Schlussbericht IDEA

Auftragnehmer:	*Kennzeichen:*
Philotech Systementwicklung und Software GmbH	20Y1712C

Auftragsbezeichnung:

Integrierte Design- und Entwicklungsumgebung

Laufzeit des Projektes:

01.01.2019 – 31.12.2022

IDEA

Integrierte Design- und Entwicklungsumgebung für Aerospace

Gefördert durch:

Bundesministerium
für Wirtschaft
und Klimaschutz

aufgrund eines Beschlusses
des Deutschen Bundestages

philotech

Systementwicklung und Software GmbH

Eschenstrasse 2

82024 Taufkirchen

Telefon +49 (0) 89 – 610 898 - 0

Telefax +49 (0) 89 – 610 898 - 10

E-Mail : info@philotech.de

WWW : www.philotech.de

Inhaltsverzeichnis

1 Einführung

1.1 Identifikation

Im Rahmen des Verbundvorhabens „Integrierte Design- und Entwicklungsumgebung für Aerospace“ *(Kurzname: IDEA);* ist ein Schlussbericht anzufertigen.

Die Eckdaten für diesen Schlussbericht sind:

Tabelle 1 Eckdaten Schlussbericht

Zuwendungsempfänger: Philotech GmbH	Förderkennzeichen: 20Y1712C
Vorhabenbezeichnung	Effiziente Entwicklung eingebetteter sicherer Luftfahrtsysteme
Laufzeit des Vorhabens	01/2019 – 12/2022
Berichtszeitraum	01/2019 – 12/2022

1.2 Zweck

Der Schlussbericht gibt in einer kurzen Darstellung die Aufgabenstellung, Voraussetzungen, Planung und Ablauf des Vorhabens wieder. Es sind ferner Angaben zum wissenschaftlich-technischen Stand und der Zusammenarbeit mit anderen Stellen enthalten.

In einer eingehenden Darstellung werden folgende Themen erläutert:

1. Verwendung der Zuwendung, erzielte Ergebnisse
2. Wichtigste Positionen des zahlenmäßigen Nachweises
3. Notwendigkeit und Angemessenheit der geleisteten Arbeit
4. Voraussichtlicher Nutzen, Verwertbarkeit des Ergebnisses
5. Fortschritt auf dem Gebiet bei anderen Stellen
6. Veröffentlichungen des Ergebnisses

2 Kurze Darstellung

2.1 Aufgabenstellung

IDEA hatte zum Ziel, die deutsche Luft- und Raumfahrtindustrie in die Lage zu versetzen, komplexe, zuverlässige und wartbare, softwareintensive Avionik Systeme, bei wachsendem Kostendruck und zunehmender Konkurrenz bereit zu stellen. Innerhalb von IDEA sollen dabei die Grundlagen und Kompetenzen der Disziplin Software Entwicklung auf die Systemebene gehoben werden. Gerade die bekannten Konzepte aus dem Bereich zur Komplexitätsbeherrschung sollen auf der Systemebene neue Synergien erzeugen.

Das Projekt zielte auf eine Erarbeitung neuer Technologien für die Umsetzung des Entwicklungsprozesses komplexer Softwaresysteme in der Luft- und Raumfahrt sowie eine stärkere Vernetzung der Unternehmen der deutschen Luft- und Raumfahrtindustrie. Die Arbeitspakete wurden deshalb explizit als querschnittliche unternehmensübergreifende Arbeitspakete definiert.

Ein weiteres wesentliches Ziel im IDEA-Vorhaben war die Erweiterung von Prozessvorgaben für den Airworthiness Security Process nach DO-326A und insbesondere die Eruierung der Umsetzung einzelner Arbeitsschritte die bereits in DO-356A beschrieben sind und einen Anhaltspunkt für die Prozesskonformität bieten. Nachdem in Vorgängerprojekten bereits erste Erfahrungen und ein Einblick in die notwendigen Prozessschritte geschaffen wurde, sollten in diesem Projekt detaillierte Umsetzungen einzelner Aspekte, wie beispielsweise die sinnvolle Umsetzung von TARAs (Threat Analysis and Risk Assessment) aber auch weiterführende Aspekte nach DO-355 wie ein Avionic SOC (Security Operation Center) auf Basis einer verteilten IDS (Intrusion Detection System) Infrastruktur, analysiert werden.

Die von Philotech zu bearbeitenden Teilvorhaben zum Projekt IDEA „Integrierte Design- und Entwicklungsumgebung“ enthielt eine angemessene Beteiligung an drei von vier Teilprojekten (TP) und befasste sich im Schwerpunkt mit zwei der vier Teilprojekte.

Das Teilvorhaben von Philotech hatte folgende zwei Teilprojekte als Schwerpunkt:

1. **TP 3: Anwendungsfälle**

   In diesem Teilprojekt wurde für eine spezifische Avionik Applikation, einem „Anwendungsfall“, eine prototypische Umsetzung von Erkenntnissen aus anderen Teilprojekten durchgeführt. Dabei wurden hauptsächlich die Erkenntnisse aus Arbeitspaket 4.1 Safety & Security umgesetzt. Zusätzlich dazu wurden die Ergebnisse der Umsetzung im Rahmen dieses Teilprojektes geprüft und evaluiert. Die Evaluation ist gerade im Hinblick auf die spätere produktive Umsetzung wichtig, weil erst Indikatoren für Aufwand und Nutzen entstehen.

   Philotech hat hier besonderen Wert auf die Erkenntnisse bezogen auf Safety & Security im Softwareentwicklungsbereich gelegt, da dieser zu einem Kerngeschäftsbereich des Unternehmens zählt. Verbesserungspotentiale in diesem

Bereich zu bergen sind mittelfristig ein Garant für die Weiterführung der Umfangreichen Geschäftsbeziehungen mit verschiedensten Kunden. Weiterführende Automatisierungen können Kosten senken und schnellere Produktzyklen ermöglichen.

2. **TP 4: Querschnittsthemen**

   In diesem Teilprojekt sollten die zum einen die Harmonisierung der Safety & Security Aktivitäten geprüft und vorangetrieben werden. Die mittlerweile verpflichtenden Anforderungen für die Security von Neuentwicklungen im Luftfahrtbereich und der damit entstehende Mehraufwand bedarf einer Optimierung, um langfristig sicher und kosteneffizient Fortschritte zu erzielen.

   Zusätzlich war in diesem Teilprojekt die Abstimmung mit Behörden wie der EASA geplant, um die Harmonisierung der Umsetzungen für Safety & Security auf Zulässigkeit zu prüfen.

Im Folgenden sind alle Arbeitspakete aufgelistet und kurz beschrieben, welche im Rahmen der Forschungskooperation zusammen mit den Projektpartnern durchgeführt wurden:

1. **AP 0 Projektinitialisierung, Analyse**
   In diesem Arbeitspaket wurden administrative Aufgaben, die für die Bearbeitung aller 10 Arbeitspakete notwendig sind, durchgeführt. Das Ziel dieses Arbeitspaketes war es, eine Arbeitsumgebung für einen reibungslosen Start der anschließenden Arbeitspakete zu gewährleisten.
2. **AP 2.1 Schnittstelle Software-System**
   In diesem Arbeitspaket wurde eruiert, welche Möglichkeiten der Übertragung von Teststrategien und Methoden speziell aus dem Softwareentwicklungsbereich auf den Systembereich bestehen.
3. **AP 3.1 Spezifische Avionik Applikation**
   In diesem Arbeitspaket wurden die gewonnenen Erkenntnisse und entwickelten Lösungsansätze für eine realistische Avionik Applikation umgesetzt. Speziell die Implementierung der Ergebnisse aus *AP4.1 Safety & Security* sollten hier realisiert werden.
4. **AP 3.2 Benchmarking**
   In diesem Arbeitspaket wurde untersucht, welche Eigenschaften die Umsetzung in Arbeitspaket 3.1 mit sich bringt.
5. **AP 4.1 Safety & Security**
   Als Kernaspekt dieses Arbeitspaketes gilt es eine systematische Umsetzung von Security als Teil von Safety auszuarbeiten. Dafür wird konkret untersucht, welche Mechanismen und Architekturen das gleichzeitige Erfüllen von Safety und Security Anforderungen erbringen können und wie diese konkret umzusetzen sind. Für etablierte Maßnahmen im Bereich Safety wird ebenfalls ergründet, welche Erweiterungen durchgeführt werden müssen, um zusätzlich Security Anforderungen gerecht zu werden.
6. **AP 4.2 Zertifizierungsaspekte**
   Neue Entwicklungen in der Luftfahrt stehen und fallen mit der Akzeptanz durch die zulassende Behörde. Sowohl Tool- und Prozessverbesserungen können sich nur

Durchsetzen, wenn es eine positive Gesamtbewertung durch die Behörde, in diesem Fall durch die EASA, gibt. In diesem Arbeitspaket sollten alle zulassungsrelevanten Aspekte aller anderen Arbeitspakete gebündelt und zusammen mit der EASA eruiert werden.

2.2 Voraussetzungen unter denen das Vorhaben durchgeführt wurde

IDEA war ein Verbundprojekt, welches aufgrund der guten Ergebnisse der Vorgängerprojekte (ASSET; ASSET-2) ins Leben gerufen und im deutlich erweiterten Verbund durchgeführt wurde. Ziel war es die offenen Aspekte des Vorgängerprojektes adäquat zu adressieren und deren Ergebnisse in die firmeninternen Entwicklungsprozesse zu integrieren. Zusätzlich dazu sollten weitere Aspekte berücksichtigt werden, welche nicht in der Zielsetzung des Vorgängerprojektes verankert waren.

Zulassungsrelevante Fragen spielten vor allem für die Harmonisierung von Safety & Security eine wesentliche Rolle und wurden in diesem Vorhaben entsprechend konsolidiert und so aufgearbeitet, dass für zukünftige Entwicklungen generelle Ansätze entstanden sind. Zusätzlich zu den zulassungsrelevanten Fragen im Bereich Safety & Security wurden durch dieses Vorhaben auch die prototypische Umsetzung einzelner Aspekte geprüft und durchgeführt.

Eine weitere Voraussetzung ist die sich zuspitzende Situation auf dem Arbeitsmarkt für sehr spezialisierte Branchen. Studenten erhalten eine möglichst breit gefächerte Ausbildung, um auf verschiedenste Anforderungen möglichst gut vorbereitet zu sein. Dabei kann oft das nötige Detailwissen nicht im Studium vermittelt werden, welches aber in hoch spezialisierten Brachen notwendig ist. Auch Quereinsteigern fehlen wichtige Ausbildungsbausteine, um direkt in der Luftfahrtindustrie tätig zu werden. Auch wenn es keine direkte Beteiligung am Arbeitspaket 4.3 Wissenstransfer geplant war, wurde in der Zusammenarbeit mit der THI weitere Konzepte für Schulungen und Fortbildungen konzipiert.

IDEA war ein Verbundprojekt, welches vom Verbundführer Diehl Aerospace geleitet wurde. Nachfolgend wird jeder Partner kurz vorgestellt:

2.2.1 Diehl Aerospace GmbH (Verbundführer)

Diehl Aerospace ist einer der führenden deutschen Anbieter für Avionik- und Beleuchtungssysteme, sowohl für zivile als auch für militärische Luftfahrzeuge, und bietet Systemlösungen und -funktionen von der Avionik über Cockpit-Ausrüstungen bis in die Kabine an. Partner sind große Luftfahrzeughersteller, wie z.B. Airbus, Boeing, Bombardier, Embraer oder Airbus Helicopters.

Diehl Aerospace entwickelt sicherheitskritische, embedded Produkte unterschiedlicher Kritikalitätsstufen. Diehl Aerospace besitzt Erfahrungen auf dem Gebiet der Software- und Systementwicklung für „Development Assurance Level“ bis zu DAL A.

Es wurde in dem Vorgängerprojekt ASSET die ersten Analysen hinsichtlich des Einsatzes von Methoden und Tools in den Softwareentwicklungsprozessen durchgeführt.

Im Nachfolgeprojekt ASSET-2 (ab 2017) werden die theoretischen Ergebnisse praxisbezogener in den Softwareentwicklungsprozess eingebunden, um dies in IDEA auf den Systementwicklungsprozess zu erweitern.

2.2.2 Nord-Micro GmbH & Co OHG

Nord-Micro mit Sitz in Frankfurt am Main ist langjähriger Lieferant von Kabinendruckregelsystemen (CPCS) und Ventilationssystemen (VCS) für die größten zivilen Flugzeughersteller Airbus, Boeing und Embraer. Hierunter fallen nicht nur die mit sehr hohen Stückzahlen verkauften Programme A320 und B737 sondern auch die neuesten Programme B787, A380 und A350. Die Entwicklung dieser Produkte ist mit Hinblick auf die Systemkomplexität und die Anforderung an die Software sehr anspruchsvoll. So gilt für CPCS als ein Merkmal die höchste Softwarekritikalität und für VCS als ein weiteres die hohe Komplexität, welche durch die Unterschiedlichkeit und die Menge der verarbeiteten Daten gegeben ist. System- und Softwareentwicklungsmethoden sind deshalb für Nord-Micro essenziell für die Entwicklungszeit bis zu einem zugelassenen System und die Höhe der dafür eingesetzten Entwicklungsressourcen. Auch spielen die Aspekte der Wiederverwendbarkeit und der Wartbarkeit von Software eine erhebliche Rolle.

2.2.3 Philotech Systementwicklung und Software GmbH

Seit 1987 ist das mittelständische Unternehmen Philotech Systementwicklung und Software GmbH mit Hauptsitz in München sehr erfolgreich auf dem Markt für hoch spezialisierte Ingenieurdienstleistungen tätig. Schwerpunkte dabei sind die Geschäftsbereiche: Systems Engineering, Support Engineering, Software-Engineering, Verifikation & Validation, Design & Stress, Manufacturing Engineering.

Die Industriebereiche sind: Luft- und Raumfahrt, Rüstung, Elektronik, Schifffahrt, Automobil, Schienenverkehr, Medizintechnik, Telekommunikation sowie Maschinenbau. Seit Januar 2022 ist die Philotech Gruppe Teil der Bertrandt AG.

2.2.4 Technische Hochschule Ingolstadt

Die Technische Hochschule Ingolstadt ist eine dynamische und engagierte Hochschule für angewandte Wissenschaften. Seit ihrer Gründung 1994 vermittelt sie eine Bildung in den Bereichen Technik und Wirtschaft, die die Studierenden in die Lage versetzt, wissenschaftliche Methoden in der Berufspraxis anzuwenden.

Die Hochschule für angewandte Wissenschaften (HAW) wurde im Juli 2013 aufgrund ihrer hervorgehobenen Positionierung insbesondere im Forschungsbereich vom bayerischen Wissenschaftsministerium zur Technischen Hochschule Ingolstadt (THI) ernannt. In Kompatibilität zu der eingereichten Skizze werden die Begriffe HAW und THI synonym verwendet.

Mit über 100 Professor/innen in drei Fakultäten versteht sich die Technische Hochschule Ingolstadt als Hochschule für Wirtschaft und Technik mit anwendungsbezogenem Profil. Die THI kann auf Basis der laufenden und durchgeführten Projekte im

Bereich der Softwarearchitektur und -optimierung im Bereich Entwicklung von verteilten, sicherheitskritischen Systemen auf einen breiten Erfahrungsschatz zurückgreifen.

2.2.5 Airbus Defense and Space GmbH, Immenstaad

Airbus DS GmbH ist weltweit führend bei der Entwicklung und Herstellung von Satellitensystemen, Nutzlasten, Bodeninfrastruktur und Raumfahrtausrüstung für eine Vielzahl ziviler und militärischer Anwendungen. Airbus DS produziert Telekommunikations-, Erdbeobachtungs-, Forschungs- und Navigationssatelliten. Die Telekommunikationssatelliten von Airbus DS, die auf den Plattformen der Eurostar-Familie basieren, machen Telefon-, TV- und Internetdienste für Millionen von Benutzern in aller Welt für Geschäft, Freizeit oder Kontaktpflege verfügbar. Die Erdbeobachtungssatelliten von Airbus DS – wie Sentinel-2 und CryoSat – ermöglichen das Sammeln von Informationen für verschiedene Bereiche, z. B. Wettervorhersagen, Klimaüberwachung und militärische Aufklärung, wohingegen die Forschungssatelliten wie das Herschel-Weltraumteleskop z. B. für die Erforschung von Planeten eingesetzt werden. In all diesen Systemen ist der Anteil der Software in den letzten Jahren enorm gewachsen. Damit die Kosten nicht explodieren sind deutliche Produktivitätsfortschritte bei der Entwicklung der Software notwendig, Das Projekt IDEA liefert einen signifikanten Beitrag zu den Bestrebungen zu Verbesserung der Produktivität

2.2.6 Aerospace Embedded Solutions GmbH

AES Aerospace Embedded Solutions ist ein 50/50 Joint Venture zwischen MTU Aero Engines und SAFRAN Electronics & Defense mit Sitz in München.

AES ist ein führender Anbieter im Bereich der Entwicklung sicherheitskritischer Regelungs- und Überwachungssysteme (Hardware und Software) für die zivile und militärische Luftfahrtindustrie. AES ist europäischer Marktführer für militärische Triebwerksregelungssysteme und kompetenter Anbieter von Steuerungssystemen, z.B. für Flugzeug-Fahrgestelle und Bremssysteme.

2.2.7 Liebherr-Aerospace Lindenberg GmbH

Liebherr-Aerospace als weltweit anerkannter Ausrüster der Luftfahrtindustrie entwickelt und produziert komplette hydraulische, mechanische und elektronische Systeme der Flugsteuerung und Fahrwerke für Anwendungen im Großflugzeugbereich, für Regionalflugzeuge, Hubschrauber und Militärflugzeuge für den globalen Markt. Kunden sind u.a. Airbus, Airbus Helicopters, Bombardier, Embraer, Israel Aircraft Industries, Sukhoi, Eurofighter, AgustaWestland, Aermacchi und Commercial Aircraft Corporation of China.

Das Unternehmen hat sich erfolgreich an allen LuFo-Programmen sowie weiteren regionalen, nationalen und EU-Technologieprogrammen beteiligt.

2.2.8 Fortiss GmbH

fortiss ist als An-Institut an der Technischen Universität München eine universitätsnahe, aber rechtlich unabhängige, nichtkommerzielle Forschungseinrichtung in der Rechtsform einer gemeinnützigen GmbH.

Gesellschafter des fortiss sind der Freistaat Bayern (als Mehrheitsgesellschafter) und die Fraunhofer Gesellschaft zur Förderung der angewandten Forschung e.V.

Am fortiss werden Forschungs- und Technologietransfer zu software-intensiven Systemen und Services möglichst weitsichtig vorbereitet, um dadurch weitere Innovationsimpulse, insbesondere in Bayern, auszulösen.

Das fortiss dient dabei als Brücke zwischen der universitären Grundlagenforschung und deren Umsetzung in die industrielle Praxis und schließt damit eine oft beklagte Lücke.

Der Kompetenzfeld „Model based Systems Engineering“ und das Labor „ESSEI Labs“ haben viele Erfahrung in den folgenden Domänen die für IDEA hochrelevant sind:

- Luftfahrt Domäne: durch die Teilnahme an den Projekte SysTavio, ASSET I & II wurde viele Erfahrung in Software-Engineering für die Luftfahrt gesammelt. Zusätzlich wurde durch die Teilnahme an AVATAR auch Erfahrung in Systems Engineering für Luftfahrt erhoben.
- Systems Engineering: durch Projekte wie SPES-XT, OBC-SA oder direkte Auftragungen mit Automotive Zulieferer hat der Kompetenzfeld viele Erfahrung an Systems Engineering gesammelt. Indirekt, aber nötig für AP1.1 hat auch der Kompetenzfeld an zahlreichende Projekte die verschiedenen Engineering Domänen zusammen bringt mitgemacht: CPSELabs, Platform4CPS, CREST, BaSys4.0.
- Safety & Security: durch Projekte wie SALSA, DMILS oder RACE wurde viele Erfahrung in Domäne funktionale Sicherheit gesammelt, insbesondere mit Nutzung von GSN (Goal Structuring Notation) für die Erstellung von Safety Cases. In ASSET II und durch enge Zusammenarbeit mit dem Lehrstuhl von Prof. Pretschner an der TU München wurde auch viele Kenntnisse in Domäne Security erhoben.

2.2.9 SYSGO AG

SYSGO wurde 1991 als SYSGO Real-Time Solutions GmbH gegründet und spezialisierte sich auf die Entwicklung und Anpassung von Software für Echtzeitanwendungen in den Bereichen Automatisierung, Netzwerk und Transport. . Eine eigenständigere Produktentwicklung begann Ende der 90er mit dem ELinOS-System (eingebettetes Linux mit Entwicklungsumgebung, Marktreife 2003) und dem PikeOS-Mikrokern (Vermarktung im Automatisierungsumfeld erstmals ab 2005/2006). 2006 wurde das Produkt-Portfolio um den Avionik-Netzwerkstack AFDX erweitert. 2009 wurde die erste DO-178B-Zertifizierung abgeschlossen, ebenso First Flight von SYSGO-eigenentwickelter Software. Im Zuge von schnellem Wachstum um die Jahrtausendwende wurde SYSGO 2002 in eine AG umgewandelt, insbesondere erlaubte dies die Integration einer französischen Niederlassung und eine Niederlassung in Tschechien.

SYSGO Fokus liegt auf der Entwicklung der Echtzeit Virtualisierungsplattform PikeOS. PikeOS bringt Virtualisierung in den embedded Markt und erlaubt damit u.a. die gleichzeitige Existenz von Anwendungen unterschiedlicher Sensitivität auf einer Hardware. Die Lösung ist für sicherheitskritische Anwendungen u.a. in der Avionik (DO-178B), IEC 61508, EN50128 zertifiziert und wird u.a. von Airbus in der neuesten Generation von Flugzeugen eingesetzt.

2.2.10 SILVER ATENA Electronic Systems Engineering GmbH

SILVER ATENA entwickelt als unabhängiger Systemlieferant sicherheitsrelevante Elektronik für Anwendungen in den Wachstumsmärkten Aerospace, Aero Engines, Automotive, Rail und Transportation. Für das Unternehmen mit Standort Deutschland sind rund 250 Mitarbeiter tätig.

SILVER ATENA übernimmt sowohl Entwicklungspakete als auch die Gesamtverantwortung für Hardware- und Software-Projekte bis hin zum Komplettsystem und liefert so einen deutlichen Mehrwert für den Kunden. Eigene Produkte wie Testsysteme, Prüfstände, Simulatoren & Entwicklungs-Tools ergänzen das Programm.

2.2.11 Karlsruher Institut für Technologie

Die Forschergruppen von Prof. Dr.-Ing. Dr. h. c. J. Becker und Prof. Dr.-Ing E. Sax am Institut für Technik der Informationsverarbeitung (ITIV) beschäftigen sich seit vielen Jahren mit dem Entwurf und der Anwendung von eingebetteten Systemen sowie dem Systems Engineering. Die Forschungsschwerpunkte liegen insbesondere im Bereich des Architekturentwurfs für eingebettete elektronische Systeme, inklusive der zugehörigen Methoden, Tools, Design-Prozesse, des Hardware/Software Co-Designs und anwendungsspezifischer Rechnerarchitekturen. Der Fokus der Forschungsarbeiten liegt auf echtzeitfähigen (domänenspezifischen) Architekturen, der Betrachtung von Safety und Security, der Integration in existierende E/E-Architekturen (Elektrik/Elektronik), Methodiken, Tools und Prozesse auch hinsichtlich ISO 26262, IT-Sicherheit und Softwareschutz. In dieses Fördervorhaben wird der Lehrstuhl seine Erfahrungen aus dem Architekturentwurf für eingebettete, gesicherte Gateway Systeme, SW- Entwicklungsprozessen, Erkenntnisse aus dem Entwurf von Safety- und Security-Mechanismen und der Grundlagen für die Anwendung und Betrieb sicherer (im Sinne von Safety und Security) Architekturen einbringen. Des Weiteren wird die Expertise im Bereich Test- und Absicherungsmethoden und Prozesse beigesteuert.

2.2.12 Konzept Informationssysteme GmbH

Seit über 20 Jahren beschäftigt sich die Konzept Informationssysteme GmbH (KIS) mit der Automatisierung, sprich Entwicklung von technologisch anspruchsvollen System- und Software- Lösungen im Bereich von sicherheitskritischen Anwendungen im Auftrag unseren Kunden aus der Luftfahrt oder dem Automobilsektor. Die durchgeführten Projekte in diesem Umfeld vereinen höchstes technologisches Niveau mit dem Anspruch absoluter Zuverlässigkeit und Sorgfalt im Entwicklungsprozess. Das Prozess-Know-how, das zum großen Teil aus langjähriger Projekterfahrung erwächst, wissen

unsere Kunden deshalb ebenso zu schätzen, wie die technische Qualifikation der Mitarbeiter. Die Schwerpunkte liegen im Bereich Requirements-Engineering, „traditionelle“ SW-Entwicklung und die Aufgaben rundum Validierung/Verifikation unter Berücksichtigung von einschlägigen Entwicklungsnormen und Zulassungsanforderungen. Daher ist es nahliegend, dass die KIS Ihre Unterstützung sowohl in den Arbeitspaketen AP1.1, 2.1, 3.1 als auch für Querschnittsthemen anbietet.

2.2.13 Airbus Defense and Space GmbH, Manching

Airbus Defence and Space GmbH, eine Division der Airbus Group, ist einer der weltweit größten Anbieter globaler Sicherheitslösungen und -systeme, der zivile und militärische Kunden als Systemintegrator und Lieferant wertschöpfender Produkte und Dienstleistungen unterstützt. Hierzu zählen Flugsysteme (Flugzeuge und unbemannte Plattformen), boden- und schiffsgestützte sowie teilstreitkräfteübergreifende Systeme, Aufklärung und Überwachung, Cybersecurity, sichere Kommunikation, Testsysteme, Flugkörper, Dienstleistungen und Supportlösungen. Im Jahr 2011 erwirtschaftete ADG mit rund 28.000 Mitarbeitern einen Gesamtumsatz von € 5,8 Milliarden.

2.3 Planung und Ablauf des Vorhabens

2.3.1 Planung

Das Projekt war in 10 Arbeitspakete aufgeteilt, die jeweils von einem Partner ausgesteuert wurden.

Nach der Projektinitialisierung sind die Abstimmungsphasen der einzelnen Arbeitspakete zu den verschiedenen Themenschwerpunkten gestartet. In diesen Phasen haben sich jeweils alle beteiligten Partner auf die Ziele und Anforderungen der Arbeitspakete geeinigt, die dann in der Folge bearbeitet wurden.

Der Zuwendungsbescheid für dieses Vorhaben ging mit einer Verzögerung von ca. 3 Monaten einher. Unterschiedlichste Gründe einzelner Projektpartner sorgten zusätzlich für Verzögerungen und teilweise Abwandlungen von einzelzielen Arbeitspaketzielen. Insbesondere die Corona Pandemie und die damit verbundenen Herausforderungen für die Gesellschaft, die Unternehmen und der beteiligten Personen sorgten teils für deutliche Abweichungen im Zeitplan. Durch angemessene Reaktion auf Verzögerungen und geeigneten Gegenmaßnahmen bei fehlenden oder nur teilweise vorhandenen Eingangsdaten konnten im Rahmen der neuen Projektlaufzeit am Ende alle Gesamt-Ziele trotzdem erreicht werden. Nicht erreichte Teilziele wurden ersetzt oder leicht abgewandelt, so dass die Gesamtziele des Projektes jederzeit erreichbar blieben und auch erreicht wurden.

In Abbildung 1 ist der Balkenplan für dieses Vorhaben und allen Arbeitspaketen mit Beteiligung der Philotech dargestellt.

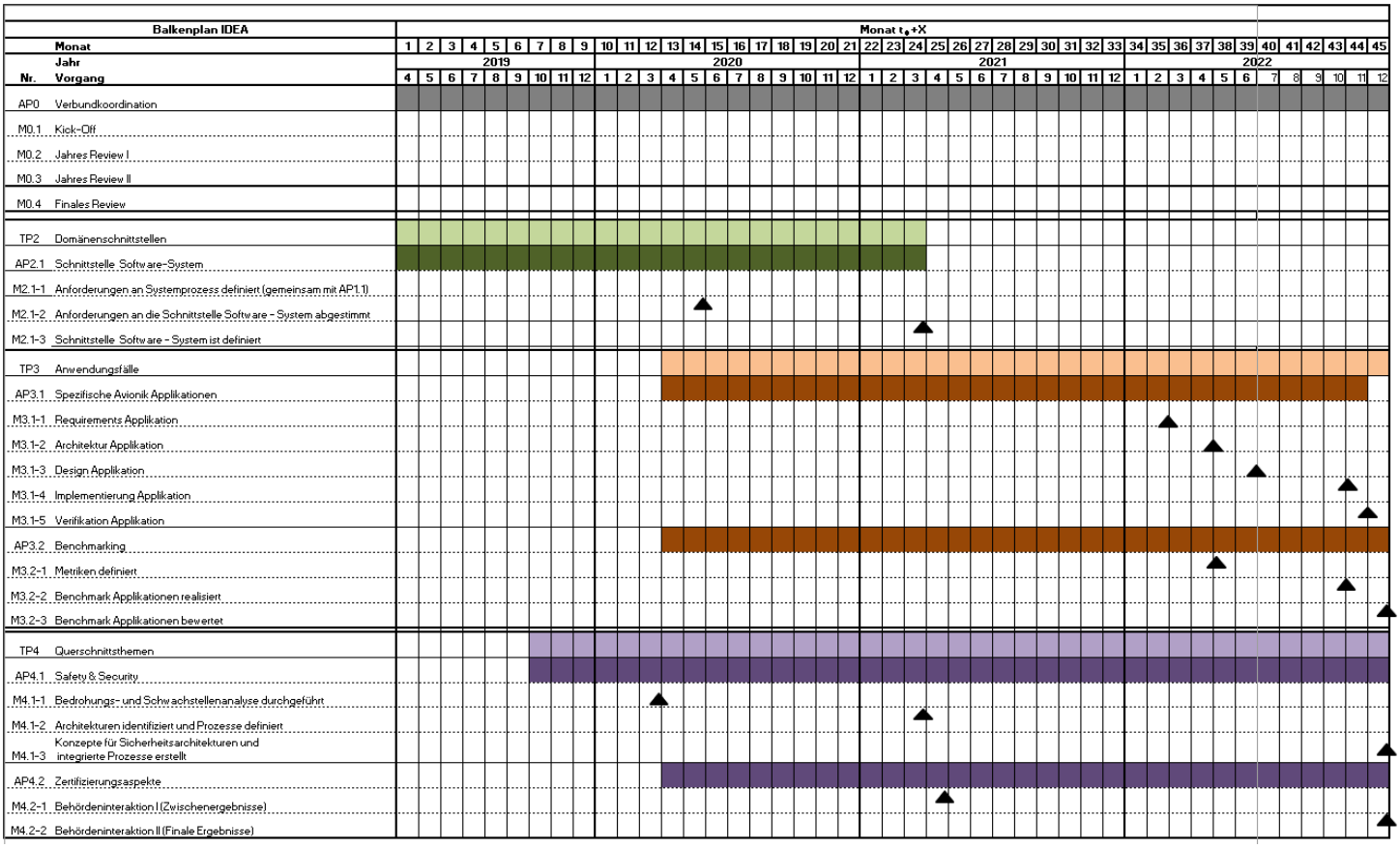

Abbildung 1 IDEA-Balkenplan der Arbeitspakete mit Beteiligung der Philotech GmbH

2.3.2 Ablauf

Folgende Arbeitspakete sind in IDEA von Philotech bearbeitet worden:

- AP 2.1 Schnittstelle Software-System
- AP 3.1 Spezifische Avionik Applikationen
- AP 3.2 Benchmarking
- AP 4.1 Safety & Security
- AP 4.2 Zertifizierungsaspekte

Philotech führte Arbeitspaket 4.1 als Arbeitspaketleiter an und arbeitete intensiv an den Teilprojekten 2.1 und 3.1 mit. Für die anderen Arbeitspakete war Philotech jeweils wie geplant an der Einigung auf Ziele und der Erhebung von Anforderungen beteiligt. Arbeitspaket 4.2 wurde nicht wie ursprünglich geplant als Unterauftrag an die EASA vergeben. Nach den gewonnenen Erkenntnissen aus Arbeitspaket 4.1 und den vorbereitenden Abstimmungen für die Unterbeauftragung wurde deutlich, dass die für Philotech relevanten Aspekte den Rahmen des möglichen Unterauftrages sprengen würden. Die EASA konnte nicht alle Themen im Unterauftrag annehmen und daher hat man sich im Konsortium auf die Themen geeinigt, welche die größte Relevanz im Gesamtvorhaben hatten.

Im Rahmen des Projektes fanden eine Reihe von Workshops statt. In den Workshops wurden unter Anderem verschiedene Implementierungswege für die Lösung von Prozessdefinitionsaufgaben besprochen, Best Practice Erfahrungen gesammelt und wiederverwendbar festgehalten. Die Arbeitspaketleiter nutzten die Workshops, um den Status der Teilaufgaben zu besprechen, die nächsten Schritte zu planen und um Wissen zu teilen.

2.3.3 Kostenneutrale Projektverlängerung

Durch die Verzögerungen durch den leicht verspäteten Zuwendungsbescheid und die größeren Herausforderungen für einzelne Partner aufgrund der Corona Pandemie die Laufzeiten einzelner Arbeitspakete um bis zu 6 Monate kostenneutral verlängert.

2.4 Anknüpfungspunkte wissenschaftlich-technischer Stand

Der wissenschaftliche und technische Stand, an dem angeknüpft wurde, wird im folgenden Abschnitt näher erläutert.

2.4.1 Bekannte Konstruktionen, Verfahren und Schutzrechte

Für die Software-Entwicklung im Luftfahrtbereich wird prozessorientiert gearbeitet. Die entstehenden Produkte sind mit Schutzrechten versehen, allerdings ist nicht zu erwarten, dass der Prozess oder die Vorgehensweise in gleicher Weise mit Schutzrechten versehen werden kann. Ein guter Prozess für den komplexen Sachverhalt der Umsetzung von RTCA DO-178C verschafft den jeweiligen Unternehmen einen Wettbewerbsvorteil über schnellere und fehlerfreiere Software-Entwicklung. Dies gilt in gleicher Weise für die in diesem Projekt etablierten Prozess zur Verwendung von paralleler Hardwarearchitektur nach RTCA DO-254, DO-297 und CAST-32A sowie für Sicherheitsbetrachtungen nach RTCA DO-326A, DO-355 und DO-356A.

2.4.1.1 Fachliteratur

Standards:

1. RTCA, *RTCA DO-178C Software Considerations in Airborne Systems and Equipment Certification, 2011*
2. RTCA, *RTCA DO-331 Model Based Development and Verification Supplement to DO-178C and DO-278A, 2011*
3. RTCA, *RTCA DO-332 Object-Oriented Technology and Related Techniques Supplement to DO-178C and DO-278A, 2011*
4. RTCA, *RTCA DO-333 Formal Methods Supplement to DO-178C and DO-278A, 2011*
5. RTCA, *RTCA DO-330 Software Tool Qualification Considerations, 2011*
6. RTCA, *RTCA DO-326A Airworthiness Security Process Specification, 2014*
7. RTCA, *RTCA DO-356A Airworthiness Security Methods and Considerations, 2018*

8. RTCA, *RTCA DO-355 Information Security Guidance for Continuing Airworthiness, 2014*
9. RTCA, *RTCA DO-254 Design Assurance Guidance for Airborne Electronic Hardware, 2000*
10. RTCA, *RTCA DO-297 Integrated Modular Avionics (IMA) Development, Guidance and Certification Considerations., 2005*
11. SAE, *ARP 4754 Certification Considerations for Highly-Integrated or Complex Aircraft Systems, 1996*

2.4.1.2 Informations- und Dokumentationsdienste

https://www.ieee.org/

https://www.springer.com/de/

https://dl.acm.org/

https://www.ossec.net/

https://suricata.io/

https://cybersecurity.att.com/products/ossim

https://www.qemu.org/documentation/

https://px4.io/

https://docs.px4.io/main/en/

https://mavlink.io/en/

https://www.ossec.net/docs/

https://suricata.readthedocs.io/en/suricata-6.0.9/

https://ieeexplore.ieee.org/

https://www.dronecode.org/

2.5 Zusammenarbeit mit anderen Stellen

Sämtliche Arbeitspakete mit Beteiligung durch Philotech wurden in einem gemeinsamen Kick Off Workshop gestartet. Im Laufe der Bearbeitung wurde in den verschiedenen Tasks mit den anderen Verbundpartnern zusammengearbeitet. Die Aufgaben mit den Verbundpartnern wurden bereits in Abschnitt 2 dargestellt. Dazu wurden regelmäßige Statusmeetings durchgeführt, später monatliche telefonische Statusrunden eingeführt. Hier wurde der aktuelle Bearbeitungsstand präsentiert und das weitere Vorgehen besprochen. Zusätzlich gab es einige speziellere Workshops zu Themen wie

Verifikation, Modellbasierte Entwicklung oder Safety & Security, bei denen ein Teilthema mit den jeweils beteiligten Partnern intensiv besprochen und weitere Schritte abgestimmt wurden.

Zusätzlich dazu wurde verschiedene themenspezifische Workshops mit Werkzeugherstellern durchgeführt, um zu eruieren, welche Möglichkeiten der Einsatz der Werkzeuge in den zu untersuchenden Forschungsthemen bestehen.

3 Eingehende Darstellung

3.1 Verwendung der Zuwendung, erzielte Ergebnisse

3.1.1 Verwendung der Zuwendung

Die Zuwendung wurde konform zur Antragstellung bzw. des überarbeiteten Arbeitsplanes (kostenneutrale Verlängerung vom 26.10.2021) verwendet. Alle Arbeitspakete der Philotech wurden in vollem Umfang bearbeitet. Aufgrund von fehlenden Kapazitäten und Umplanungen im Konsortium wurde die geplante Unterbeauftragung der EASA in AP4.2 nicht umgesetzt. Der gewünschte Wissenszuwuchs der Philotech kann aufgrund der Vielzahl von Themen im Konsortium für diesen Unterauftrag nicht erreicht werden. Philotech wird sich allerdings weiterhin im Arbeitspaket in direkter Zusammenarbeit mit den beteiligten Partnern abstimmen und austauschen. Diese Änderung zum ursprünglichen Projektplan wurde im 4. Zwischenbericht vom August 2022 dem Projektträger angezeigt und argumentiert.

Für weitere Details zum Thema zahlenmäßiger Nachweis siehe das Dokument zahlenmäßiger Nachweis gem. Nr.19.3 NKBF 98.

3.1.2 Übersicht der wichtigsten Ergebnisse

In den folgenden Abschnitten werden nun die wichtigsten Ergebnisse des Vorhabens dargestellt. Die Ergebnisse sind folgend thematisch und nicht anhand der Tätigkeiten in den einzelnen Arbeitspaketen gruppiert. Für die jeweiligen Ergebnisse ist aus Gründen der Transparenz dennoch eine Verknüpfung mit den dazugehörigen Arbeitspaketen erkenntlich.

3.1.2.1 Komplexitätsbeherrschung

Ein im Forschungsvorhaben gestelltes Ziel war der Versuch der Übertragung von bewährten Methoden zur Komplexitätsbeherrschung aus dem Software-Entwicklungsbereich auf den System-Entwicklungsbereich.

Dazu wurden die zum einen die Aspekte im Bereich Teststrategien und Methoden als auch im Bereich Methoden und Herangehensweisen für Safety & Security betrachtet.

Dieses gesteckte Ziel wurde in den Arbeitspaketen 2.1 Schnittstelle Software-System und 4.1 Safety & Security bearbeitet.

3.1.2.1.1 Teststrategien und Methoden

Anhand von Literaturrecherchen und Mitarbeiterbefragungen ist ein umfassender Überblick zu Möglichkeiten der Übertragung von Teststrategien aus dem Softwarebereich in den Systembereich entstanden. Zudem wurde die bereits etablierte ISTQB Schulung zum Certified Tester durch die gewonnenen Erkenntnisse erweitert und umfasst jetzt zusätzlich zum Software- ebenfalls den Systembereich.

Im zweiten Aspekt dieses Arbeitspaketes wurde eine Gegenüberstellung der gängigen Luftfahrtstandards ARP 4754A, DO-178, DO-254 und dem Systemansatz in INCOSE durchgeführt. Zusätzlich zur Gegenüberstellung der Dokumente und der darin formulierten Anforderungen an die jeweiligen Prozessschritte konnten unvollständig

abgedeckte Anforderungen bzw. Herangehensweisen aus INCOSE identifiziert werden. Diese können bei entsprechender zukünftiger Beachtung eine wesentliche Verbesserung in Systementwicklungsprozessen mit sich bringen.

3.1.2.1.2 Safety & Security

Zum Erreichen der gestellten Zielsetzung wurde für den Bereich Safety & Security insbesondere geprüft, wie Verfahren aus der Bedrohungsanalyse im Software-Entwicklungsbereich im System-Entwicklungsbereich eingesetzt werden können.

In diesem Bereich ist vor allem deutlich geworden, dass die bisherigen Ansätze aus dem Softwareentwicklungsbereich zwar inhaltlich gute Ergebnisse liefern können, aber dazu eine umfassende Systemmodellierung notwendig ist. Eruierte Ansätze der Modellbasierten Entwicklung bieten hier einen entscheidenden Vorteil, da hier die Modellierung schon in der Systementwicklung beginnt und kontinuierlich weitergeführt wird. Auf der späteren Ebene der Software-Entwicklung sind dann keine zusätzlichen Modellierungsschritte notwendig.

Mit Modellbasierten Entwicklungsansätzen sind somit nicht nur Kosteneinsparungen realistisch. Gleichzeitig sinkt die Fehleranfälligkeit durch doppelte Modellierungen.

In Abbildung 2 ist der erstellte Prozess zur Integration von Safety und Security im gesamten Entwicklungsprozess von Systemen der Luftfahrt dargestellt. Dieser ist unabhängig der gewählten Modellierungsherangehensweise und ist damit auch für Methoden, die nicht Modellbasiert sind, anwendbar.

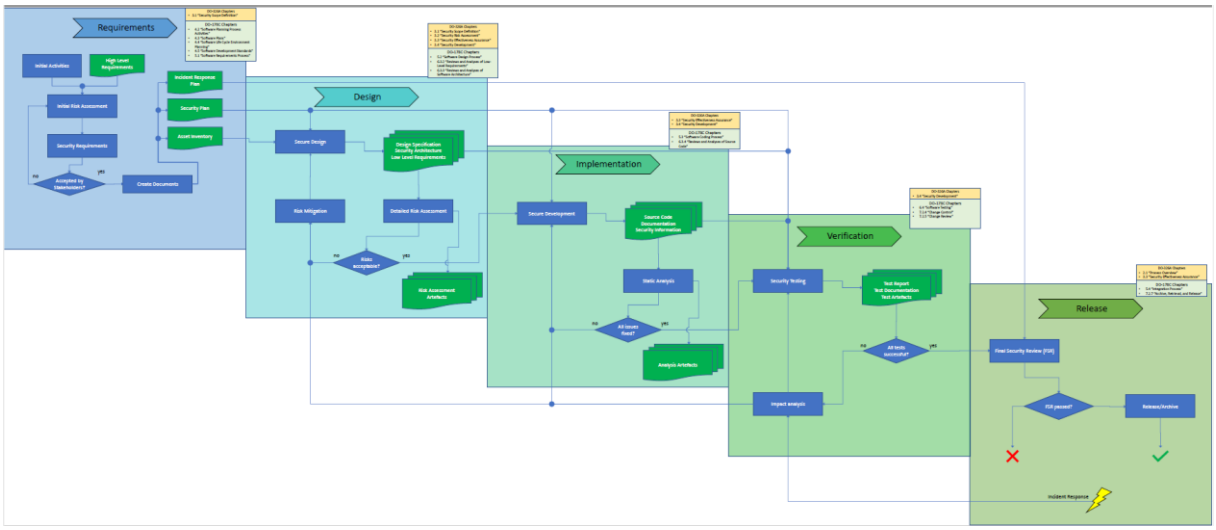

Abbildung 2 Finale Prozessbeschreibung zur Integration von Safety und Security im Entwicklungsprozess von Systemen für die Luftfahrt

3.1.2.2 Security Technologie

In diesem Abschnitt werden die wichtigsten Erkenntnisse des Projektes vorgestellt, die sich mit dem Vorhabenziel der Anwendung von Security Technologien auf der Systemebene beschäftigten.

Dazu wurden vor allem im Bereich Security und hier speziell der Angriffs- beziehungsweise Anomalie Erkennung betrachtet.

Dieses gesteckte Ziel wurde in den Arbeitspaketen 3.1 Spezifische Avionik Applikation, 3.2 Benchmarking und 4.1 Safety & Security bearbeitet.

Die durchgeführten Untersuchungen und Recherchen dienten in erster Linie zum Erlangen des grundlegenden Verständnisses der vorhandenen Methoden und Tools und der Möglichkeiten der Umsetzung dieser im Luftfahrtbereich.

IDS-Lösungen (Intrusion Detection System) und IPS-Lösungen (Intrusion Prevention System) sind im IT-Infrastrukturbereich seit vielen Jahren bekannt und stehen in diversen Open-Source Projekten zur Verfügung. Auch kommerziellen Produkten sind auf dem Markt und bedienen die Nachfrage im IT-Infrastruktur Sektor.

Generell sind die Aufgaben dieser Lösungen völlig unabhängig ihres späteren Einsatzzieles und daher war es initial interessant, welche Lösungen mit welchen Eigenschaften bereits existieren und für den Einsatz in der Luftfahrt verwendet werden können.

In einem ersten Schritt wurden einige auf dem Markt erhältliche IDS-Systeme untersucht, um ihre Funktionsweise, ihre Grenzen und die Durchführbarkeit ihrer Verwendung im Luftfahrtbereich in eingebetteten Systemen zu verstehen.

Zunächst wurde versucht nachzuvollziehen mit welcher Motivation diese IDS-Systeme entwickelt wurden. Ausgehend von diesem Verständnis konnte ein Vergleich zwischen der klassischen IT- und Systemebene und den entsprechenden Sicherheitsanforderungen aus der Luftfahrt angestellt werden.

HIDS vs. NIDS-Systeme

IDS-Lösungen sind Systeme, die in der Lage sind, potenzielle Bedrohungen oder Anomalien zu erkennen, die den Betrieb des Netzes oder des Computers gefährden können. Inhaltlich gibt es unterschiedliche Ansätze, auf welcher Basis diese Erkennung stattfindet.

Grundlegend funktionieren alle Systeme sehr ähnlich. Aufgrund einer Wahrnehmung (observierte Daten) erfolgt eine Mustererkennung, die gewünschtes von ungewünschtem Verhalten unterscheiden soll. Im Anschluss dazu erfolgt eine Reaktion. Diese Reaktion ist bei IDS-Systemen typischerweise eine einfache Meldung, welche dann manuell bearbeitet wird. Bei IPS-Systemen werden typischerweise automatisierte Aktionen anhand der Mustererkennung ausgelöst.

IDS-Systeme unterscheiden sich damit durch die Beantwortung folgender Fragen:

- Wie werden die Daten eingesammelt?
- Welche Art der Mustererkennung wird angewendet?

Im Rahmen des Vorhabens wurde vor allem die erste Frage adressiert. Diese zielt speziell auf die Machbarkeit von vorhandenen Lösungsansätzen im Bereich von eingebetteten Systemen eine größere Rolle. Die Beantwortung der zweiten Fragen zur Kategorisierung von IDS-Systemen ist Gegenstand diverser Forschungsinstitute und war für die praxisnahe Umsetzung im Rahmen dieses Vorhaben nicht weiter relevant. Gleichwohl ist die Mustererkennung in Bezug auf die Effizienz wichtig.

Die Möglichkeit der Datenerhebung kann in IDS-Systemen durch zwei verschiedene Optionen realisiert werden NIDS (Network Intrusion Detection System) oder HIDS (Host-based Intrusion Detection System).

HOST-BASED INTRUSION DETECTION SYSTEM

HID-Systeme werden mit Hilfe eines Agenten auf dem eigentlichen Host Computer umgesetzt. Dafür wird das Zielsystem mit einer zusätzlichen Applikation ausgestattet und diese wird so konfiguriert und mit Rechten versehen, dass es gewünschte Ereignisse in der Lage ist zu sammeln, kumulieren und an eine weitere Instanz zu senden. HIDS bieten in der Regel umfassendere Möglichkeiten zur Erkennung von Angreifern in Systemen, da sie sehr breit gestreute Informationen einsammeln können. Änderungen von Dateien, erhöhter Speicher- oder CPU-Bedarfe und auch Auswertungen von Log-Dateien.

NETWORK INTRUSION DETECTION SYSTEM

NID-Systeme sind in der Regel als unabhängige Plattform im Netzwerkverkehr eingebunden und dienen der Überwachung des Netzwerkverkehrs. Unerwünschte Kommunikationen zwischen bestimmten Systemen oder auch erhöhter Netzwerkverkehr lassen sich damit detektieren und entsprechend auswerten. Durch die häufig verschlüsselte Kommunikation von Systemen über Netzwerke sind unerwünschtes Verhalten oft nur durch die Analyse der Metadaten zu identifizieren.

Im Rahmen des Vorhabens wurden einige IDS-Lösungen eruiert und auf ihre Einsatzfähigkeit in eingebetteten Systemen geprüft. Unter den evaluierten Lösungen finden sich sowohl kommerzielle als auch Open-Source Lösungen. In Abbildung 3 ist ein Auszug der evaluierten IDS-Lösungen aufgezeigt. Die blau hinterlegten Zeilen zeigen Lösungen, welche im weiteren Verlauf des Projektes im Detail analysiert und prototypisch umgesetzt worden sind.

Name	Type	OS	Architecture	Positives	Negatives	Other features	Is it suitable for our usecase?
OSSEC	HIDS	Client for all OS	Client/server	World's Most Widely Used HIDS, Client is extremly light	none	FIM, rootkit detection, log monitor, configurable alerts, many plugins	yes
AlienVault - OSSIM	HIDS/NIDS Manager	Linux	Client/Server	On-premises Physical & Virtual Environments. Intrusion detection Alarms	Single Server Only. Online product documentation and knowledge base not provided in open source version.	Log monitor.	yes
Samhain Labs	HIDS	Multi-platform	Client/server,	Processing occurs on client itself	Client overload possibility, complicated installation	Very simmilar to OSSEC, flexible client, FIM, rootkit detection, log monitor, port monitor	yes
Snort	NIDS	Linux, Win	Single host, single thread	Use simple, scriptable configuration, more than 200+ plugins, network probe	More than 20-years-old, designed for older infrastructure	Third party tools provide a web front end to query and analyze alerts comming from Snort IDS	Partially, only NIDS
Suricata	NIDS	Multi-platform	Single host, multi thread	Like snort but with more plugins and much more performance, very popular	none	HW GPU acceleration, scalable code base, Lua scripting	Partially, only NIDS
Sagan	HIDS+NIDS	Linux, MacOS	Single host, multi thread	Written in C, high performance log & event analysis	Complicated instalation	Compatible with Snort and Suricata rule management software, analyzes Snort data	yes
Bro (renamed Zeek)	NIDS	Linux, MacOS	Single host	Own policy script interpreter -> Bro-Script robust programming language – which is used to interact with events and understand what those events mean in terms of network security	No GUI, complicated to set up, needs a lot of dependecies	Converts data about network traffic into higher-level events	Partially, only NIDS
Security Onion	NIDS	Linux	Single host	consisting of multiple, leading open-source solutions. Provides an easy setup tool for installing the whole stack	As a platform made up of several technologies, Security Onion inherits the drawbacks of each constituent tool	combining the best of Snort, Suricata, Zeek, as well as other tools such as Sguil, Squert, Snorby, ELSA, Xplico	maybe

Abbildung 3 Auszug der Einstufung der untersuchten IDS-Lösungen

Im Folgenden werden mehr Informationen der untersuchten und in einem Prototypen umgesetzten Lösungen beschrieben.

Abbildung 4 OSSEC Logo von https://www.ossec.net/

OSSEC:

Das weltweit am häufigsten verwendete Host Intrusion Detection System. OSSEC ist ein plattformübergreifendes, quelloffenes und kostenloses HIDS. Es beinhaltet umfangreiche Konfigurationsoptionen und die Möglichkeit das System vollständig auf seine Bedürfnisse zu optimieren. So existieren beispielsweise Optionen zum Hinzufügen von benutzerdefinierten Warnregeln oder dem Einbringen von eigenen Skripten zur automatisierten Auswertung auf Host Computern und Systemen. Dazu kommen Optionen zur Überwachung der Dateiintegrität, welche vollständig konfigurierbar ist.

Das Setup zum Kumulieren von Daten über mehrere Systeme hinweg ist vielfältig und besteht aus Client- und Server-Strukturen und bietet auch lokale oder agentenlose Überwachungsoptionen an.

Abbildung 5 SURICATA Logo von https://suricata.io/

SURICATA:
Suricata ist ein hochleistungsfähiges Netzwerk-IDS, IPS und Netzwerksicherheits-überwachungsmodul. Es ist Open Source und gehört einer gemeinnützigen Stiftung, der Open Information Security Foundation (OISF). Suricata wird von der OISF entwickelt.
Diese IDS, IPS-Lösung ist darauf ausgelegt den Netzwerkverkehr in komplexen Infrastrukturen zu überwachen und auf unerwünschtes Verhalten hinzuweisen. Suricata verwendet nicht nur Meta Informationen der Kommunikation, sondern kann auch dazu genutzt werden die Transportverschlüsselung aufzuheben und damit entsprechende Analyse bis zum Applikation-Layer durchzuführen.

Abbildung 6 Alien Vault OSSIM Logo von https://cybersecurity.att.com/

Alien Vault - OSSIM:
AlienVault OSSIM ist ein funktionsreiches Open-Source-Sicherheitsinformations- und Ereignisverwaltungssystem (SIEM), das die Sammlung, Normalisierung und Korrelation von Ereignissen umfasst.
Alien Vault basiert auf OSSIM und empfängt Protokollnachrichten von mehreren Geräten, normalisiert die Nachrichten in ein für Menschen lesbares Format und speichert sie in einer Datenbank. Diese Ereignisse werden mit den Regeln zur Risikoberechnung

korreliert, um Alarme auszulösen, die das Security Operation Center verwendet, um die Netzwerkumgebung in Echtzeit zu überwachen.
Protokollnachrichten werden im Syslog-Format über eine TCP/UDP-Verbindung auf Port 514 an OSSIM weitergeleitet werden. Der OSSIM-Sensor empfängt alle von den Protokollquellen weitergeleiteten Protokollnachrichten und normalisiert diese für weitere Auswertungen.

Installationsprozess der verwendeten IDS-Lösungen

Nach der initialen Evaluation der IDS-Lösungen wurden die notwendigen Schritte zu der Installation einzelner Lösungen mit entsprechenden Problemen und dazugehörigen Lösungen gesammelt und für spätere Umsetzungen ausführlich dokumentiert.
Diese Dokumentation stellt auch die Baseline für alle Untersuchungen im Kontext dieses Projektes dar.

Tabelle 2 Ergebnisartefakte IDS-Installationsverfahren

Artefakt	Beschreibung
IDS-Installationsverfahren1.0.docx	Dokumentation Schritt für Schritt Installation IDS

Testumgebung

In einem weiteren Schritt wurde eine prototypische Testumgebung aufgesetzt und mit den jeweiligen IDS-Lösungen erweitert. Mit Hilfe der Testumgebung wurde zuerst die jeweilige Funktion der IDS-Lösung und später die Machbarkeit der Umsetzung in ein eingebettetes System geprüft werden. Die entstandene Testumgebung im Rahmen dieses Vorhaben ist eine vollständig virtualisierte Umgebung, welche aber bei der Verwendung der virtualisierten Host Systeme auf Kompatibilität zu bestehenden Host Systemen aus der Luftfahrt zurückgreift.
In Abbildung 7 ist die virtualisierte Testumgebung dargestellt. Zusätzlich zu den Systemen unter Test wurden weitere Komponenten hinzugefügt, welche zur Validierung der Funktionalität und des Benchmarkings notwendig sind.

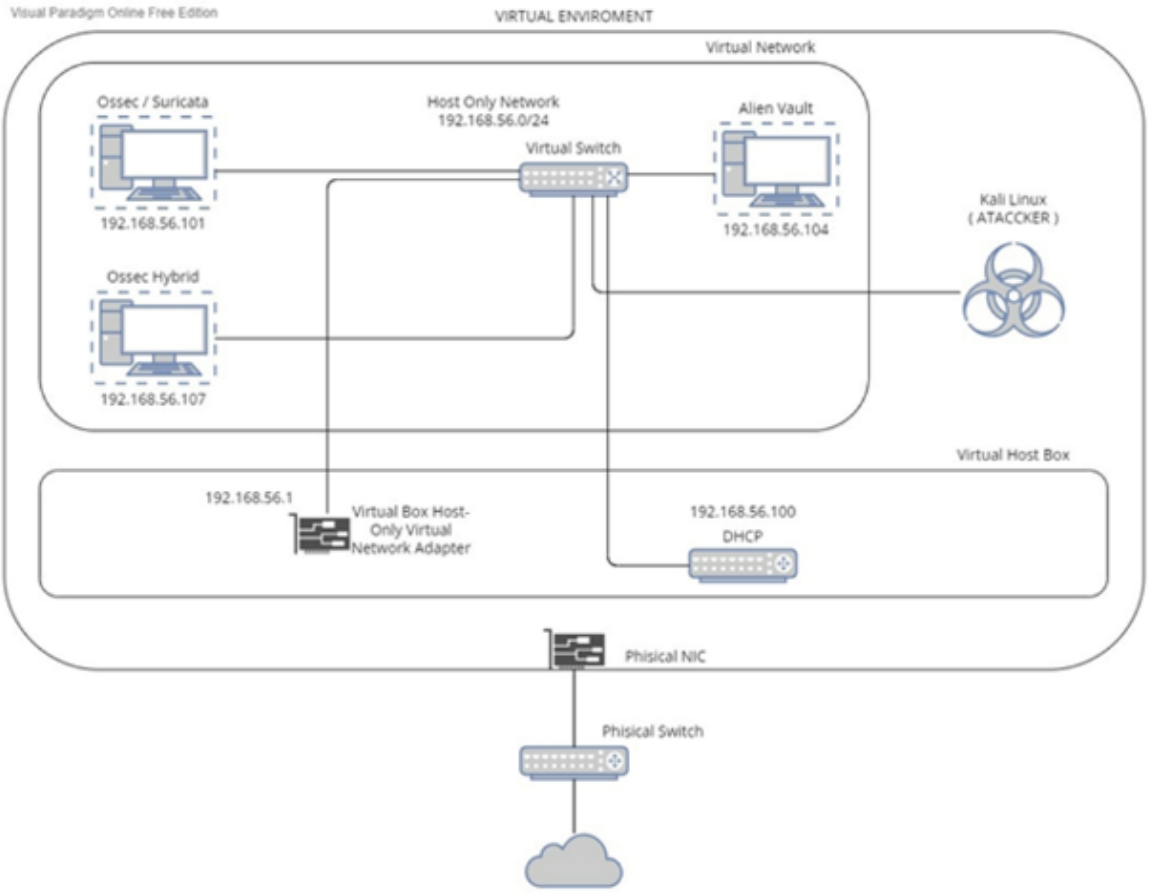

Abbildung 7 Aufbau der virtualisierten Testumgebung

Für die Simulation von Angriffen auf die jeweiligen Systeme wurde die Integration eines weiteren Host Systems umgesetzt. Dieses Hostsystem stellt ein Betriebssystem (Kali Linux) zur Verfügung welches speziell zum Testen von Sicherheitseigenschaften aufgesetzt wurde und mit diversen Programmen für Penetrations- und andere Sicherheitstests ausgestattet ist.

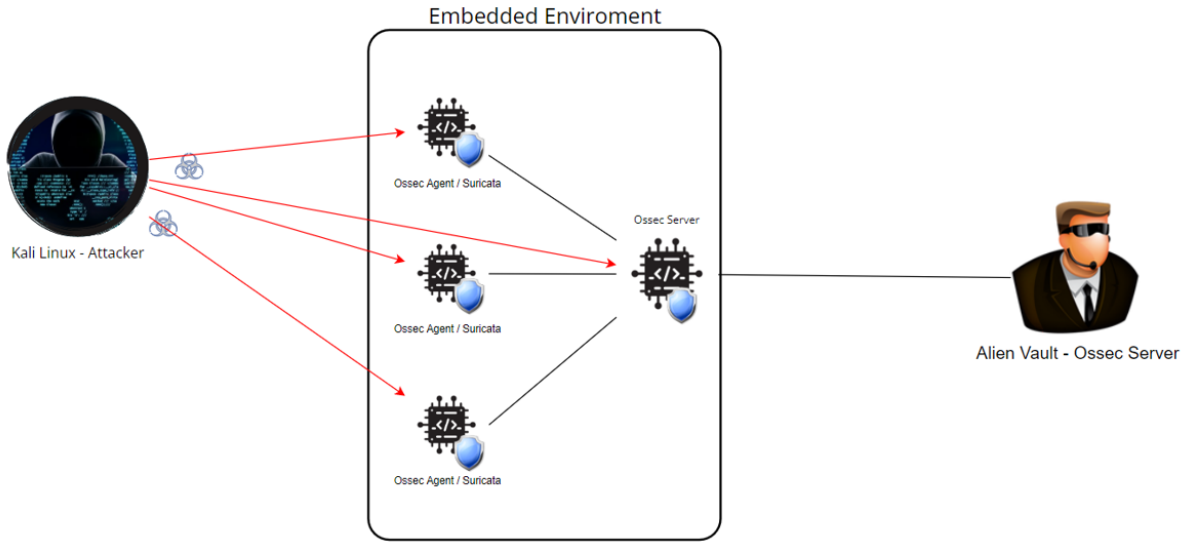

Abbildung 8 Darstellung der Testumgebung im Kontext eines eingebetteten Systems

In Abbildung 8 ist die Testumgebung in Bezug auf das Szenario eines eingebetteten Systems mit integrierten IDS-Lösungen dargestellt. Die eingebettete Umgebung beinhaltet einzelne Systeme, die mit entsprechenden IDS-Lösungen versehen sind. Die IDS-Lösungen werden in einer zentralen Komponente, hier der OSSEC Server, in ihrer Umgebung kumuliert und zur weiteren Auswertung an ein externes System, hier der Alien Vault – OSSEC Server, weitergeleitet. Zusätzlich gibt es einen weiteren Akteur, den Angreifer, welcher die jeweiligen Host Systeme in der eingebetteten Umgebung angreift.

Dieses Szenario bietet eine zwar rudimentäre, aber sehr ähnliche Umgebung eines eingebetteten Systems im Luftfahrtbereich dar. Mit Hilfe dieser virtualisierten Umgebung wurden die Eigenschaften aller beteiligten Komponenten geprüft. Zusätzlich dazu wurden Anforderungen and die IDS-Lösung formuliert, umgesetzt und ausgewertet.

Da immer mehr traditionelle Systeme wie Linux für eingebettete Systeme verwendet werden, wird es in Zukunft wahrscheinlicher, dass bisherige kommerzielle und open source Lösungen aus dem IT-Infrastrukturbereich auch in eingebetteten Systemen häufiger und flächendeckender zum Einsatz kommen.

Die definierten Anforderungen an eine IDS-Lösung eines eingebetteten Systems im Luftfahrtbereich wurden in einem separaten Dokument zusammengefasst. In diesem Dokument wurden auch die Ergebnisse der Evaluation der prototypischen Umsetzung dokumentiert.

Tabelle 3 IDS-Anforderungen und Ergebnisse des Prototyps

Artefakt	Beschreibung
Anforderungen_IDS1.0.docx	IDS-Anforderungen und Ergebnisse der prototypischen Umsetzung

Emulation eingebetteter Systeme

Nachdem die IDS-Tools ausgewählt und ihre wichtigsten Funktionen in der virtualisierten Umgebung getestet wurden, konnte die Modifikation des Prototyps hinzu einer eingebetteten Umgebung umgesetzt werden. Hierfür wurden Möglichkeiten der Emulation von eingebetteten Systemen eruiert und final der Prototyp auf eine QEMU-Virtualisierung angepasst. Mit Hilfe der QEMU-Virtualisierung wurde es möglich spezifische Hardwarekonfigurationen wie bspw. AARCH64- und diverse ARM-Architekturen zu emulieren und die Funktionsfähigkeit des Prototyps auf eingebetteten Systemen zu prüfen.

Mit dieser realistischeren Emulation wurden diverse Eigenschaften der IDS-Lösung eruiert, die in Bezug auf Hardware-Ressourcenverbrauch und notwendiger Systemleistung erste wichtige Indikationen liefern, die in späteren Umsetzungen in Zielsystemen zu berücksichtigen sind. Dieser Aspekt deckt den geplanten Inhalt des Arbeitspaketes AP3.2 Benchmarking ab.

Für diese Umsetzung wurden virtuelle Netzwerke mit TAP-Schnittstellen unter Linux erstellt, die die Kommunikation zwischen verschiedenen Geräten vereinfachen. Die Umsetzung der virtualisierten Umgebung mit TAP-Schnittstellen und dem QEMU-Emulator sind in Abbildung 9 dargestellt.

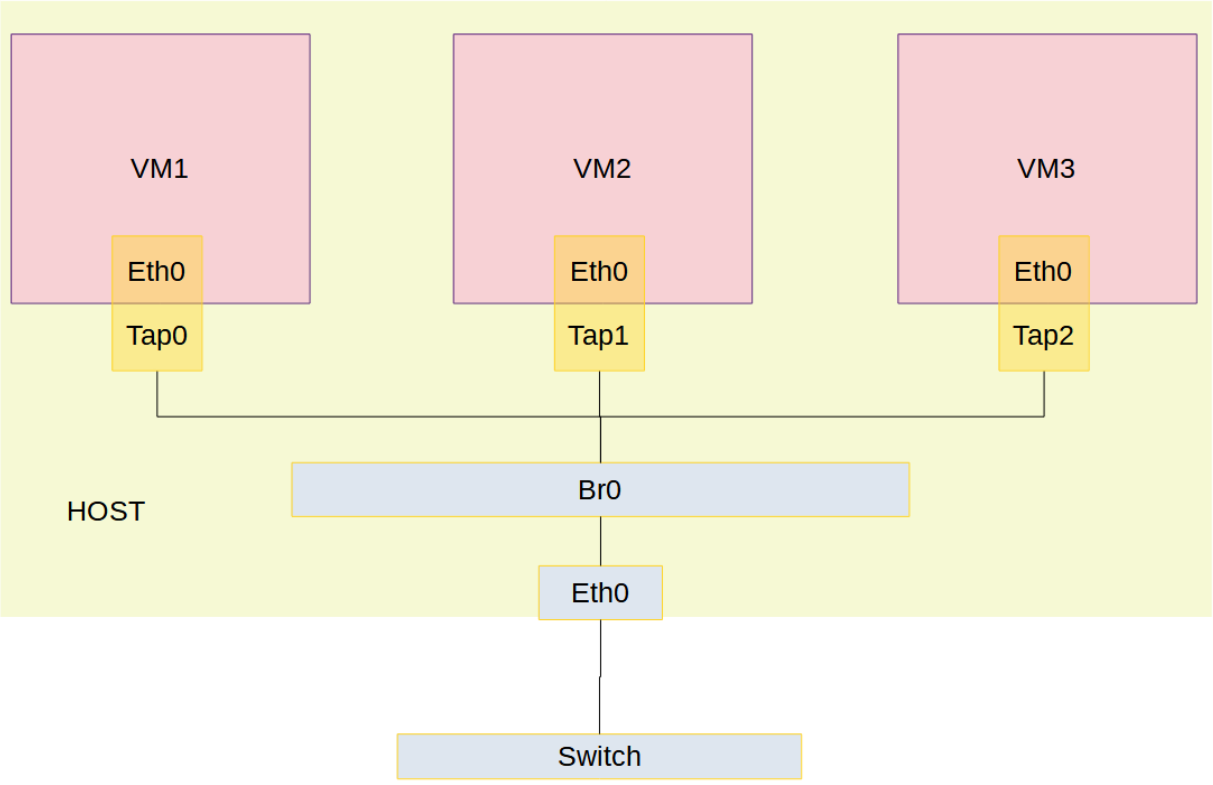

Abbildung 9 Darstellung eines virtuellen Netzes über TAP-Schnittstellen

Die notwendigen Schritte zur Umsetzung der Virtualisierungsumgebung mittels TAP-Schnittstellten wurden separat dokumentiert.

Tabelle 4 Artefakte zum Aufsetzen der Virtualisierungsumgebung mittels TAP-Schnittstellen

Artefakt	Beschreibung
Skript_TAP1.0.docx	Umsetzung der Virtualisierungsumgebung mittels TAP-Schnittstellen

Die jeweilig verwendeten Skripte bieten eine einfache Möglichkeit der Wiederverwendung und sollen für spätere Arbeiten und Umsetzungen die Baseline darstellen. Beispielhaft ist in Abbildung 10 ein Skript-Aufruf zur Virtualisierung in QEMU mittels TAP-Schnittstelle dargestellt.

```
qemu-system-aarch64 -M virt -cpu cortex-a53 -nographic -m 1G -initrd initrd.img-5.10.0-16-arm64 \
-kernel vmlinuz-5.10.0-16-arm64 -append "root=/dev/vda2 console=ttyAMA0" \
-drive if=virtio,file=debian-3607-aarch64.qcow2,format=qcow2,id=hd \
-device e1000,netdev=vm0,mac=52:54:00:12:34:50 \
-netdev tap,id=vm0,ifname=tap0,script=no,downscript=no \
-device virtio-rng-pci -device VGA,vgamem_mb=64,edid=on
```

Abbildung 10 Beispiel für einen Befehl in QEMU zur Durchführung einer Emulation über die TAP-Schnittstelle

Diese Umsetzung stellt die Grundlage zur Emulation einer vernetzten Umgebung dar. Im späteren kann dann die Ausführung und das Installieren von Betriebssystemen und den entsprechenden IDS-Lösungen erfolgen. Diese wurden ebenfalls detailliert in einer separaten Beschreibung zusammengefasst.

Tabelle 5 OSSEC-QEMU Installationsmanual

Artefakt	Beschreibung
IDS_QEMU1.0.docx	OSSEC-QEMU-Installationsmanual

Anwendung von IDS-Lösungen im Luftfahrtbereich

Im folgenden Abschnitt wird beschrieben, welche Schritte im Rahmen des Vorhabens durchgeführt wurden, um die gesammelten Erkenntnisse für IDS-Lösungen in einem Anwendungsfall zu eruieren.

Als Anwendungsfall für den Luftfahrtbereich wurde die Umsetzung der IDS-Lösungen für Drohnen geprüft. Dazu wurde die PixHawk PX4-Plattform verwendet.

PX4-Plattform

PX4 ist ein leistungsstarker Open-Source-Autopilot-Stack.

Einige der wichtigsten Merkmale des PX4 sind:

- Steuert viele verschiedene Fahrzeugstrukturen/-typen, darunter: Flugzeuge (Multicopter, Starrflügler und VTOLs), Landfahrzeuge und Unterwasserfahrzeuge.
- Große Auswahl an Hardware für Fahrzeugsteuerung, Sensoren und andere Peripheriegeräte.
- Flexible und leistungsstarke Flugmodi und Sicherheitsfunktionen.

PX4 ist zentraler Bestandteil einer umfassenderen Plattform, die die QGroundControl-Bodenstation, die Pixhawk-Hardware und das MAVSDK zur Integration mit Computern, Kameras und anderer Begleithardware über das MAVLink-Protokoll umfasst. Der PX4 wird vom Dronecode-Projekt unterstützt.

PX4 kann auf einer Vielzahl von Plattformen laufen, von denen Pixhawk vielleicht die bekannteste ist. Das Herzstück des Pixhawk ist ein ARM Cortex M4-Prozessor, der die Schätzung und Flugsteuerung einer Drohne übernimmt. Auf dem ARM Cortex M4 läuft ein Echtzeitbetriebssystem namens Nuttx. Das PX4-System baut auf der Nuttx-Anwendungsschicht auf. In Abbildung 11 ist PX4 Controller beispielhaft abgebildet.

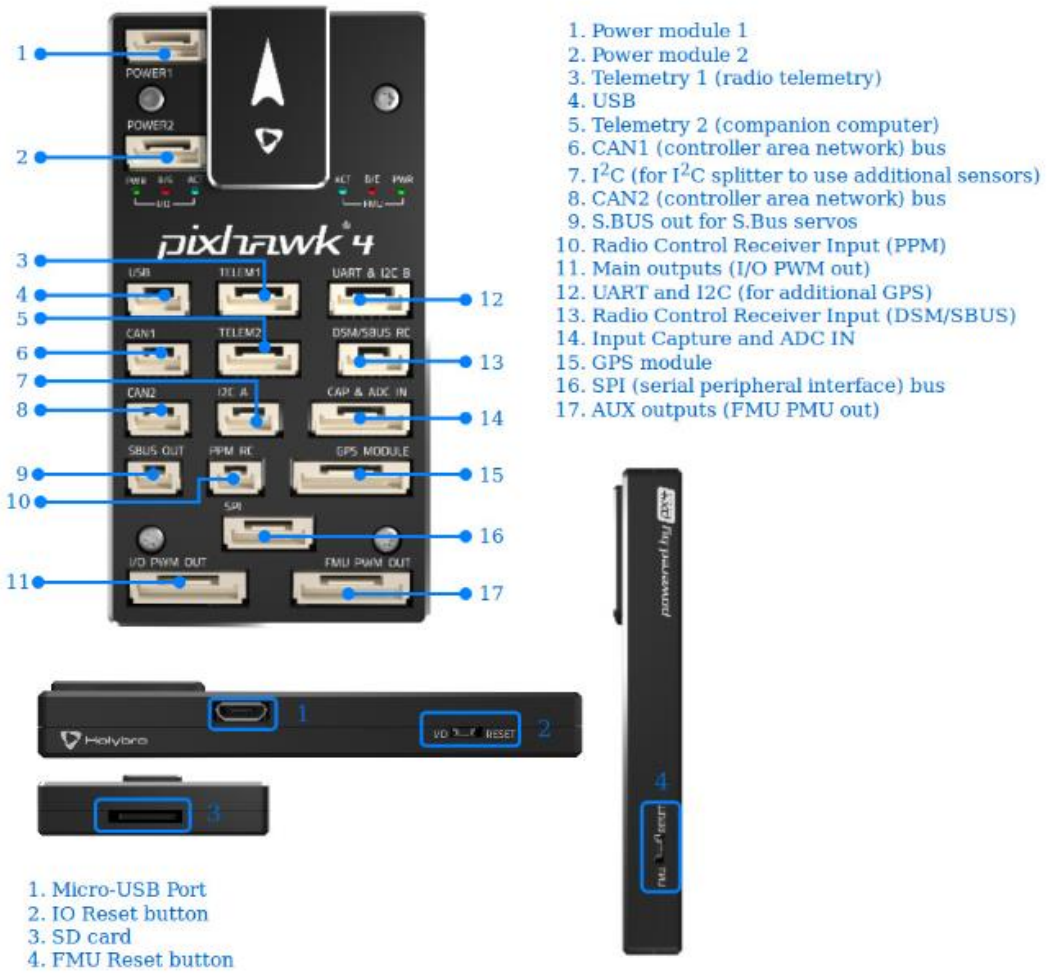

Abbildung 11 Abbildung eines PX4-Controllers

Mit Hilfe der PX4 Systems gibt es unterschiedliche Möglichkeiten der Umsetzung von Drohnenprojekten. In Abbildung 12 folgenden ist die Umsetzung eines einfachen Drohnenprojektes mit dem PX4-System dargestellt. Dieses beinhaltet dann lediglich eine Flugsteuerung ohne direkte Möglichkeiten der Kommunikation mit weiteren Drohnen im Verbund.

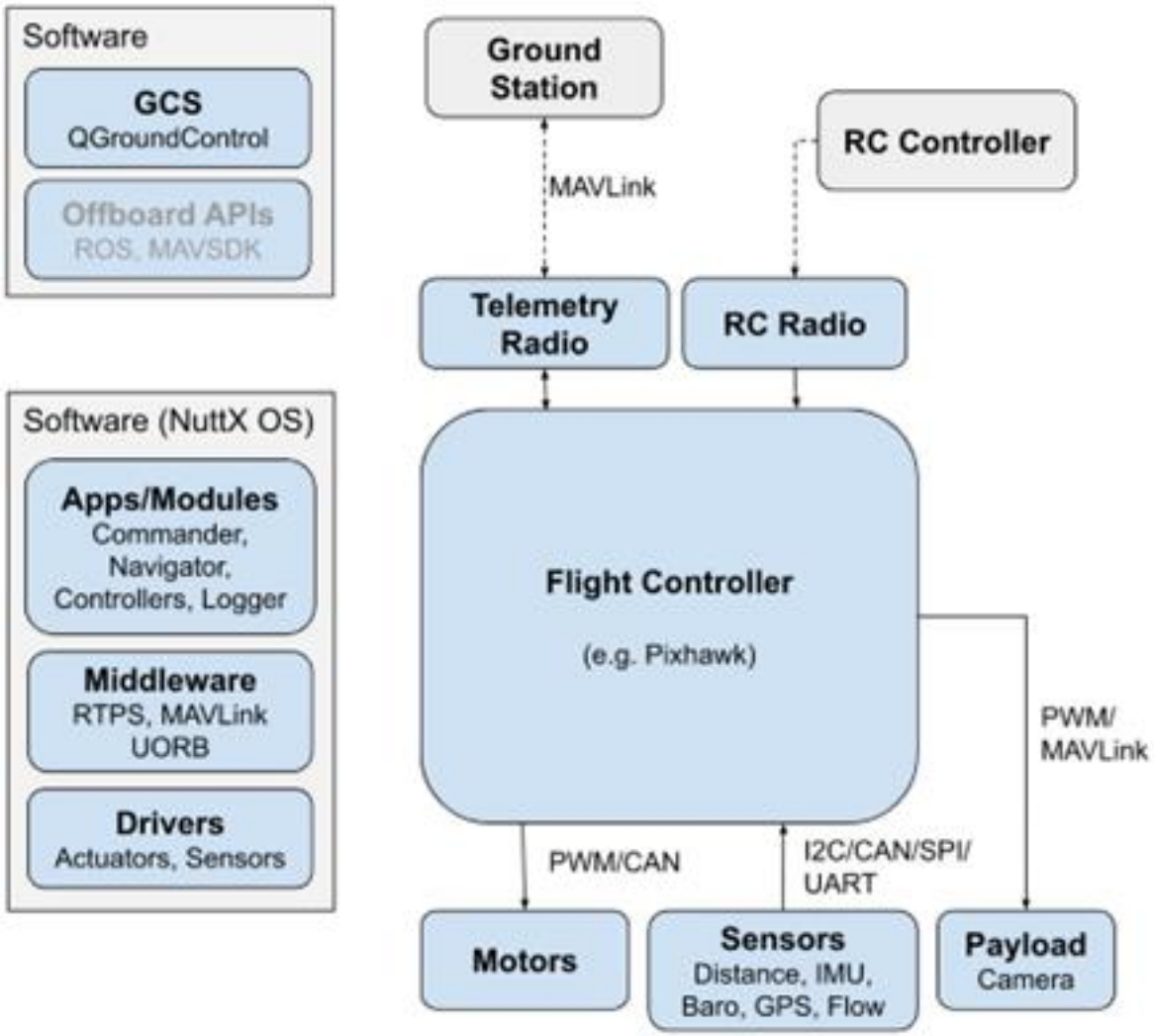

Abbildung 12 Architektur nur mit Flugsteuerung https://docs.px4.io/

Eine weitere Konfigurationsmöglichkeit ist in Abbildung 13 dargestellt. Hierbei handelt es sich um ein Setup, welches zusätzlich zur Flugsteuerung auch einen Begleitcomputer (bzw. Mission Computer) beinhaltet. Dieser Mission Computer ermöglicht eine umfassendere Verwendung von weiterer Peripherie. Zusätzlich dazu stellt dieses Setup eine Ideale Grundvoraussetzung für ein System dar, welches um IDS-Komponenten erweitert werden kann, da der Mission Computer beliebig ausgelegt werden kann und damit auch die Systemvoraussetzung für IDS-Lösungen erfüllen kann.

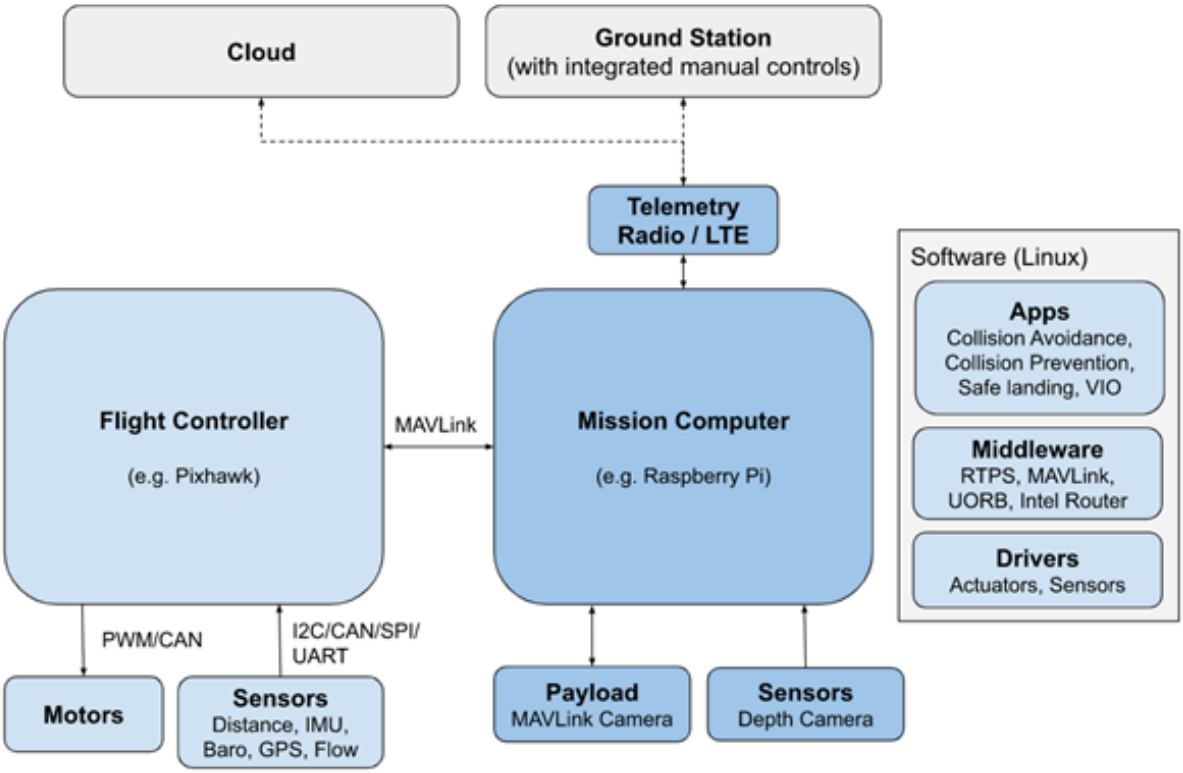

Abbildung 13 Architektur mit Flugsteuerung und Begleitcomputer https://docs.px4.io/

Aufgrund der Eigenschaften der Option mit Begleitcomputer wurde diese ausgewählt und für die weiteren Untersuchungen verwendet.

Die Architektur in Abbildung 13 stellt Konzepte für eine sich selbst verwaltende Drohne vor, bei der die gesamte Steuerung von einem Begleitcomputer mit Linux erfolgt, auf dem IDS-Lösungen konfiguriert werden können die auch im bisher umgesetzten virtualisierten Prototyp genutzt wurden.

Auf dem Flugcontroller läuft der normale PX4-Flugstack, während der Begleitcomputer erweiterte Funktionen wie Objekt- und Kollisionsvermeidung bietet. Die beiden Systeme sind über eine schnelle serielle oder IP-Verbindung miteinander verbunden und kommunizieren normalerweise über das MAVLink-Protokoll. Die Kommunikation mit Bodenstationen und der Cloud erfolgt in der Regel über den Begleitcomputer (z. B. über den MAVLink Router). MAVLink ist ein sehr leichtgewichtiges Nachrichtenprotokoll für die Kommunikation mit Drohnen (und zwischen Drohnen-Bordkomponenten).

Auf PX4-Systemen läuft in der Regel ein Linux-Betriebssystem auf dem Begleitcomputer (da das PX4/PX4-Avoidance-Projekt ROS-basierte Evader-Bibliotheken für Linux bereitstellt). Linux ist eine viel bessere Plattform für die "allgemeine" Softwareentwicklung als NuttX; es gibt viel mehr Linux-Entwickler und eine Menge nützlicher Software wurde bereits geschrieben (z. B. für Computer Vision, Kommunikation, Cloud-Integrationen, Hardware-Treiber).

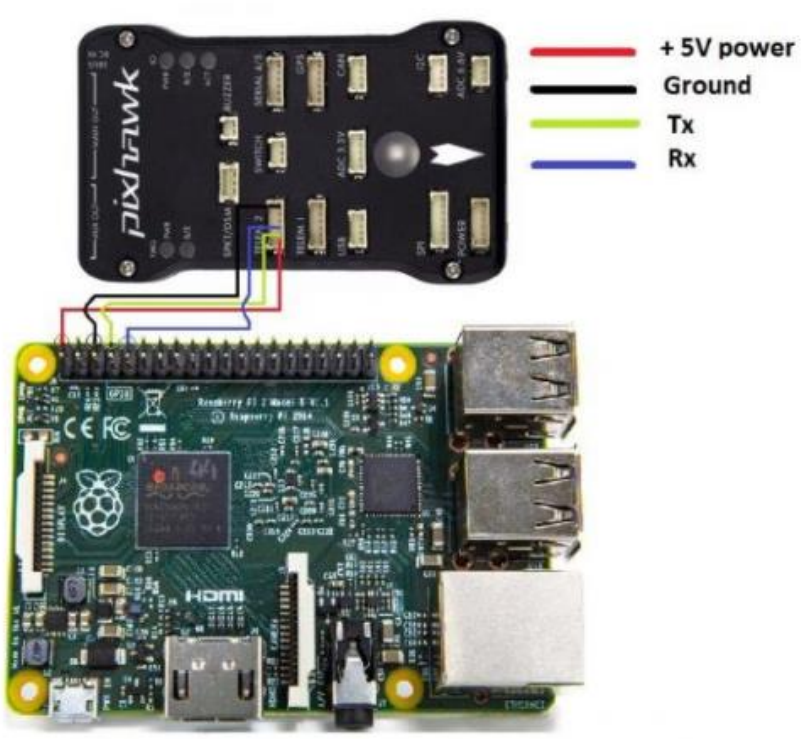

Abbildung 14 Verbindung zwischen PX4 und Companion Computer (UART)

Der Begleitcomputer befindet sich ebenfalls in der Drohne und kommuniziert mit dem Autopiloten, um diesen beispielsweise zu steuern. Der Begleitcomputer empfängt alle vom Autopiloten erzeugten MAVLink-Daten (einschließlich GPS-Daten) und kann sie nutzen, um während des Fluges intelligente Entscheidungen zu treffen. Dies ermöglicht eine breite Palette von Funktionen, von computergesteuerten Trajektorien bis hin zu sehr rechenintensiven Funktionen wie der Bildverarbeitung. Alle über MAVLINK übertragenen Daten können verarbeitet werden.

In Abbildung 14 ist das Setup von PX4 Controller mit Begleitcomputer abgebildet. Um eine Kommunikation zu ermöglichen, wird das MAVLink Protokoll benötigt und auf dem Begleitcomputer muss eine entsprechende Software installiert sein, die mit dem seriellen Anschluss kommuniziert.

In unserem Szenario wurde eine Ubuntu Linux und Debian Linux ARM/AARCH64 Distribution für den Begleitcomputer verwendet. Diese sind in der Lage die im Vorfeld beschriebenen IDS-Lösungen auszuführen und zu nutzen. Somit war eine erfolgreiche Umsetzung der virtualisierten Umgebung in eine physische möglich.

Nach der erfolgreichen Umsetzung der IDS-Lösung in einem Prototyp wurden verschiedene Angriffsszenarien auf Drohnen kategorisiert und die Möglichkeit Erkennung mittels IDS-Lösungen geprüft. Eine Zusammenfassung findet sich in Abbildung 15.

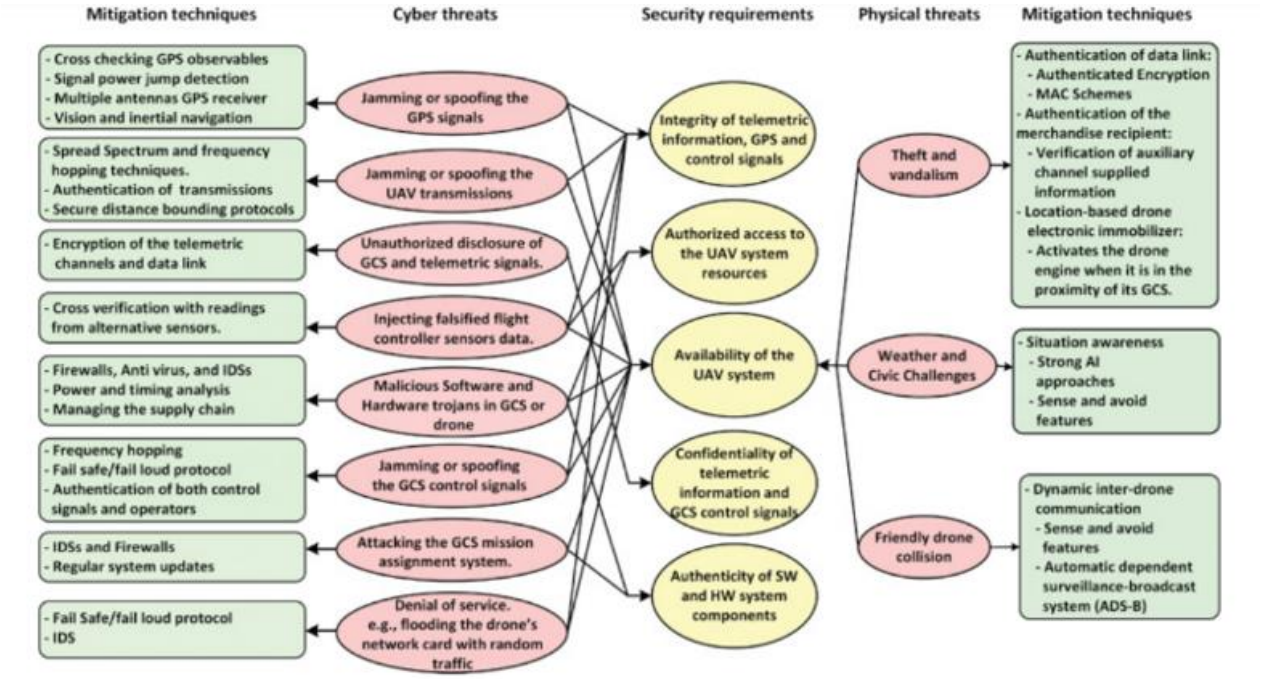

Abbildung 15 Misuse Cases für Drohnen und entsprechende Gegenmaßnahmen

In fast allen vorgestellten Szenarien haben wir die Möglichkeit, Bedrohungen mit IDS-Lösungen zu erkennen, da alle Informationen über die MAVLINK-Nachrichten verfügbar sind und jede Nachricht nach eigenem Ermessen behandeln werden kann.

3.1.2.3 Synergieeffekte Safety & Security

In diesem Abschnitt werden die wichtigsten Erkenntnisse des Projektes vorgestellt, die sich mit dem Vorhabenziel der Ausnutzung von Synergieeffekten zwischen Safety und Security beschäftigten.

Dazu wurden vor allem relevante Methoden und Herangehensweisen aus beiden Bereichen untersucht, um Schnittmenge zu identifizieren, welche im praktischen Arbeitsablauf Effizienzsteigerungen mit sich bringen.

Dieses gesteckte Ziel wurde im Arbeitspaket 4.1 Safety & Security bearbeitet.

Die notwendigen Analyseschritte für Safety und Security Betrachtungen wurden im Rahmen dieses Arbeitspaketes detaillierter untersucht. Dazu zählen zum einen die FMEA(Failure Mode and Effects Analysis) in Abbildung 16 aus dem Safety Bereich und als Pendant die TARA (Threat Analysis and Risk Assessment) aus dem Security Bereich in Abbildung 17.

FMEA

Produktname		Änderungsstand	
Produktnummer		erstellt durch	
Modell/System/Fertigung		verantwortlich	
		Datum	

Element/ Funktion	n°	mögliche Fehler / Fehlfunktion	Fehlerursache	Fehlerfolge	Derzeitiger Zustand: Vermeidungs-Maßnahme	Entdeckungs-Maßnahmen	Bedeutung 1-10	Auftreten 1-10	Entdeckung 1-10	Risikoprioritäts-zahl	empfohlene Maßnahme	Zuständigkeit	Verbesserter Zustand: getroffene Maßnahme	Bedeutung 1-10	Auftreten 1-10	Entdeckung 1-10	Risikoprioritäts-zahl
Motorwelle	1	Bruch der Welle	Belastung nicht korrekt erkannt	Totalausfall	Standardtests	keine	8	6	10	480	Belastung durch geeigneten Berechnungsansatz	Herr Mustermann	neuer überarbeiteter Belastungstest	8	2	7	112
	2	Spiel in der Lageranordnung	unexakte Funktionserfüllung	Lockern der Lagermutter im Betrieb	Befestigung mit Drehmomentschlüssel	keine	3	8	10	240	Zusätzliche Sicherung der Lagermutter	Herr Mustermann	Konterung der Mutter mit zweiter Mutter	1	8	10	80
	3																
	4																
	5																

Legende

Auftreten (Wahrscheinlichkeit des Fehlers (Fehler ist möglich))		Bedeutung (des Fehler (Auswirkung auf den Kunden))		Entdeckung (Wahrscheinlichkeit der Entdeckung des Fehlers (vor Auslieferung an	
unwahrscheinlich	1	kaum wahrnehmbar	1	hoch	1
sehr gering	2-3	unbedeutender Fehler	2-3	mäßig	2-3
gering	4-6	mäßig schwerer Fehler	4-6	gering	4-6
mäßig	7-8	schwerer Fehler	7-8	sehr gering	7-8
hoch	9-10	äußerst schwer. Fehler	9-10	unwahrscheinlich	9-10

Abbildung 16 FMEA Tabelle mit Auftritts- und Entdeckungswahrscheinlichkeit sowie Bedeutung des Fehlers

TextID	Damage Potential	Description	Value
SGE	Very low		1
GER	Low		2
MOD	Moderate		3
HOH	High		4
SHO	Very high		5

Risk Factor Category	Risk Factor	Description	TextID	Value	ID	Einträge
Expertise	Layman	Typical vehicle owner / driver	LAY	0	1	4
	Proficient	Technically interested driver / garage staff	PRO	3	2	
	Expert	Garage staff with special expertise (<20%) (e.g., installation of corrupted smart cards)	EXP	6	3	
	Specialist	Highly qualified person (<1%) (z.B. side-channel-attacks, cryptanalysis)	SPC	8	4	
Access	Remote access	Remote access without the need for direct access to a vehicle	REM	0	5	4
	Easy	Simple direct access to the vehicle	EAS	1	6	
	Moderate	Complex access to vehicle parts required (e.g., access to flash memory inside an ECU)	MOD	4	7	
	Hard	Access on micro-electronic level (e.g., voltage or current measurement)	HAR	10	8	
Elapsed time	Hours		HOU	0	9	5
	Days		DAY	1	10	
	Weeks		WEE	3	11	
	Months		MON	7	12	
	Decades		DEC	35	13	
Equipment	Standard	Common IT-equipment (e.g., notebook, freely available OBD diagnosis-tools)	STA	0	14	4
	Specialized	Professional garage-equipment (CAN-cards, diagnosis-equipment)	SPE	4	15	
	Bespoke	electron microscope))	BES	7	16	
	Multiple bespoke	Multiple bespoke tools	MBE	9	17	
Knowledge of SUD	Public	Public information	PUB	0	18	4
	Internal	Internal information	INT	3	19	
	Classified	Classified information: information leak can endanger product- or project-goals	CLA	7	20	
	Confidential	Confidential information: critical to the enterprise	CON	11	21	

Risk levels	Number	TextID	Threshold	Required AP	TextID	Number
Low risk	1	VNL	0	Very low	ENF	1
Moderate risk	2	TOL	14	Low	FOR	2
High risk	3	ERN	20	Moderate	SWE	3
Very high risk	4	HOH	25	High	BSW	4
			35	Beyond high	NIM	5

Risk Table
Damage Potential x Required Attack Potential

	Beyond high	High	Moderate	Low	Very low
Very low	Low risk	Low risk	Low risk	Moderate risk	Moderate risk
Low	Low risk	Low risk	Moderate risk	High risk	High risk
Moderate	Low risk	Moderate risk	High risk	High risk	Very high risk
High	Moderate risk	High risk	High risk	Very high risk	Very high risk
Very high	Moderate risk	High risk	Very high risk	Very high risk	Very high risk

Abbildung 17 Übersicht der Klassifizierung einer TARA

In den anschließenden Arbeiten wurden zunächst bestehende Abhängigkeiten der beiden Prozessschritte identifiziert. Grundsätzlich kann man festhalten, dass es Abhängigkeiten in beide Richtungen gibt.

Beispiele:

- [Safety -> Security] Eine als sicherheitskritisch (safety) eingestufte Funktion im System sollte zwingend eine tiefergehende Betrachtung aus Security Sicht anstoßen. Eine

Safety-Betrachtung berücksichtigt in den üblichen Herangehensweisen typischerweise keine externe unkooperative Aktoren sondern.

- [Security -> Safety] Eine detaillierte Bedrohungsmodellierung (siehe Abbildung 3) im Security Prozessschritt führt häufig dazu, dass für unterschiedliche Angriffsszenarien verschiedene Systemzustände entstehen. Hier ist es für Security Experten äußerst schwierig die genauen Auswirkungen dieser Systemzustände auf Sicherheitseigenschaften (safety) zu bewerten und vor allem Auswirkungen und die damit verbundenen Schadenshöhe ausreichend gut zu kategorisieren. Häufig muss nach einer Risikobewertung aus Security Sicht eben auch eine erneute Betrachtung durch Safety Experten erfolgen.

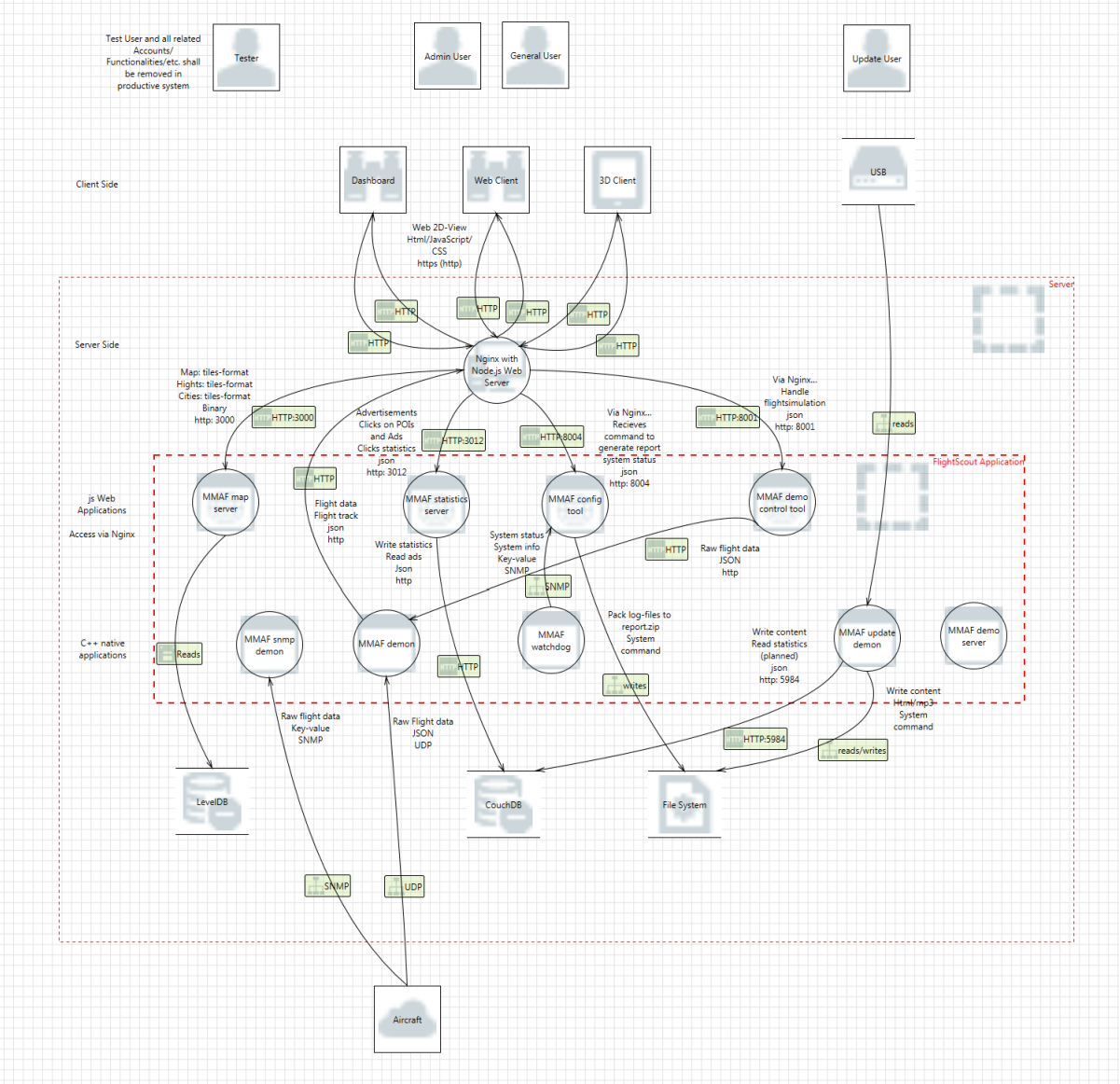

Abbildung 18 Bedrohungsmodellierung eines komplexen Systems

3.2 Wichtigste Positionen des zahlenmäßigen Nachweises

Die Zuwendung wurde hauptsächlich zur Deckung der Personalkosten verwendet. Demzufolge ist dies auch die wichtigste Position im zahlenmäßigen Nachweis.

Ausgaben für Material wurden nicht getätigt. Unterbeauftragungen im Rahmen des Projektes gab es keine.

Für weitere Details zum Thema zahlenmäßiger Nachweis siehe das Dokument zahlenmäßiger Nachweis gem. Nr.19.3 NKBF 98.

3.3 Notwendigkeit und Angemessenheit der geleisteten Arbeit

Die geleistete Arbeit war notwendig, um Kenntnisse und Know-how aufzubauen, die dargestellten Ergebnisse zu erzielen, sowie die Grundlagen und Kriterien für strategische Entscheidungen zum Projekt zu erarbeiten. Ressourcen wurden entsprechend der Planung eingesetzt. Etwaige Abweichungen wurden in den Zwischenberichten dem Projektträger angezeigt.

3.4 Voraussichtlicher Nutzen, Verwertbarkeit des Ergebnisses

Die größten Verwertungsmöglichkeiten der Ergebnisse sehen wir in unserem Kerngeschäften bei der Durchführung von Sicherheits- und Zuverlässigkeitsanalysen und der Erstellung von Testprozessen und dem Testen von sicherheitskritischer Software. Das gewonnene Know-how bei der Durchführung der entwickelten Prozesse können wir in unseren Hauptindustriebereichen Luftfahrt, Automotive und Medizintechnik übernehmen. Weiterhin wollen wir in eventuellen Folgeprojekten Sicherheitsanalysen, Verifikationsstrategien, und Testdurchführung aufbauend auf den entwickelten Prozessen anbieten. Im Folgenden werden der Nutzen und die Verwertbarkeit der Ergebnisse anhand der drei Kernarbeitspakete beschrieben.

3.4.1 Erstellung von Angeboten

Da die Partner des Projektes Ihre Prozesse zumindest bis zu einem gewissen Grad einheitlicher gestalten und im Rahmen des Projektes Vorgehensweisen beschrieben und konsolidiert wurden fällt es einem Dienstleistungsunternehmen wie Philotech nun leichter Angebote abzuschätzen und dadurch existiert eine höhere Wahrscheinlichkeit für Projektzuschläge. Vor allem die im Forschungsprojekt betrachteten Technologien und ihr sicherer sowie zulassungsfähiger Einsatz stellt einen enormen Mehrwert bei Abschätzung von Aufwänden und notwendigen Fähigkeiten für Projekte dar.

3.4.2 Beratung

Auf Basis der Erkenntnisse und Erfahrungen im IDEA-Projekt konnten die Seminare von Philotech ergänzt und verbessert werden. Weiterhin ist es durch Kenntnis der Best Practices und durch Kenntnis der unterschiedlichen Varianten vorzugehen möglich, für Beratungsprojekte das Wissen anzuwenden und Kunden besser hinsichtlich möglicher Umsetzungsvarianten zu Beraten. Dies gilt vor allem für die im Projekt gesetzten Schwerpunktthemen.

3.4.3 Security Anforderungen für Software und Systeme

Durch die zunehmende Vernetzung von technischen Systemen steigt die Anfälligkeit technischer Systeme gegenüber mutwilligen Angriffen abseits des physikalischen Zugangs. Neue Regulierungen sind bereits auf dem Weg, welche zusätzliche Maßnahmen in System und Softwareentwicklung erfordern, um diesem Risiko adäquat entgegenzutreten.

Mit den in diesem Projekt gewonnenen Erkenntnissen bei der Umsetzung von regulatorischen Vorgaben bezüglich Security können wir uns als Philotech weiterhin mit Erfolg an neuen Ausschreibungen und Projekten bewerben. Diese Maßnahme sichert uns Arbeitsplätze am Standort Cottbus, München und Hamburg und hilft uns bei der Schaffung neuer Arbeitsplätze.

Im Jahr 2023 konnten durch die gewonnenen Erkenntnisse bereits eine Ausschreibung im Bereich Risikomodellierung für Luftfahrtsysteme für den Standort Cottbus gewonnen werden.

3.5 Fortschritt auf dem Gebiet bei anderen Stellen

Aufgrund des Fassettenreichtums dieses Vorhabens ist ein vollständiger Überblick über Fortschritte einzelner Aspekte nicht möglich. Dennoch wurden für einzelne Aspekte Fortschritte entdeckt, welche entsprechend in den einzelnen Arbeitspaketen berücksichtigt wurden.

3.6 Veröffentlichungen des Ergebnisses

Während der Projektlaufzeit wurden keine Veröffentlichungen publiziert.

Abbildungsverzeichnis

Tabellenverzeichnis