausgetauscht werden, anzugreifen. Wesentlich bei diesem Test ist die Überprüfung der Authentizität der beiden Kommunikationspartner bzw. der zwischen ihnen übertragenen Daten. Üblicher weiterer Punkt im Rahmen einer Kommunikationsanalyse ist auch die Überprüfung auf Anfälligkeit gegenüber Angriffen durch Replay. Während des Betriebs des UNCOVER-Demonstrators (sprich, nach dessen Programmierung per UART-Schnittstelle) erfolgt die Informationsübertragung unidirektional von der Monitoring-Plattform hin zum Backend. Konkret werden die von der Monitoring-Plattform aufgezeichneten CAN-Daten zunächst lokal gespeichert und anschließend, ausgelöst durch Drücken der am Demonstrator-Gehäuse angebrachten "SEND"-Taste, per Debug-Schnittstelle an das Backend übertragen. Wie in Kapitel 4.2 beschrieben, werden Vertraulichkeit, Authentizität und Integrität der betreffenden CAN-Daten bereits im Speicher der Monitoring-Plattform per Authenticated Encrpytion geschützt. Die ins Backend übertragenen Daten bestehen entsprechend aus 3-Tupeln der Form (IV, Chiffretext, Authentication Tag). Dabei kann mittels der IV, welche im Rahmen der Verschlüsselung grundsätzlich niemals mehrfach unter demselben Schlüssel verwendet werden darf, im Backend auch leicht eine Überprüfung auf mögliche Replay-Angriffe erfolgen.

4.3.1.2 Fuzzing

Um die Robustheit einer Implementierungen zu testen, wird häufig Fuzzing angewandt. Während des Fuzzings werden typischerweise automatisiert bekannte Angriffsmuster, randomisierte Daten oder numerische Grenzwerte in Protokollfelder eingetragen und gesendet. Dabei wird die Reaktion des Gerätes auf jede Nachricht bzw. Nachrichtenfolge beobachtet und dokumentiert. Relevante Reaktionen können Abstürze, undefiniertes Verhalten, schwankende Leistungsaufnahme oder Temperaturveränderungen sein.

Im Folgenden werden das entsprechende Vorgehen bei der Prüfung des UNCOVER-Demonstrators und exemplarisch einige der Fuzzing-Ergebnisse samt deren Einfluss auf die Entwicklung der Monitoring-Plattform beschrieben.

UNCOVER Demonstrator Setup

Wie zuvor erwähnt, wurde im Rahmen des UNCOVER-Demonstrators die gleiche Hard- und Software verwendet, welche auch bei der Entwicklung der Monitoring-Plattform zum Einsatz kam. Zentraler Bestandteil ist dabei ein *AMD Zynq UltraScale+ MPSoC ZCU102 Evaluation Kit*[4]. Drei Schnittstellen dieses Entwicklungsboards sind über entsprechende Anschlüsse am Demonstrator-Gehäuse erreichbar (vgl. Abbildung 4):

- der Anschluss "UART" (eine UART-Schnittstelle), über welchen das initiale Deployment (sprich, die Programmierung des Microblaze Softcore-Prozessors des Boards) erfolgt;
- der Anschluss "CAN" (eine CAN-Schnittstelle), über welchen die Monitoring-Plattform mit dem CAN-Bus des Demonstrator-Fahrzeugs verbunden wird, um die dort übertragenen Nachrichtenframes zu überwachen;
- der Anschluss "DEBUG" (eine UART-Schnittstelle), über welchen die Datenübertragung an das Backend erfolgt.

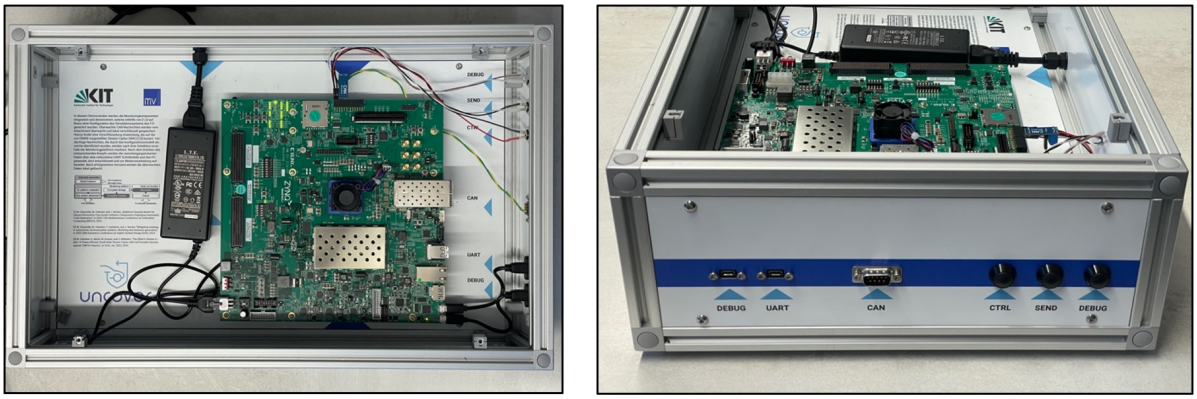

Abbildung 4: Demonstrator der UNCOVER Monitoring-Plattform.

Steuern lässt sich der Demonstrator über drei am Gehäuse angebrachte Tasten: "SEND", "DEBUG" und "CTRL". Per "SEND"-Taste können die gespeicherten Nachrichten über den mit "DEBUG" bezeichneten Anschluss ausgelesen werden. Per "CTRL"-Taste kann die Verschlüsselung zu Demonstrations- und Testzwecken ein- bzw. ausgeschaltet werden. Per "DEBUG"-Taste kann der Debug-Modus aktiviert bzw. deaktiviert werden. Bei aktiviertem Debug-Modus werden alle akzeptierten ankommenden Nachrichten mitgeschrieben. Im normalen Betrieb hingegen wartet die Monitoring-Plattform regelbasiert auf ein festgelegtes Ereignis, bevor sie mit der Aufzeichnung von Nachrichten beginnt. Außerdem verfügt der Demonstrator über mehrere Status-LEDs. Die beiden linken LEDs leuchten, wenn die Ausgabe verschlüsselt erfolgt. Rechts daneben blinken zwei LEDs immer dann auf, wenn mehrere Nachrichtenframes zu einer Nachricht kombiniert werden konnten und in den Speicher geschrieben werden. Die verbleibenden drei LEDs zeigen an, ob sich der Demonstrator im Debug-Modus befindet.

[4] *https://www.xilinx.com/products/boards-and-kits/ek-u1-zcu102-g.html*

Im Rahmen der Tests der Monitoring-Plattform während der Entwicklung und auch im Zuge des hier beschriebenen CAN-Bus-Fuzzings kam die Software *Busmaster* zum Einsatz, um entsprechende Nachrichten auf den CAN-Bus des Demonstrator-Fahrzeugs (sprich, an die CAN-Schnittstelle der Monitoring-Plattform) zu übertragen. Busmaster verfügt unter anderem über ein Replay-Feature, mittels dessen ein zuvor aufgezeichnetes Transkript von Nachrichtenframes erneut gesendet werden kann. Die entsprechenden Log-Dateien lassen sich aufgrund ihres Textformats leicht bearbeiten, sowohl manuell als auch skriptgesteuert. In den im Rahmen von AP 3 durchgeführten Fuzzing-Tests wurde diese Möglichkeit genutzt, um eigene/modifizierte Nachrichtenframe-Abläufe zu generieren. Abbildung 5 zeigt exemplarische die Busmaster Log-Datei einer legitimen Nachrichtenfolge auf dem CAN-Bus des UNCOVER-Demonstrators.

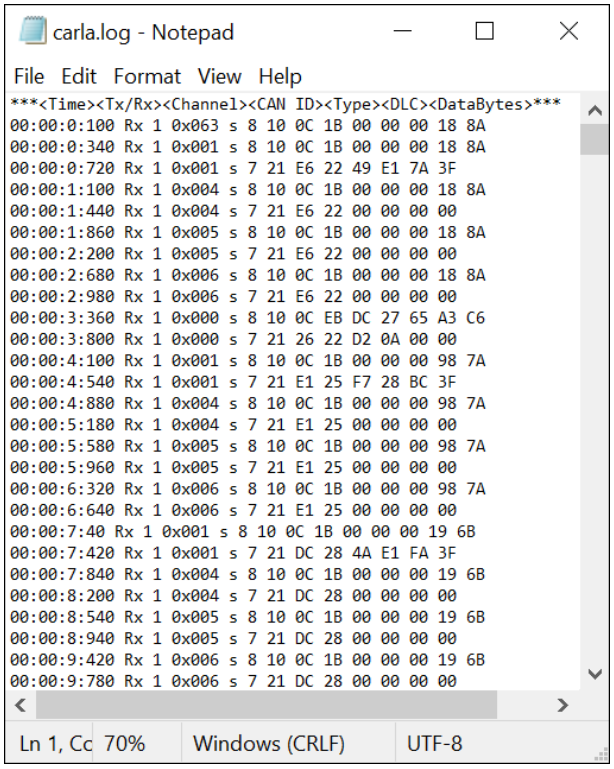

Abbildung 5: Beispielhafte Busmaster Log-Datei.

Fuzzing im Allgemeinen

Fuzzing bezeichnet eine automatisierbare Technik im Penetration-Testing (Pentesting), welche zur Aufdeckung von Programmfehlern eingesetzt wird. Entsprechende Fehler können von Angreifern potenziell ausgenutzt werden, um die Funktionsweise eines Systems zu stören (*Degradation of Service Attack*), ein System vollständig zum Absturz zu bringen (*Denial of Service Attack*), Daten zu stehlen oder sogar Kontrolle über den Programmablauf zu erlangen.

Aufgabe des Pentestings ist daher die Aufdeckung jeglich gearteter Programmfehler, die als Angriffspunkte genutzt werden könnten. Dabei zielen viele Strategien darauf ab, das System gezielt und systematisch auf Schwachstellen zu überprüfen. Entsprechende Findings können hier bspw. der Einsatz veralteter, bekannt verwundbarer externer Bibliotheken oder auch die Verwendung unsicherer Konfigurationen (bspw. hinsichtlich kryptographischer Verfahren) sein. Auch die in Kapitel 4.3.2 beschriebenen Source Code Audits können wichtiger Teil entsprechender Pentests sein (insb. im Falle sogenannter *White Box Pentests*). Auch wenn hierdurch häufig bereits viele Fehler gefunden werden, führen verschiedene Faktoren (wie bspw. die Komplexität des untersuchten Systems oder eine eingeschränkte Informationsbasis des Pentesters im Rahmen sogenannter *Grey/Black Box Pentests*) naheliegenderweise dazu, dass andere Fehler zunächst weiterhin unentdeckt bleiben.

Fuzzing ist eine Strategie, welche insbesondere auf die Entdeckung genau solcher, bisher unentdeckt gebliebener Schwachstellen abzielt. Die Besonderheit in der Vorgehensweise ist dabei, dass das Testsystem mit einer enormen Menge zufällig generierter Eingabewerte betrieben wird. Dabei wird das System als eine Black Box betrachtet, deren akzeptierte Eingabeformate zwar bekannt sind, jedoch beim Test nicht immer eingehalten werden müssen. Somit werden sehr viele Eingaben getestet, die bei einem systematischen Test nicht enthalten wären, und eine entsprechend breite Testabdeckung erreicht. Löst eine bestimmte zufällige Eingabe einen Programmfehler aus, findet anschließend eine gezielte Suche nach der Fehlerursache statt. Diese Suche kann potenziell sehr aufwendig sein, da im Rahmen der Analyse zunächst nur die betreffende problematische Eingabe und das beim System ausgelöste fehlerhafte Verhalten als Informationsbasis zur Verfügung stehen.

Fuzzing der Monitoring-Plattform des UNCOVER-Demonstrators

Ziel des Fuzzing-Tests des UNCOVER-Demonstrators war es, mögliches Fehlverhalten der Monitoring-Plattform als Reaktion auf über den CAN-Bus erhaltene Nachrichten zu erkennen. Über diesen könnten im späteren Praxiseinsatz der Monitoring-Plattform durch einen Angreifer bereits kompromittierte, mit dem CAN-Bus verbundene Fahrzeugkomponenten versuchen, die Monitoring-Plattform mittels entsprechender CAN-Nachrichten zum Absturz oder schlimmstenfalls sogar ebenfalls unter die Kontrolle des Angreifers zu bringen.

Entsprechend wurde als wichtiger Teil der im Rahmen von AP 3 durchgeführten Tests eine große Anzahl 'zufälliger' CAN-Nachrichtenframes generiert und diese dann auf den CAN-Bus-Anschluss des Demonstrator geschickt. Wie bereits allgemein beschrieben, zielte diese Vorgehensweise zunächst darauf ab, ein fehlerhaftes Verhalten oder einen vollständigen Systemausfall der Monitoring-Plattform im Rahmen der Nachrichtenverarbeitung herbeizuführen. Im 'Erfolgsfall' fand anschließend in enger Zusammenarbeit mit dem für die Entwicklung der Monitoring-Plattform zuständigen Projektpartner KIT eine entsprechende Suche nach dem zugrundeliegenden Programmfehler statt.

Um eine breite Abdeckung hinsichtlich der Eingabedaten zu erzielen, wurde eine Hälfte der Testdaten rein zufällig generiert. Dabei variierten folgende Parameter:

- Länge des Nachrichtenframes,
- Inhalt des Nachrichtenframes,
- CAN-ID des Nachrichtenframes.

Neben der Verwendung rein zufälliger Eingabedaten zur Erzielung einer möglichst breiten Abdeckung gilt es im Rahmen eines Fuzzing-Tests jedoch auch, das betreffende System zusätzlich in einem realitätsnahen Kontext auf seine Fehlerbeständigkeit zu prüfen. Daher wurde die andere Hälfte der Testdaten auf Grundlage 'gutartiger' CAN-Nachrichtenfolgen generiert, welche zuvor durch das Simulationssystem CARLA für ein legitimes Fahrszenario erzeugt worden waren. Die verwendete Simulation enthielt dabei ca. 16.600 Nachrichtenframes. Ein für das UNCOVER-Projekt entwickeltes Fuzzing-Skript generierte aus diesem legitimen Transkript dann eine Vielzahl von Varianten mit jeweils 1, 5, 15, 100, 10.000 oder 100.000 Veränderungen. Jede dieser Veränderungen wurde dabei wiederum zufällig mittels einer der folgenden Operationen erzeugt:

- Duplizieren eines zufällig ausgewählten Nachrichtenframes,
- Löschen eines zufällig ausgewählten Nachrichtenframes,
- Vertauschung zweier zufällig ausgewählter Nachrichtenframes,
- Ergänzung eines zufällig erstellten Nachrichtenframes an einer zufälligen Position,
- Veränderung eines zufällig ausgewählten Daten-Bytes eines zufällig ausgewählten Nachrichtenframes,
- Verlängerung eines zufällig ausgewählten Nachrichtenframes durch das Einfügen eines zufälligen Daten-Bytes an einer zufälligen Position,
- Löschung (und somit Nachrichtenverkürzung) eines zufälligen Daten-Bytes eines zufällig ausgewählten Nachrichtenframes,
- Änderung der CAN-ID eines zufällig ausgewählten Nachrichtenframes auf einen zufälligen Wert.

Insgesamt belief sich das Volumen der getesteten Nachrichtenabläufe auf eine Summe von mehr als 7.000.000 Nachrichtenframes. Abbildung 6 zeigt den geschilderten Test- und Diagnosezyklus übersichtsartig anhand einer im Rahmen der Fuzzing-Tests entdecken CAN-Nachrichtenfolge, welche zu einem entsprechenden Fehlverhalten der Monitoring-Plattform des Demonstrators führte.

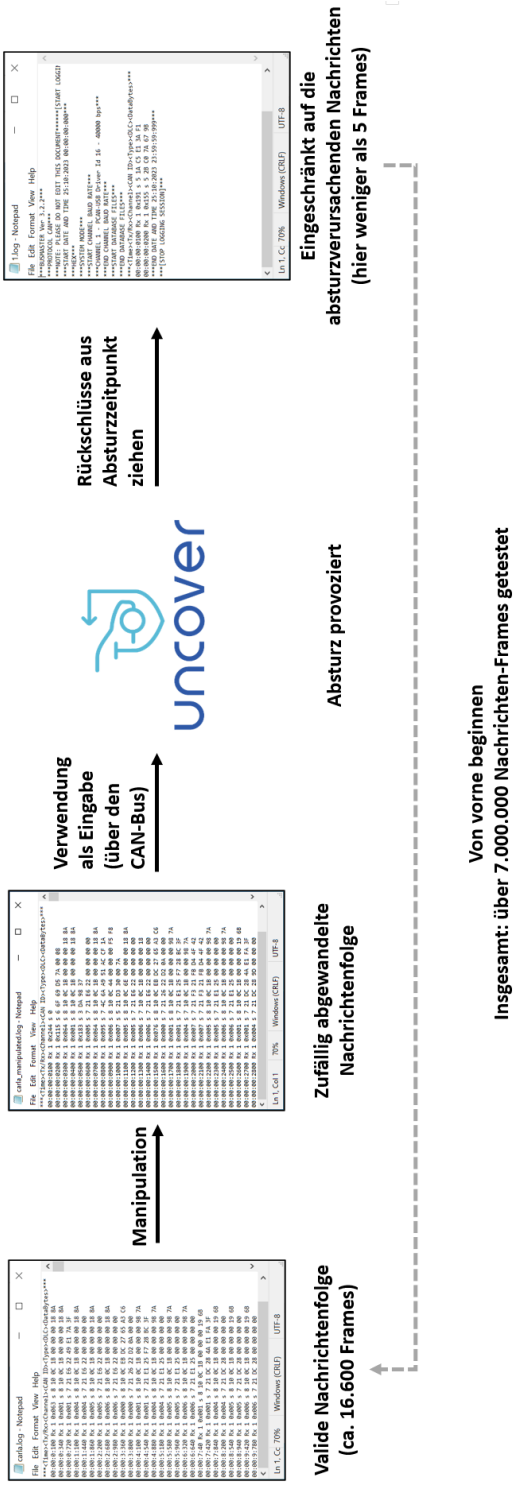

Abbildung 6: Fuzzing-Analyse der Monitoring-Plattform.

Mittels der Fuzzing-Tests konnten zwei verschiedene Arten von Fehlverhalten festgestellt werden:

1. Eine der gefundenen CAN-Nachrichtenkombinationen führte zu einem sogenannten "Degradation of Service"-Fehler. Dabei blieb die Monitoring-Plattform zwar grundsätzlich ansprechbar, erfüllte ihre eigentliche Funktion jedoch nicht mehr korrekt. Genauer führte die betreffende, fehlerverursachende Nachrichtenfolgen (Block (1) in Listing 1) dazu, dass ab diesem Zeitpunkt sämtliche weiteren (auch legitimen) CAN-Nachrichten fehlerhaft in den Speicher der Monitoring-Plattform geschrieben wurden und der Demonstrator nur noch den Wert "00000000FFFFFFFF0000000000000000" (bzw. Abwandlungen davon in unterschiedlichen Nachrichtenlängen) zurückgab. Tasten, LED-Anzeigen und Debugging-Ausgaben des Demonstrators funktionierten zwar weiterhin, die eigentliche Funktion der Monitoring-Plattform ließ sich nach dem beschriebenen Fehler jedoch nur durch einen vollständigen Neustart des Boards wiederherstellen. Abbildung 7 zeigt die entsprechende Debugging-Ausgabe der Monitoring-Plattform während des Fuzzing-Tests.

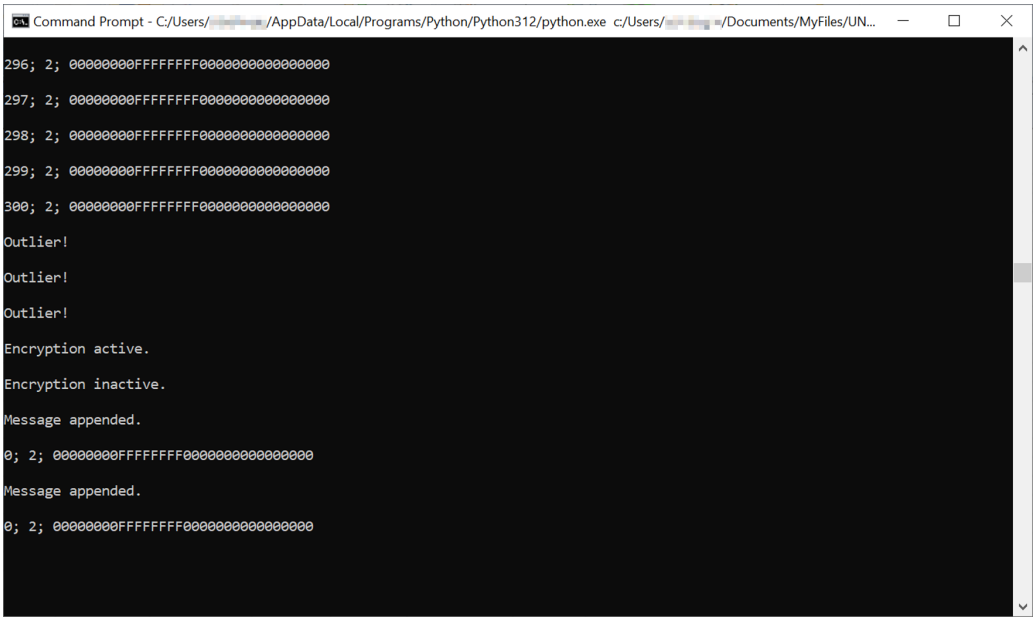

Abbildung 7: "Degradation of Service"-Finding der Fuzzing-Tests.

2. Das zweite im Rahmen der Fuzzing-Tests beobachtete Fehlverhalten war vom Typ "Denial of Service", sprich, die auslösende CAN-Nachrichtenkombination führte zu einem vollständigen Absturz der Monitoring-Plattform. Konkret verblieben fortan alle Status-LEDs des Demonstrators in ihrem bestehenden Zustand, das Drücken von Tasten am Demonstrator-Gehäuse hatte keinerlei Effekt mehr (weder hinsichtlich Aktivieren/Deaktivieren der Verschlüsselung, noch bzgl. Auslesen des Nachrichtenspeichers) und es fanden keine Ausgaben über den "DEBUG"-Anschluss des Demonstrators mehr statt. Auch dieser Fehlerzustand konnte nur durch einen vollständigen Neustart der Monitoring-Plattform behoben werden.

Nach der Beobachtung der Symptomatiken mussten jeweils die spezifischen Nachrichtenframes ermittelt werden, welche das Fehlverhalten verursachten. Die Realisierung einer automatisierten Erkennung des exakten 'Zeitpunkts' des Fehlerauftretens (in Bezug auf die problematischen Kombinationen von CAN-Nachrichtenframes) gestaltete sich im beschriebenen Demonstrator-Setup als unverhältnismäßig aufwendig. Stattdessen wurden die fehlerverursachenden CAN-Transkripte im Rahmen eines manuellen Suchprozesses sukzessive eingeschränkt. Dabei wurden iterativ immer wieder Blöcke von Nachrichtenframes aus den entsprechenden Busmaster Log-Dateien entfernt und anschließend überprüft, ob die verbliebenen Frames das jeweilige Fehlverhalten weiterhin verursachten. Listing 1 zeigt die resultierenden minimalen CAN-Nachrichtenfolgen zur Auslösung der beschriebenen Fehlerszenarien (1) und (2).

```
***<Time><Tx/Rx><Channel><CANID><Type><DLC><DataBytes>***

(1)
00:00:00:0100 Rx 1 0x191 s 5 1A C5 E1 3A F1
00:00:00:0200 Rx 1 0x155 s 5 2B C0 7A 67 9B

(2)
00:00:00:0100 Rx 1 0x01a s 8 10 1C EC DC 27 65 06 69
00:00:00:0200 Rx 1 0x085 s 8 23 00 00 00 60 30 62 3E
00:00:00:0300 Rx 1 0x006 s 8 10 0C 25 00 00 00 32 1E
00:00:00:0400 Rx 1 0x006 s 7 21 C3 37 A9 9E 64 36
00:00:00:0500 Rx 1 0x007 s 8 10 0C FO DC 27 65 2B 09
```

Listing 1: Fehlerverursachende CAN-Nachrichtenfolgen aus den Fuzzing-Tests.

Auf Basis dieser Informationen konnte beide beobachteten Probleme schließlich auf eine fehlende Validitätsprüfung hinsichtlich der CAN-IDs zurückgeführt werden. Konkret beschreibt der für die Entwicklung der Monitoring-Plattform zuständige Projektpartner KIT die Hintergründe der per Fuzzing aufgedeckten Schwachstelle wie folgt:

> "Beide Ausfälle waren [...] Symptome der gleichen Schwachstelle. Das Zusammensetzen einer gültigen Nachricht aus mehreren CAN-Multiframe-Nachrichten wird durch das Puffern von Teilnachrichten in Nachrichten-spezifischen Puffern implementiert. Diese sind in einer Builder-Klasse gekapselt. Jeder CAN-ID ist dabei eine eigene Builder-Instanz zugeordnet. Für die insgesamt 10 überwachten CAN-IDs sind insgesamt 10 Builder-Instanzen vorhanden.
>
> Das Befüllen der ID-spezifischen Puffer durch die Builder-Instanzen erfolgt dabei unter der Überprüfung der erwarteten CAN-Multiframe-ID und der zuvor übermittelten erwarteten Gesamtlänge. Ist die angekündigte Gesamtlänge von der zuvor spezifizierten Gesamtlänge verschieden, oder ein empfangener Frame trägt den falschen Index, wird die bisher gepufferte Nachricht verworfen und der Puffer gelöscht.
>
> Die Builder-Klassen sind dabei in einem C-Array angeordnet und die Indizierung dieses Arrays erfolgt über die Empfangene CAN-ID. Einem Angreifer (und dem Fuzzing-Algorithmus) war es so möglich durch

> das Versenden einer Nachricht mit einer ID größer 10 auf Speicherbereiche hinter den Adressen der Builder-Klassen zuzugreifen. Zusammen mit dem Offset der Funktions-Pointer innerhalb dieser Instanzen wurde so Code ausgeführt, der nicht zu der eigentlichen Funktionalität gehörte. Dabei ist es irrelevant, ob es sich ursprünglich um Daten oder Funktionen handelte. Die Vermutung liegt nahe, dass [im "Degradation of Service"-Szenario] entweder die Zieladresse der Nachrichten-spezifischen Puffer, die Zieladresse des gesamten Ringpuffers oder die Adresse der Ausgabefunktion umgeschrieben wurde. Im ["Denial of Service"-] Fall liegt der Verdacht nahe, den Program-Counter an eine Stelle springen zu lassen, die eine NOP-Loop zur Folge hat.
>
> Beide Probleme konnten durch das Einführen einer Begrenzungsüberprüfung behoben werden. Nachrichten außerhalb der Operational Design Domain werden nun getrennt behandelt und lösen das Aufzeichnen aus."

Die Ursachenbeschreibung des Projektpartners KIT zeigt, dass den im Rahmen des Fuzzings aufgedeckten "Degradation of Service"- bzw. "Denial of Service"-Szenarien eine Schwachstelle mit potenziell gravierenden Implikationen zugrunde lag. Denn die fehlende Überprüfung der IDs der erhaltenen CAN-Nachrichten durch die Monitoring-Plattform ist an der betreffenden Code-Stelle gleichbedeutend mit einer fehlenden Überprüfung eines zum Zugriff auf ein C-Array verwendeten Indexes. Ein Angreifer, der (bspw. aufgrund einer zuvor kompromittierten, anderen Fahrzeugkomponente) bereits Zugriff auf den CAN-Bus erlangt hat, könnte versuchen, diese klassische "Out of Bounds"-Schwachstelle wie folgt zur Erlangung von Code Execution auf der Monitoring-Plattform ausnutzen. Durch Versenden von CAN-Nachrichten mit passenden CAN-IDs ist ihm aufgrund der fehlenden Indexüberprüfung der Zugriff auf Speicherbereiche außerhalb des besagten C-Arrays möglich. Dies wiederum kann es ihm, wie in obigem Zitat des Projektpartners KIT beschrieben, ggf. erlauben, Funktionspointer zu manipulieren und dadurch den Programmfluss der Monitoring-Plattform unter seine Kontrolle zu bringen. Schlimmstenfalls wäre es ihm damit nun möglich, eigenen Code anzuspringen und zur Ausführung zu bringen, den er zuvor bspw. als 'Daten' anderer CAN-Nachrichten, welche durch die Monitoring-Plattform mitgeschrieben wurden, in deren Speicher platziert hat.

Nach Behebung der aufgedeckten Schwachstellen wurde die Monitoring-Plattform des Demonstrators erneut entsprechenden Fuzzing-Tests unterzogen. Im Rahmen dieser wurden keine Auffälligkeiten mehr festgestellt.

4.3.2 Source Code Audit

Als Teil der Sicherheitsüberprüfung der Monitoring-Plattform wurde deren Quellcode (vorwiegend C/C++ und Verilog) mit statischen Code-Analyse-Werkzeugen und durch die ERNW Research GmbH entwickelten Skripten untersucht. Zudem fand eine manuelles Review des Quellcodes statt. Dabei standen neben klassischen Schwachstellen auch die Umsetzung von Security Best Practices im Fokus der Untersuchung. Im Folgenden werden die einzelnen Schritte der Untersuchung näher erläutert.

4.3.2.1 Scan des Codes

Mit Hilfe dedizierter Scan Tools wurde eine Untersuchung des Quellcodes der Plattform durchgeführt. Hierbei wurde geprüft, ob die Komponenten entsprechend Best Common Practices der Programmiersprache Angriffe wie Buffer Overflows, Format-String-Fehler, unsichere Datenablage, unsichere Kommunikation etc. sinnvoll verhindern können. Die Ergebnisse wurden manuell verifiziert.

4.3.2.2 Manuelle Untersuchung des Codes

Die manuelle Überprüfung des Quellcodes erfolgte mit Hilfe von Code-Navigationswerkzeugen, eigenen Skripten sowie basierend auf Security-Code-Review-Checklisten. Folgende Module sind typische Bestandteil eines manuellen Code Reviews:

- **Bewertung der genutzten Standardkomponenten.** Die im Entwickler-Framework benutzten Standardkomponenten werden bzgl. ihres Sicherheitsniveaus bewertet. Grundlage dafür sind die Dokumentation sowie die Historie bekannter Sicherheitslücken für diese Komponenten.
- **Design und Architektur.** Hier wird geprüft, inwieweit das Design eine Funktionstrennung der Komponenten ermöglicht. Dies ist im Embedded-Bereich nicht immer möglich, jedoch sollte zumindest eine grobe Trennung der Verantwortlichkeiten vorherrschen, um die Komplexität so gering wie möglich zu halten.
- **Sicherer Umgang mit Daten.** Sensible Informationen müssen zu jeder Zeit geschützt werden. Dies beinhaltet die sichere Ablage von vertraulichen Informationen wie beispielsweise kryptographischer Schlüssel außerhalb des Quellcodes. Speziell unter Embedded-Systemen stehen dafür meist sichere Möglichkeiten (hardwarebasierte Speicher (Trustzone) etc.) zur Verfügung – während der Entwicklung muss jedoch darauf geachtet werden, diese auch für entsprechende Daten zu verwenden.
- **Authentifizierung.** Hier werden vorhandene Authentifizierungsmechanismen hinsichtlich ihrer sicherheitsrelevanten Merkmale untersucht. Je nach Implementierung des Features werden verschiedene Testcases angewendet, wobei üblicherweise eine Umgehung der Authentifizierung im Vordergrund steht.

4.3.3 Untersuchung der Update/Deployment-Mechanismen

Bei der Untersuchung von Update/Deployment-Mechanismen werden üblicherweise

- Updates via Netzwerk,
- Updates via Speichermedium,
- Updates via Diagnosegerät

als entsprechende Wege in Betracht gezogen.

Durch unzulängliche Konfiguration kann ein Angreifer den Update-Vorgang missbrauchen, um Schadsoftware auf das System zu laden oder sich Zugang zu verschaffen. Mögliche Prüfaspekte sind hier:

- Evaluation der Transportsicherheit.
- Generelle Evaluation der Security-Maßnahmen wie
  - Code-Signing,
  - Verschlüsselung,
  - Speicherung von Schlüsselmaterial.

Wie in Kapitel 4.2 beschrieben, kommt der für UNCOVER entwickelten, leichtgewichtigen Authenticated-Encryption-Lösung auf Basis der Stromchiffre DRACO eine zentrale Bedeutung im Rahmen des späteren Deployment-Prozesses zu. Während im Rahmen des UNCOVER-Demonstrators Updates der Monitoring-Plattform (bspw. nach Integration neuer Monitoring-Regeln im Backend als Reaktion auf einen erkannten Sicherheitsvorfall) noch per UART-Schnittstelle auf das zur Demonstration verwendete Entwicklungsboard (*AMD Zynq UltraScale+ MPSoC ZCU102 Evaluation Kit*) überspielt werden, würden diese im späteren praktischen Einsatz typischerweise als Over-the-Air-Updates erfolgen. Durch Implementierung der in Kapitel 4.2 detaillierter beschriebenen Authenticated-Encryption-Lösung mittels DRACO als separates FPGA-Modul ist die Monitoring-Plattform bereits jetzt aus kryptographischer Sicht entsprechend vorbereitet, da das besagte FPGA-Modul in späteren, praktischen Instanziierungen der Monitoring-Plattform auch entsprechenden Updater-Modulen direkt zur Entschlüsselung und Verifikation von Aktualisierungen zur Verfügung stünde.

In Abhängigkeit des konkreten Anwendungs-/Umsetzungsszenarios der Monitoring-Plattform kann später auch der Einsatz von auf Public-Key-Kryptographie basierenden Code-Signing/Verification-Lösungen sinnvoll/notwendig sein. Dies wäre bspw. der Fall, wenn ein Update ohne Involvierung des Backends durch eine dritte, ggf. nicht vertrauenswürdige Entität (bspw. eine Werkstatt) an die Monitoring-Plattform übertragen werden soll. Dann könnte der legitime Ersteller des Updates (bspw. der Hersteller der Monitoring-Plattform) dieses mittels seines Private Keys signieren und jede Monitoring-Plattform mittels des zugehörigen, ihr 'ab Werk' bekannten Public Keys vor der Installation eine entsprechende Verifikation durchführen. Nachteil dieser Variante wäre jedoch, dass die Monitoring-Plattform nun neben symmetrischer Verschlüsselung zusätzlich Public-Key-Kryptographie beherrschen müsste und entsprechende Verfahren einen signifikant höheren Ressourcenbedarf als bspw. die im Kontext von UNCOVER entwickelte Authenticated-Encryption-Lösung auf Basis der leichtgewichtigen Stromchiffre DRACO aufweisen. Eine entsprechende Abwägung zwischen Anwendungserfordernissen und Ressourcenverbrauch ist daher für spätere, praktische Instanziierungen der Monitoring-Plattform unumgänglich.

ERNW RESEARCH pursuing knowledge.

5 Verwertung und Veröffentlichungen

In diesem Kapitel werden die verschiedenen Verwertungsperspektiven der Projektergebnisse zusammen mit den zugehörigen, im Kontext von UNCOVER entstandenen/vorangetriebenen Publikationen dargestellt.

5.1 Wissenschaftliche Verwertbarkeit und Anschlussfähigkeit der Arbeiten

Die wissenschaftliche Verwertbarkeit und Anschlussfähigkeit der durch die ERNW Research GmbH im Zuge dieses Teilvorhabens geleisteten Arbeiten wird durch eine ganze Reihe begleitender Publikationen belegt, welche im Projektzeitraum erfolgreich auf verschiedenen internationalen Konferenzen platziert werden konnten (vgl. Kapitel 5.5).

Die Entwicklung der in Kapitel 4.2 beschriebenen Stromchiffre DRACO hat zudem eine internationale Forschungskooperation zwischen der ERNW Research GmbH, der Universität Mannheim, der Bauhaus-Universität Weimar und der Universität Hyogo, Japan, hervorgebracht, im Rahmen derer die Sicherheit von Stromchiffren in Post-Quantum-Szenarien untersucht und entsprechende neue, leichtgewichtige Designs entwickelt werden. Hier hat sich DRACO als vielversprechender Kandidat erwiesen. Allgemein erlaubt das modulare Design der Chiffre den Einsatz in einer Vielzahl auch über den Automobilkontext hinausgehender Szenarien wie bspw. bei der Absicherung medizinischer Implantate.

Weiterhin wurde auf Basis der Erfahrungen mit DRACO Mitte 2023 zusammen mit der Universität Mannheim und der Universität Hyogo, Japan, ein Projekt zur Entwicklung einer energieeffizienten Chiffre für 6G-Applikationen ins Leben gerufen. Gegenseitige Forschungsbesuche dieses neuen Projekts, welche voll durch staatliche japanische Fördergelder finanziert werden, haben bereits in 2023 stattgefunden; weitere sind geplant. Nach Abschluss der initialen Designphase steht hier die Einbindung relevanter japanischer Industrieakteure im Bereich 6G an.

Zu den Details der wissenschaftlichen Verwertbarkeit und Anschlussfähigkeit der Ergebnisse im Kontext der in bilateraler Zusammenarbeit mit dem Projektpartner KIT entstandenen Publikationen [SHB23, SHTB23] wird auf den Abschlussbericht zum Teilvorhaben des besagten Projektpartners verwiesen. Entsprechende Informationen bzgl. der in Zusammenarbeit mit sämtlichen Projektpartnern entstandenen Publikation [SLS+24] finden sich im Gesamtbericht.

5.2 Verwertbarkeit im Projektanschluss

Die Aufarbeitung und Verwertung der Ergebnisse und Erkenntnisse des Projekts ist für die ERNW Research GmbH wichtig, da sie sich ihrem Selbstverständnis nach in der Verantwortung sieht, sicherheitsrelevante Probleme und Lösungen mit der wissenschaftlichen Gemeinschaft und der Industrie gleichermaßen zu teilen. Hinsichtlich ihrer wirtschaftlichen Verwertbarkeit werden die Projektergebnisse mit bestehenden ERNW-Produkten wie Penetrationstests, Schwachstellenanalyse und Erstellung von Security-Konzepten vernetzt, um dort eine effizientere sowie präzisere Vorgehensweise zu ermöglichen und damit den Sicherheitsgrad von Fahrzeugsystemen weiter zu erhöhen. Dabei spielt die im Zuge von UNCOVER erfolgte Weiterentwicklung von Methoden und firmeninternen Tools eine wichtige Rolle.

Die folgenden konkreten Aktivitäten/Produkte sind im Projektanschluss vorgesehen:

- Vorstellung der Projektergebnisse und Publikationen auf weiteren Konferenzen, Workshops und Industrietreffen.
- Ausbau der in Kapitel 5.1 beschriebenen, neu entstandenen Forschungs- und Industriekontakte einschließlich Erweiterung des ERNW-Produktportfolios auf die zugehörigen Themenfelder.
- Definition von Beratungsprodukten für die Automotive-Industrie:
  - Bedrohungsmodellierung Fahrzeugsysteme,
  - Kontinuierliche Begleitung der Entwicklung von Fahrzeugsystemen,
  - Review sicherer Engineering-Prozess im Automotive-Bereich.
- Schulungsangebot: Sicherheitsmodellierung im Automotive-Bereich.

5.3 Bekanntgewordener Fortschritt an anderen Stellen

Der im Projektverlauf bekanntgewordene Fortschritt an anderen Stellen wird im Abschlussbericht des Gesamtvorhabens im Detail beschrieben.

Während der Laufzeit des Projekts wurden keine Forschungsergebnisse bekannt, die mit den hier geschilderten Konzepten und Methoden im Widerspruch stehen.

5.4 Bezug zum zahlenmäßigen Nachweis

Der mit Abstand größte Anteil der im Rahmen des Teilvorhabens entstandenen Kosten ist durch die Personalkosten der mit der Projektdurchführung betrauten Mitarbeiter gegeben.

Zur Durchführung der in Kapitel 4.3 geschilderten Sicherheitsüberprüfung der Monitoring-Plattform war die Anschaffung des Entwicklungsboards *AMD Zynq UltraScale+ MPSoC ZCU102 Evaluation Kit* sowie des CAN-Adapters *PEAK PCAN-USB-FD* nötig.

Reisekosten sind im Rahmen der Konsortialtreffen und Ergebnisvorstellungen angefallen.

5.5 Veröffentlichungen

Die folgenden Veröffentlichungen sind im Projektkontext unter Beteiligung von Mitarbeitern der ERNW Research GmbH entstanden:

- Matthias Hamann, Alexander Moch, Matthias Krause, Vasily Mikhalev. *The DRACO Stream Cipher: A Power-efficient Small-state Stream Cipher with Full Provable Security against TMDTO Attacks*. IACR Transactions on Symmetric Cryptology (ToSC), 2022(2): 1–42. [HMKM22]

- Matthias Hamann, Alexander Moch, Matthias Krause, Vasily Mikhalev. *The DRACO Stream Cipher – FSE 2023 Presentation* (mit zusätzlichen, neuen Vorschlägen für ein Update des Original DRACO Key Schedules). *https://iacr.org/submit/files/slides/2023/fse/fse2023/tosc2022_2_14/slides.pdf*. FSE 2023. 2023-03-20, Kobe, Japan. [HMKM23]
- Matthias Stammler, Matthias Hamann, Jürgen Becker. *Multilevel Security Model for Secure Information Flow Inside Software Components Employing Automated Code Generation*. 2023 12th Mediterranean Conference on Embedded Computing (MECO). IEEE, 2023. [SHB23]
- Matthias Stammler, Matthias Hamann, Tanja Harbaum, Jürgen Becker. *Mitigating Masking in Automotive Communication Systems: Modeling and Hardware Generation*. 2023 26th Euromicro Conference on Digital System Design (DSD). IEEE, 2023. [SHTB23]
- Matthias Stammler, Julian Lorenz, Eric Sax, Jürgen Becker, Matthias Hamann, Patrick Bidinger, Andreas Dewald, Paraskevi Georgouti, Alexios Camarinopoulos, Günter Becker, Klaus Finsterbusch, Maximilian Kirschner, Laurenz Adolph, Carl Philipp Hohl, Maria Rill, Daniel Vonderau, Victor Pazmino. *UNCOVER: Data-Driven Design Support through Continuous Monitoring of Security Incidents.* 2024 Design, Automation and Test in Europe Conference and Exhibition (DATE). [SLS+24]

ERNW RESEARCH
pursuing knowledge.

Literaturverzeichnis

[ÅHJM11] Martin Ågren, Martin Hell, Thomas Johansson, and Willi Meier. *Grain-128a: A New Version of Grain-128 with Optional Authentication*. IJWMC, 5(1):48–59, December 2011.

[AM15] Frederik Armknecht and Vasily Mikhalev. *On Lightweight Stream Ciphers with Shorter Internal States*. In FSE 2015, pages 451–470. Springer, 2015.

[Bab95] Steve H. Babbage. *Improved "exhaustive search" attacks on stream ciphers*. In European Convertion on Security and Detection 1995, pages 161–166, May 1995.

[Ban22] Subhadeep Banik. *Cryptanalysis of Draco*. IACR Transactions on Symmetric Cryptology, 2022(4), 92–104.

[BCI+21] Subhadeep Banik, Andrea Caforio, Takanori Isobe, Fukang Liu, Willi Meier, Kosei Sakamoto, and Santanu Sarkar. *Atom: A Stream Cipher with Double Key Filter*. IACR Transactions on Symmetric Cryptology, 2021(1), 5–36.

[BS00] Alex Biryukov and Adi Shamir. *Cryptanalytic Time/Memory/Data Tradeoffs for Stream Ciphers*. In Tatsuaki Okamoto, editor, ASIACRYPT 2000, pages 1–13. Springer, Berlin, Heidelberg, 2000.

[CP05] Christophe De Cannière and Bart Preneel. *Trivium – Specifications*. eSTREAM: the ECRYPT Stream Cipher Project, 2005.

[DNR21] Yuri Gil Dantas, Vivek Nigam, and Harald Rueß. *Security Engineering for ISO 21434*. *https://www.fortiss.org/fileadmin/user_upload/06_Ergebnisse/Whitepaper/fortiss-whitepaper-security-engineering-ISO-web.pdf*, fortiss Whitepaper, 2021.

[Gol96] Jovan Dj. Golić. *On the security of nonlinear filter generators*. In Dieter Gollmann, editor, FSE 1996, pages 173–188. Springer, Berlin, Heidelberg, 1996.

[HJM06] Martin Hell, Thomas Johansson, and Willi Meier. *Grain – A Stream Cipher for Constrained Environments*. eSTREAM: the ECRYPT Stream Cipher Project, 2006.

[HJM+2021] Martin Hell, Thomas Johansson, Alexander Maximov, Willi Meier, Jonathan Sönnerup, and Hirotaka Yoshida. *Grain-128AEADv2 - A lightweight AEAD stream cipher*. NIST Lightweight Cryptography, Finalists, 2021.

[HK18] Matthias Hamann and Matthias Krause. *On stream ciphers with provable beyond-the-birthday-bound security against time-memory-data tradeoff attacks*. Cryptography and Communications, pages 959-1012. Springer, 2018.

[HKM17a] Matthias Hamann, Matthias Krause, and Willi Meier. *A Note on Stream Ciphers that Continuously Use the IV*. IACR Cryptology ePrint Archive, 2017:1172, 2017.

[HKM17b] Matthias Hamann, Matthias Krause, and Willi Meier. *LIZARD – A Lightweight Stream Cipher for Power-constrained Devices*. IACR Transactions on Symmetric Cryptology, 2017(1), 45–79.

[HKM19] Matthias Hamann, Matthias Krause, and Alexander Moch. *Tight Security Bounds for Generic Stream Cipher Constructions*. In SAC 2019, pages 335–364. Springer, 2019.

[HKMZ18] Matthias Hamann, Matthias Krause, Willi Meier, and Bin Zhang. *Design and Analysis of Small-state Grain-like Stream Ciphers*. Cryptography and Communications, 10(5):803–834, 2018.

[HMKM22] Matthias Hamann, Alexander Moch, Matthias Krause, and Vasily Mikhalev. *The DRACO Stream Cipher: A Power-efficient Small-state Stream Cipher with Full Provable Security against TMDTO Attacks*. IACR Transactions on Symmetric Cryptology, 2022(2), 1–42.

[HMKM23] Matthias Hamann, Alexander Moch, Matthias Krause, and Vasily Mikhalev. *The DRACO Stream Cipher – FSE 2023 Presentation*. *https://iacr.org/submit/files/slides/2023/fse/fse2023/tosc2022_2_14/slides.pdf*. FSE 2023. 2023-03-20, Kobe, Japan.

[MAM16] Vasily Mikhalev, Frederik Armknecht, and Christian Müller. *On Ciphers that Continuously Access the Non-volatile Key*. IACR Transactions on Symmetric Cryptology, 2016(2), 52–79.

[MV15] Charlie Miller and Chris Valasek. *Remote exploitation of an unaltered passenger vehicle*. Black Hat USA 2015, S 91 (2015): 1-91, 2015.

[SHB23] Matthias Stammler, Matthias Hamann, and Jürgen Becker. *Multilevel Security Model for Secure Information Flow Inside Software Components Employing Automated Code Generation*. 2023 12th Mediterranean Conference on Embedded Computing (MECO). IEEE, 2023.

[SHTB23] Matthias Stammler, Matthias Hamann, Tanja Harbaum, and Jürgen Becker. *Mitigating Masking in Automotive Communication Systems: Modeling and Hardware Generation*. 2023 26th Euromicro Conference on Digital System Design (DSD). IEEE, 2023.

[SLS+24] Matthias Stammler, Julian Lorenz, Eric Sax, Jürgen Becker, Matthias Hamann, Patrick Bidinger, Andreas Dewald, Paraskevi Georgouti, Alexios Camarinopoulos, Günter Becker, Klaus Finsterbusch, Maximilian Kirschner, Laurenz Adolph, Carl Philipp Hohl, Maria Rill, Daniel Vonderau, Victor Pazmino. *UNCOVER: Data-Driven Design Support through Continuous Monitoring of Security Incidents*. 2024 Design, Automation and Test in Europe Conference and Exhibition (DATE).