BMBF Verbundprojekt

DIAMONDS

Projektpartner

Dornier Consulting, Fraunhofer FOKUS,
Giesecke & Devrient, Testing Technologies

Schlussbericht

für die Projektlaufzeit vom 1.10.2010 bis zum 30.06.2013

Das diesem Bericht zugrunde liegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter den Förderkennzeichen 01 IS 100 31A, 01 IS 100 31B, 01 IS 100 31C, 01 IS 100 31D gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt bei den Autoren.

Ina Schieferdecker, Jürgen Großmann, Martin Schneider, Johannes Viehmann, FhG FOKUS;

Stephan Pietsch, Testing Technologies IST GmbH; Andrej Pietschker, Giesecke & Devrient;

Felix Jakob, Andreas Schulze, Dornier Consulting

Inhaltsverzeichnis

ABBILDUNGEN

TABELLEN

HISTORIE

Vers.	Datum	Autor	Beschreibung
1.0	15.12.2013	DIAMONDS	Abschlussbericht
1.1	07.03.2013	DIAMONDS	Ergänzungen

Zusammenfassung

Wachsende Sicherheitsbedenken von Endbenutzern, Anbietern und Regulierungsbehörden, gemeinsam mit der tatsächlich steigenden Bedrohung durch Hacker-Angriffe auf vernetzte Systeme, stärken die Nachfrage nach Methoden und Werkzeugen, mit denen sich IT-Sicherheit prüfen und belegen lässt. Das ITEA-geförderte Forschungsprojekt DIAMONDS (www.itea2-diamonds.org) entwickelte systematische, modellbasierte Test- und Überwachungsansätze für die Prüfung von IT-Sicherheitseigenschaften in softwaregesteuerten, vernetzten Systemen. Ziel des Projekts war es, die Sicherheit von Systemen und Anwendungen durch eine systematische Prüfung von IT-Sicherheitseigenschaften zu erhöhen und gleichzeitig über den Weg der Testautomatisierung und des modellbasierten Testens kosteneffiziente Lösungen zu entwickeln.

Im europäischen DIAMONDS-Projekt bündelten sich ergänzende interdisziplinäre Expertisen und Technologien der Partner aus Österreich, Finnland, Frankreich, Deutschland, Luxemburg und Norwegen, die gemeinsam ein weitverzweigtes Kompetenznetzwerk durch Partnerschaften mit Industrie und Forschung geschaffen haben. Die Projektergebnisse wurden entlang von acht Fallstudien aus den Anwendungsbereichen Automobilelektronik, Smart Cards, Bankwesen, Telekommunikation, Industrieautomation und verteilte Funknetzwerke entwickelt und validiert. Zu den innovativen Ergebnissen des Projekts zählen bereits heute Techniken für den modellbasierten Robustheitstest durch Smart Behavioural Fuzzing, Techniken für den modellbasierten Passivtest, die Dokumentation und Wiederverwendung von Know-how in Form von Security Test Pattern und eine Methodik für den risikobasierten IT-Sicherheitstest. Die Überführung von ausgesuchten Projektergebnissen in Standardisierungsaktivitäten bei der ETSI (MTS-SIG, ISG-ISI) sorgen für eine nachhaltige Konsolidierung und Verfügbarkeit der Projektergebnisse. Direkte Anschlussprojekte, wie z.B. das FP7-Projekt RASEN (Compositional Risk Assessment and Security Testing of Networked Systems) erlauben darüber hinaus den gezielten Ausbau interessanter und vielversprechender Forschungsschwerpunkte.

Dieser Schlussbericht dokumentiert die Ergebnisse des deutschen DIAMONDS-Projekts, das bestehend aus den Partnern Fraunhofer FOKUS, Giesecke & Devrient, Dornier Consulting und Testing Technologies Fallstudien aus den Bereichen des Bankwesens und Automobilindustrie bearbeitet hat. Das deutsche DIAMONDS-Projekt ist in das europäische DIAMONDS-Projekt integriert gewesen.

1. AUFGABENSTELLUNG

Das Ziel von DIAMONDS war es, systematische, modellbasierte Prüf- und Überwachungsansätze für Sicherheitsprüfungen zu entwickeln, um durch eine frühzeitige Prüfung und Testautomatisierung hochsicherer Systeme zu ermöglichen. Fortschrittliche modellbasierte Sicherheitsprüfmethoden erlauben eine frühzeitige Identifizierung von Implementierungs- und Designschwächen sowie effiziente System- und Prüfdesigns mit dem Ziel hoher Systemsicherheit. Die DIAMONDS-Sicherheitsprüfmethodik sollte auf verschiedene Multi-Domänen-Sicherheitsstandards anpassbar sein und die Generierung von Risikoanalyseprüfungen sowie die Risikoabschätzung durch Auswertung der Prüfergebnisse erlauben.

DIAMONDS sollte eine gut erkennbare europäische Sicherheitsprüfmethodik industriellen Ausmaßes entwickeln, die sich bereits als erfolgreich für sicherheitskritische Systeme in verschiedenen Anwendungsdomänen erwiesen hat. Das Projekt basierte auf den sich ergänzenden interdisziplinären Expertisen und Technologien der Partner aus Österreich, Finnland, Frankreich, Deutschland, Luxemburg, Norwegen und Spanien, die gemeinsam ein weitverzweigtes Kompetenznetzwerk durch Partnerschaften mit Industrie und Forschung geschaffen haben und sich dieses Problems annehmen: Domänenexpertise, vernetzte Systeme und -dienste, Sicherheitstechnik, Prüfinfrastrukturen und modellbasierte Prüfungen.

DIAMONDS zielte auf den steigenden Bedarf an systematischen Sicherheitsprüfmethoden, indem es Techniken und Werkzeuge entwickelt, die effizient zur Absicherung von vernetzten Anwendungen in verschiedenen Domänen eingesetzt werden können.

DIAMONDS Hauptaufgabe war es, Innovationen in vier Bereich der Sicherheitsprüfmethoden und Sicherheitstechnologien einzuführen. Diese Innovationen zielen darauf ab, eine Vornorm für modellbasierte Sicherheitsprüfungen in heterogenen und verteilten Systemen und Diensten zu schaffen, und stellen gleichzeitig die für eine Einführung formeller Sicherheitsprüfungen notwendige Technologie in der Industrie dar:

- **Fortschrittliche modellbasierte Sicherheitsprüfmethoden**, die verschiedene Techniken miteinander kombinieren, um so bessere Ergebnisse hinsichtlich Multi-Domänen-Sicherheit zu erzielen.
- **Entwicklung autonomer Prüftechniken basierend auf automatisierten Überwachungstechniken**, um die Belastbarkeit sich dynamisch entwickelnder Systeme zu erhöhen.
- **Vornormung von Multi-Domänen-Sicherheitsprüfmethoden und Testmustern**, die es DIAMONDS ermöglicht, vollständig kompatible Sicherheitsprüftechniken und -werkzeuge anzubieten.
- **Open-Source-Plattform für die Integration von Sicherheitsprüfprogrammen**, um eine einheitliche Plattform zu schaffen, welche dem Anwender jeweils eine einzige Benutzeroberfläche für die Werkzeuge sowie präzise Berichte der verschiedenen Prüfprogramme zur Verfügung stellt.

Diese Innovationen, die mit DIAMONDS Einzug halten, können dann von mehreren Interessengruppen genutzt werden:

- **Entwickler** profitieren von der Prüfung der Software auf Schwachstellen und von der Verhinderung der Einschleppung dieser **Schwachstellen** in den Software-Lebenszyklus.
- **Systemintegratoren, Prüfer, Software-Qualitätssicherer und Software-Einkäufer** können die Qualität einer Software bereits vor deren Einsatz abschätzen. DIAMONDS stellt eine unabhängige Metrik für seine **Aktivitäten** zur Verfügung und gewährleistet auf diese Weise sichere Abläufe.
- **Forscher** werden in der Lage sein, die neuen Erkenntnisse in der ICT zu untersuchen und zu etablieren.

Die drei wichtigsten Aufgaben von DIAMONDS bestanden in der Entwicklung von:

- **Definition von Vornormen für Modelle und Prüfmuster.** Diese Modelle und Muster wurden so entwickelt, dass sie von verschiedenen Werkzeugen in verschiedenen Domänen genutzt werden können und so ihren Einfluss auf die Software Systems Community erhöhen.
- **Arbeitsergebnis Sicherheitsprüfmethodik.** Dieses Dokument wird die Lernkurve verbessern und die Übernahme der DIAMONDS-Modelle und Techniken vereinfachen.
- **Plattformen, welche die Techniken und Werkzeuge integrieren.** Diese Plattformen zeigen die Anwendungsmöglichkeiten und wie diese Werkzeuge den Entwicklungsprozess verbessern können.

DIAMONDS legt großen Wert auf Verbreitungs- und Verwertungsaktivitäten. Tatsächlich besteht unsere Absicht darin, die besten Praktiken und wissenschaftlichen Erkenntnisse, die innerhalb des DIAMONDS-Projektes entstehen, außerhalb der Grenzen dieses Projektes zu verbreiten und so andere interessierte europäische Parteien zu erreichen.

2. VORAUSSETZUNGEN, UNTER DENEN DAS VORHABEN DURCHGEFÜHRT WURDE

Wachsende Sicherheitsbedenken von Endbenutzern, Anbietern und Regulierungsbehörden zusammen mit der steigenden Bedrohung der Sicherheit, insbesondere für vernetzte Systeme und Anwendungen, stärken die Nachfrage nach IT-Sicherheitstests, unterstützenden Methoden und Werkzeugen. Nach [14] wird der Sicherheitstest-, Mess- und Analysegerätemarkt Zeuge hoher Wachstumsraten von 2006 bis 2016: "Der Gesamtumsatz in 2006 für den Sicherheitsprüfgerätemarkt betrug 67.200.000 US$. Im Jahr 2008 wird der Markt voraussichtlich eine hohe Wachstumsrate von 37,8 Prozent aufweisen. Es wird von einer jährlichen Wachstumsrate von 32,0 Prozent für die Jahre 2006 bis 2013 ausgegangen." Diese Studie analysierte insbesondere Testgeräte für Firewall, VPN und Intrusion Detection and Prevention. Sie identifiziert die drei wichtigsten Anforderungen ans Sicherheitstesten: (1) Zero-Day-Schwachstellen- und veröffentlichte Schwachstellen-Tests, (2) wiederholbare Nutzung der bestehenden Test- und Analyse-Skripts, und (3) einfach zu bedienende, grafische Benutzeroberflächen für Sicherheitstest- und Messwerkzeuge.

Auf der anderen Seite, wie zum Beispiel in [38] hingewiesen wird, steht fest, dass "die Sicherheit eines Systems mit großem Softwareanteil direkt mit der Qualität ihrer Software in Zusammenhang stehen". Über 90% der Software-Sicherheitsvorfälle wurden durch Ausnutzung bekannter Software-Defekte durch Angreifer verursacht. Eine Analyse von 45 e-Business-Anwendungen zeigte, dass 70% der Sicherheitsmängel Konstruktionsfehler waren. Darüber hinaus hat das SEI festgestellt, dass selbst erfahrene und fähige Software-Ingenieure im Durchschnitt einen Defekt alle neun Zeilen Code einbringen. Dies führt dazu, dass ein System mit einer Million Zeilen Code typischerweise 1.000-5.000 Mängel beim Kunden enthält.

Laut Gartner [17] "Die Unternehmen erkennen allmählich die Bedeutung der Sicherheitslückenerkennung, dass eine Nachfrage nach Sicherheits-Testing-Tools entstehen lässt. Weil Prozess- und kulturelle Veränderungen erforderlich sind, um diese Tools in den Software-Lebenszyklus zu integrieren, wird es mehr als fünf Jahre dauern bevor statische Application Security Testing-Technologien den Gipfel der Produktivität erreichen.“ Dies gilt nicht nur für alle statischen Analyse-Tools, sondern auch für alle systematischen Sicherheits-Testmethoden. Andererseits hat eine große Zahl von Unternehmen, die kritische Systeme herstellen und nutzen, längst die Bedeutung der Beseitigung potenzieller Sicherheitsprobleme in den frühen Entwicklungszyklen eines Systems erkannt. Dies gilt insbesondere für die teilnehmenden Unternehmen, die Fallstudien in DIAMONDS anbieten (Banken, Funkverkehr, Smart Cards, Verkehr), aber auch im Raumfahrt- und Verteidigungsgeschäft, und gewinnt schnell an Interesse in anderen Bereichen.

DIAMONDS adressierte diesen zunehmenden Bedarf an systematischen IT-Sicherheits-Testmethoden durch die Entwicklung von Techniken und Werkzeugen, die effizient verwendet werden können, um vernetzte Anwendungen in unterschiedlichen Domänen zu sichern.

3. PLANUNG UND ABLAUF DES VORHABENS

3.1 PARTNER

Partner	Typ	Aufgabe	BEITRÄGE
Giesecke &Devrient	LC	Industrieller Partner	Definition von Fallstudien, Entwicklung von Methoden für Sicherheitsprüfungen sowie Anwendung selbiger auf die Fallstudien.
Dornier Consulting	LC	Technologielieferant	Definition von Fallstudien und Entwicklung der Prüfrahmen als Beitrag zur Experimentalplattform.
Testing Technologies IST GmbH	KMU	Technologielieferant	Entwicklung von Methoden und Werkzeugen innerhalb der Experimentalplattform
FhG FOKUS	R&D	Forschungspartner, Projekt- und deutscher Koordinator WP0-Leiter	Entwicklung von Methoden und Werkzeugen zur risikogesteuerten Generierung von Prüfungen und Ausfallanalysen.

Tabelle 1 Projektpartner

3.2 ARBEITSBESCHREIBUNG

DIAMONDS war in sechs Arbeitspakete unterteilt: In Arbeitspaket WP1 (Fallstudien) wurden die Anforderungen an die Techniken und Methoden festgelegt, die jeweils in Arbeitspaket WP2 und WP4 entwickelt werden sollten. Die Experimentalplattformen und Werkzeuge, welche die Sicherheitsprüftechniken und -methoden unterstützend begleiten, wurden in Arbeitspaket WP3 entwickelt. In Arbeitspaket WP1 wurden in zwei Runden die Fallstudien ausgearbeitet, die sicherheitsrelevanten Risiken analysiert, Sicherheitsprüfungen unter Anwendung der DIAMONDS-Sicherheitsprüftechniken sowie Methoden und Techniken entwickelt, Ergebnisse ausgewertet und Rückmeldung an die Arbeitspakte WP2, WP3 und WP4 gegeben. DIAMONDS wurde durch Arbeitspaket WP5, welches sich mit der Verbreitung und Verwertung befasst und Arbeitspaket WP0, welches sich mit dem Management des gesamten Projekts befasst, abgerundet.

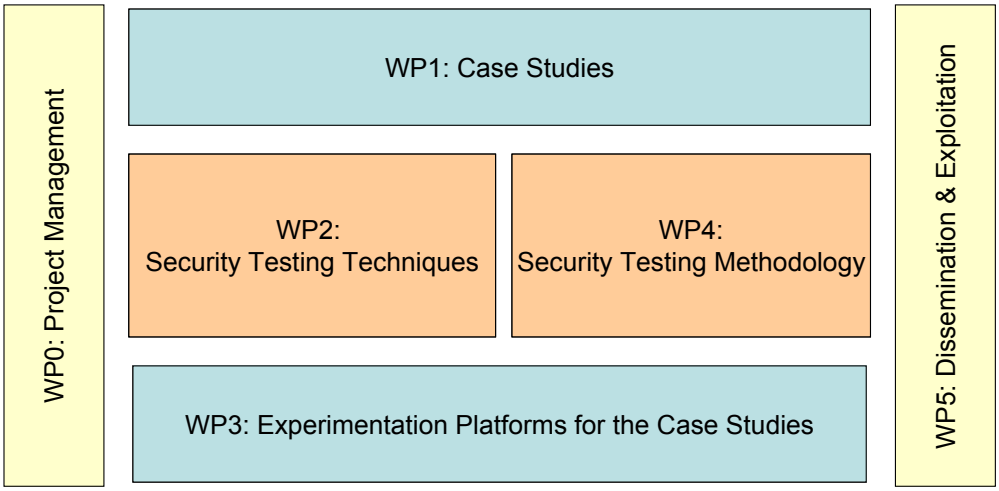

Abbildung 1: Aufteilung der Arbeitspakete in DIAMONDS

Arbeitspaket 0 – Projektmanagement

Das allgemeine Ziel dieses Arbeitspaketes war die Sicherstellung, dass das Projekt effizient gemäß der Arbeitsbeschreibung und den Verträgen mit dem Zuwendungsgeber sowie den ITEA-Richtlinien abläuft sowie Meilensteine und Arbeitsergebnisse wie geplant entwickelt werden.

Arbeitspaket 1 – Industrielle Fallstudien und Demonstratoren

Dieses Paket hat eine Reihe von industriellen Fallstudien in verschiedenen Anwendungsbereichen definiert und umgesetzt. Die Fallstudien wurden auf Basis der in WP3 entwickelten Werkezuge durchgeführt, um die modellbasierten Testtechnologien für Sicherheit, die in WP2 entwickelt wurden, zu bewerten. Die Ziele dieser empirischen Untersuchung waren es, Leitlinien für die Verwendung von Testtechniken zu erstellen. Mit Hilfe solcher Leitlinien konnte festgelegt werden, welche Techniken für welche Sicherheitstestziele, welche unterschiedlichen Anwendungsbereiche, verschiedenen Testebenen oder Phasen der Systementwicklung anwendbar sind und wie diese Techniken zu der allgemeinen Zuverlässigkeit und Verlässlichkeit des Systems beitragen können. Die Fallstudien wurden genutzt um die Anforderungen an Techniken und Werkzeuge zu definieren, die im Verlauf dieses Projekts entwickelt wurden. Die Arbeitspakete WP2, WP3 und WP4 konnten dann diese Anforderungen als Input für ihre Entwicklungsarbeit nutzen. Nach der Bereitstellung der Techniken, Werkzeuge und Methoden an WP1 konnte im WP1 bewertet werden, ob diese Technologien einen messbaren Vorteil bei dem Test der Sicherheit in den verschiedenen Anwendungsbereichen aufzeigen. WP1 konnte als Kunde und Lieferant für die Arbeitspakete WP2, WP3 und WP4 ein kontinuierliches Feedback geben, und dadurch zu einer kontinuierlichen Verbesserung der DIAMONDS-Technologien beitragen.

Arbeitspaket 2 – Sicherheitsprüftechniken

Das Arbeitspaket 2 hat an der Entwicklung und Bereitstellung von Methoden und Algorithmen für eine modellbasierte Sicherheitsprüfung gearbeitet, um Sicherheitseigenschaften vernetzter Anwendungen in unterschiedlichen Domänen (Netzwerkanwendungen, Dienste, Automobilsoftware, Chipkarten etc.) prüfen zu können. Da diese Domänen sehr wichtig sind, musste sichergestellt werden, dass ein bestimmtes Sicherheitsniveau dauerhaft erreicht werden kann. Darüber hinaus war von entscheidender Bedeutung, die Prüfung, ob das System hinreichend robust und belastbar ist, dahingehend zu erweitern, dass festgestellt werden kann ob die Sicherheitseigenschaften eines Systems selbst im Falle von Fehlern oder Angriffen aufrechterhalten werden. Das Arbeitspaket 2 widmete darüber hinaus dem Problem der Formalisierung

Sicherheitsrichtlinien. Das Sicherheitsniveau eines Systems wird üblicherweise durch eine Sicherheitsrichtlinie bestehend aus einer Reihe von Regeln, die das Verhalten des Systems steuern, beschrieben. Eine solche Sicherheitsrichtlinie durch Spezifikationen in natürlicher Sprache definiert, die nicht ohne weitere Formalisierung automatisiert prüfbar sind. DIAMONDS löste das Problem, indem es formelle Prüf- und Überwachungstechniken zur Prüfung der Sicherheitseigenschaften definiert und entwickelt hat.

Arbeitspaket 3 – Experimentelle Plattform und Werkzeugverbindungen

Das Arbeitspaket 3 widmete sich der Aufgabe, bestehende Modellierungs- und Testwerkzeuge entsprechend der in WP2 entwickelten IT-Sicherheitstesttechniken, zu erweitern und geeignete Versuchsplattformen zur Unterstützung der Fallstudien aus WP1 aufzubauen. Die deutschen Partner konzentrierten sich hierbei hauptsächlich auf die Anforderungen der deutschen Fallstudien. Im Projektverlauf.

Arbeitspaket 4 – Sicherheitsprüfmethodik

Dieses Arbeitspaket verfolgte und löste drei Aufgaben. Die erste Aufgabe war die Entwicklung und Bereitstellung eines Katalogs für Sicherheitsprüfmuster. In der Softwareentwicklung ist ein Entwurfsmuster eine allgemein wiederverwendbare Lösung für ein übliches, während des Entwurfs auftretendes Problem. Muster wurden bereits für viele verschiedene Software-Aspekte geschrieben, einschließlich der Benutzeroberflächengestaltung, Sicherheit, Telekommunikation und zum Teil für Systemprüfungen. Doch trotz der weitverbreiteten Nutzung solcher Prüfungen und der Wichtigkeit der Systemsicherheit, existierte vor dem Beginn des Projekt DIAMONDS kein Katalog für Sicherheitsprüfmuster.

Die zweite Aufgabe des Arbeitspakets 4 war die Entwicklung einer modellbasierten Sicherheitsprüfmethode. Vor Projektbeginn existierten verschiedene Ansätze für modellbasierte Prüfverfahren, doch nur wenige beschäftigen sich im Besonderen mit der IT-Sicherheit. IT-Sicherheit ist eine nicht-funktionale Systemeigenschaft, die oftmals von beinahe jeder einzelnen Komponente eines Software-Systems beeinflusst wird. Die in DIAMONDS entwickelten Ansätze berücksichtigen dieses.

Die dritte Aufgabe des Arbeitspaketes 4 war die Entwicklung risikobasierter Prüfmethoden. Obwohl die Kombination aus die Risikoanalyse und Prüfverfahren durchaus sehr nützlich sein kann, da sie die Möglichkeit bietet, Prüfungen zu priorisieren und unternehmensbezogene Risiken auf hoher Ebene in den Prüfprozess mit einzubeziehen, waren uns zu Projektbeginn keine Ansätze bekannt, die Risikoanalyse und Prüfverfahren miteinander kombinieren. DIAMONDS konnte mehre Ansätze für den risikobasierten Test entwickeln und diese in einer Methode für den risikobasierten Test zusammenfassen.

Arbeitspaket 5 – Verbreitung und Nutzung

Dieses Arbeitspaket hatte zwei Hauptaufgaben. Die erste Aufgabe war, die Ergebnisse des Projektes sowohl in der Industrie als auch im akademischen Bereich zu verbreiten. Die zweite Aufgabe war, zur Standardisierung von Techniken und Methoden des IT-Sicherheitstest. Die deutschen Partner konzentrierten sich auf die Ergebnisse aus den deutschen Fallstudien.

3.3 DIE WICHTIGSTEN MEILENSTEINE/ARBEITSERGEBNISSE

Nr.	Datum	Jahr	Meilenstein [M] oder Arbeitsergebnisbezeichnung [A]
1	Q2	2011	[A1] D1.WP1: Sammlung und Priorisierung der Anforderungen aus einer initialen Fallstudienspezifikation (WP1); D1.WP2: Stand der Technik und Wissenschaft der Sicherheitstesttechniken (WP2); D1.WP3: Stand der Technik und Wissenschaft der Sicherheitstestwerkzeuge (WP3);
	Q2	**2011**	**[M1] Anforderungen und Stand der Technik und Wissenschaft (Identifizierung der Sicherheitsprüftechniken)**
2	Q4	2011	[A2] D2.WP2: Konzepte für modellbasiertes Sicherheitstesten D2.WP3: Entwurf von Sicherheitsprüfprogrammen (WP3) initiale Definition des Common Framework (WP1); [D2.WP4.T2: Stand der Technik und Wissenschaft für modellbasierte Sicherheitstestmethoden (WP4.2) und [D2.WP4.T3: Stand der Technik und Wissenschaft für risikobasierte Sicherheitstestmethoden (WP4.3)
	Q4	**2011**	**[M2] Identifizierung der Sicherheitstesttechniken für die Fallstudien**
3	Q2	2012	[A3] D3.WP1: Verfeinerte Beschreibung des Demonstrators, erste Fallstudienergebnisse und überarbeitete Fallstudienspezifikationen (WP1) D3.WP2: Erste modellbasierte Sicherheitstestmethoden (WP2) D3.WP3: Erste Sicherheitstestwerkzeuge (WP3) D3.WP4a: Erster Katalog mit Sicherheitsprüfmustern (WP4.1) D3.WP4b: Erste Modellbasierten Sicherheitsprüfmethoden (WP4.2) und D3.WP4b: Erste risikobasierte Sicherheitstestmethoden (WP4.3)
	Q2	**2012**	**[M3] Erste Sicherheitstestmethoden fertig**

Nr.	Datum	Jahr	Meilenstein [M] oder Arbeitsergebnisbezeichnung [A]
	Q3	2012	[A4] D4.WP1: Finale Beschreibung des Demonstrators, Überarbeitete Fallstudienergebnisse und überarbeitete Fallstudienspezifikationen (WP1) D4.WP5: Spezifikationen der Schulungsumgebung für Sicherheitstesten
	Q3	**2012**	**[M4] Überarbeitete Fallstudien**
5	Q1	2013	[A5] D5.WP1: Finale Version Fallstudienergebnisse (WP1) D5.WP2: Finale Version modelbasierte Sicherheitstesttechniken (WP2) D5.WP3: Finale Version Sicherheitsprüfprogrammen (WP3) D5.WP4: Finale Version Sicherheitstestmethodik (WP4.1, WP4.2, WP4.3)
	Q1	**2013**	**[M5] Sicherheitstest Framework und Anleitungen, Erfahrungsbericht zu den Fallstudien**

Tabelle 2 Arbeitsergebnisse und Meilensteine

Fraunhofer FOKUS leitete sowohl das deutsche wie auch das europäische DIAMONDS Projekt. Im Rahmen der Leitungstätigkeiten wurden Arbeitstreffen auf nationaler und internationaler Ebene sowie der Auftritt der Projekte auf den ITEA Symposien sowie auf Messen und Konferenzen durchgeführt und geplant. Insbesondere wurden der Projektfortschritt überwacht und die Arbeiten zu den Meilensteinen und den DIAMONDS Reviews geplant und koordiniert. Alle deutschen Projektpartner haben ihre administrativen und technischen Arbeiten wie im Plan vorgesehen geleistet. Das DIAMONDS Projekt hat alle Meilensteine fristgerecht abgeliefert. Auf den nationalen Treffen wurde die inhaltliche Arbeit an den deutschen Fallstudien der Firmen Dornier Consulting und Giesecke & Devrient abgestimmt und die weitere Arbeit im deutschen Projekt geplant. Die internationalen Treffen mit Partnern aus Deutschland, Finnland, Norwegen, Frankreich, Österreich und Luxemburg dienten der Abstimmung der Arbeiten auf internationaler Ebene und zur Vorbereitung der gemeinsamen Projektauftritte in Workshops, auf Messen und Konferenzen.

Abbildung 2: DIAMONDS gewinnt den Exhibition Award 2011

Im Rahmen der jährlichen ITEA & ARTEMIS Co-summit Projektpräsentation war DIAMONDS insgesamt vier Mal vertreten. In den Jahren 2011 und 2012 wurde DIAMONDS jeweils mit dem Exhibition Award ausgezeichnet. Dieser Preis wurde durch eine Abstimmung unter den Teilnehmern ermittelt und geht an das Projektteam, das seine Projektinhalte am besten darstellen und den Besuchern vermitteln kann. Die Dauer des Projektes war geplant vom 1.10.2010 bis 31.03.2013. Sie wurde im Laufe des Projektes aufwandsneutral bis zum 30.06.2013 verlängert, um sie an die Projektdauer der anderen europäischen Partner anzugleichen.

4. WISSENSCHAFTLICHER UND TECHNISCHER AUSGANGSSTAND

Das Hauptziel der Sicherheit ist die Verminderung oder Beseitigung von unerwünschten Ereignissen. Das Feld der Informationssicherheit ist gewachsen: es reicht viel weiter als nur die Kryptographie und ist eine übergreifende Tätigkeit im Softwarelebenszyklus geworden. Praktische Sicherheit bezieht genügendes Verständnis des Risikos mit ein, um in der Lage zu sein, dieses Risiko zu handhaben und es sorgfältig zu beseitigen. Der grundsätzliche Zweck von Sicherheitsstandards ist es, Gegenmaßnahmen zu spezifizieren, die ein System gegen bestimmte Formen des Missbrauchs schützen können. Gegenmaßnahmen hängen hauptsächlich vom Verständnis des Aufspürens und des Erkennens der Schwachstellen innerhalb eines Systems ab.

Testen kann als Tätigkeit gesehen werden, solche Schwachstellen proaktiv aufzudecken. Boehm und Basili haben eine Liste der Vorteile der proaktiven Fehler-Beseitigung kritisch durchgearbeitet und aktualisiert. Zusammenfassend kann gesagt werden: Ein Software-Problem nach der Auslieferung zu finden und zu beheben ist häufig 100mal teurer als dies während der Anforderungs- und Design- Phase zu tun. Fehler werden normalerweise lokalisiert. Testen ist ein wirksamer Weg, Fehler zu finden.

Testtechniken

Testen ist nach wie vor die Hauptmethode, Funktionalität, Robustheit, Performance, Skalierbarkeit, Zuverlässigkeit und die Flexibilität von Systemen zu überprüfen, weil es die einzige Methode ist, Eigenschaften eines Systems in seiner Zielumgebung objektiv abzuleiten.

Testtechniken zielen darauf ab, die Konformität eines Systems im Test bezüglich seiner Spezifikation zu bestätigen und sicherzustellen, dass das System richtig funktionieren wird. Das wird auch Konformitätstest genannt. Es werden zwei Haupt-Testtechniken unterschieden: aktives und passives Testen.

Aktives Testen

Diese Technik wird durch externe Tester angewendet, indem eine Reihe von Eingaben erzeugt und geprüft wird, ob die Reihe der Ergebnisse der erwarteten entspricht. Eine Systemspezifikation besteht normalerweise aus Steuerungs- und Datenanteilen. Mehrere Methoden sind für die Testgenerierung vorgeschlagen worden, die auf unterschiedlichen Systemmodellen und Fehler-Abdeckungs-Kriterien basieren. Die meisten Arbeiten beschäftigen sich mit dem Test der Programm-Steuerungs-Anteile, die durch endliche Zustandsautomaten, Labelled Transition Systems (LTS), etc. modelliert werden. Diese Diagrammdarstellungen sind für die Beschreibung und Überlegungen zur Testgenerierung sehr nützlich. Für das Testen des Softwareanteils sind verschiedene Modelle mit unterschiedlichen Spezifikationssprachen vorgeschlagen worden. Traditionelle Testmethoden prüfen üblicherweise ein System als Ganzes oder testen die enthaltenen Bestandteile isoliert. Diese Systeme als Ganzes zu prüfen wird schwierig wegen der Vielzahl von Kombinationen von Systemzuständen und Variablenwerte, auch bekannt als Explosion des Zustandsraums. Es ist eine Herausforderung, die Zahl von erforderlichen Tests minimieren und gleichzeitig eine gute Fehlerabdeckung garantieren zu können. Diese Standardmethode ist hauptsächlich an praktischen Anforderungen orientiert.

Es ist zu beachten, dass die meisten modernen Testtechnologien sich auf modellbasiertes Testen [37][30] beziehen und unser Projekt sich in diesen Rahmen einfügt.

Passives Testen

Passives Testen ist eine Vorgehensweise, Fehler in einem System im Test zu entdecken, indem dessen Ein-/Ausgabe-Verhalten beobachtet wird, ohne seine normalen Operationen zu beeinflussen. Die übliche Herangehensweise bei passiven Tests besteht darin, das Verhalten unter Testbedingungen aufzuzeichnen und zu versuchen, durch Vergleich dieses Verhalten mit der Spezifikation Fehler zu finden. Andere Ansätze erforschen relevante Eigenschaften, die für eine richtige Ausführung erforderlich sind und überprüfen diese dann am zu testenden System. Der größte Teil der Arbeit des passiven Testens beruht auf endlichen

Zustandsautomaten und ist auf den Steuerungsanteil des zu testenden Systems ausgerichtet, ohne den Datenanteil zu berücksichtigen. Um Protokolldatenanteile zu bewältigen, werden endliche Zustandsautomaten eingesetzt, um das System zu modellieren, die Parameter und Variablen einschließen, um Daten zu kodieren. Passive Testtechniken sind besonders interessant, wenn dem Tester keine direkte Schnittstelle zur Verfügung steht (sog. Beobachtungs- und Kontrollpunkt), um mit der Implementierung im Test („Implementation Under Test", IUT) zu interagieren oder wenn die Implementierung aus Bestandteilen gebildet wird, die in ihrer eigenen Umgebung laufen und für längere Zeit nicht gestoppt oder unterbrochen werden können. Tatsächlich braucht ein passiver Tester nicht mit der IUT zu interagieren, er sammelt nur die Ausführungsprotokolle, um diese dann zu analysieren, ohne das Verhalten der IUT zu stören.

In den letzten Jahren sind Beobachtungsmethoden, die auf passivem Testen beruhen, auf verschiedenen Gebieten (bewegliche vernetzte Systeme, Web-basierte Systeme) [20][46] erfolgreich angewandt worden. Es war eines der Ziele von DIAMONDS, diese Techniken für Sicherheitstest anzuwenden.

Sicherheitstest- und -prüftechniken

Die Prüfung eines Systems auf seine Sicherheit ist überraschenderweise (wenn man die Bedeutung des Problems und den Umfang der Forschung an anderen Methoden, die das Thema Sicherheit von auf Software-basierten Systemen behandeln, in Betracht zieht) ein relativ neues Aufgabengebiet, welches erst in den letzten Jahren beachtet wurde, und der erste Forschungs-Workshop zu diesem Thema wurde 2008 durchgeführt (SecTest08, Lillehammer). Natürlich sind seit längerem mehrere Werkzeuge verfügbar, die auf Angriffe auf Systeme (z.B. Vulnerability Scanners) abzielen, aber wir beziehen uns hier auf die systematischere Prüfung von Systemen unter Berücksichtigung spezifizierter Leitlinien oder Sicherheitseigenschaften.

Testen der Sicherheit wird angewendet, um festzustellen, ob ein software-basiertes System Daten schützt und die Funktionalität wie gewünscht aufrechterhält. Sicherheitstesten prüft auf Vertraulichkeit, Integrität, Authentizität, Berechtigung, Verfügbarkeit und Echtheitsgarantie. Sicherheitstesten ist in Service basierenden Umgebungen besonders herausfordernd, da service-basierte Netzwerk-Anwendungen verteilte Systeme in dynamischen Konfigurationen sind, die Netzwerk-Domänen überschreiten und in nicht bekannten Umgebungsbedingungen und Anwender-Szenarien ablaufen.

Sicherheitskontrolle wird typischerweise durch Sicherheitsstrategien definiert. Der größte Teil der auf Sicherheitsstrategien beruhenden Arbeit kann in zwei Hauptbereiche gegliedert werden: die Beschreibung der Strategie selbst und die Verifizierung der Regeln. Bis vor kurzem gab es für fast alle Modelle keine echte Strategiespezifikation, sondern nur eine Beschreibung auf niedriger Ebene wie z.B. Zugriffsberechtigungslisten.

So führt die Analyse der Zugriffskontrolle dazu, mehrere Zugriffskontrollmodelle zu definieren, die eine formale Darstellung von Sicherheitspolicen bereitstellen könnten und in einigen Fällen den Nachweis von Eigenschaften mit Hilfe der Zugriffskontrolle erlauben. Für die große Mehrheit der Modelle werden die Sicherheitsregeln mit drei Hauptmodalitäten (Erlaubnis, Verbot und Verpflichtung) definiert, die die möglichen Einschränkungen auf das Verhalten des Systems [31] ausdrücken. Unter diesen Modellen können wir zum Beispiel die Policy Description Language (PDL) [26], Ponder [32] und OrBAC (Organisational Based Access Control) [1] erwähnen. Bezüglich der Überprüfung dieser Regeln beschäftigen sich die meisten Arbeiten mit dem Test von Firewall-Regeln. Die ersten Vorschläge bestanden im manuellen Durchführen der Tests der Regeln. Das setzt voraus, dass der Testaufbau von Experten durchgeführt wird, die sich darauf konzentrieren, Spuren bekannter Angriffe zu entdecken. Später neigten Forschungen dazu, sich auf die Überprüfung der Sicherheitsregeln zu konzentrieren, um Fehler oder fehlerhafte Konfigurationen wie Redundanzen oder Widersprüche zu entdecken [22][21]. Einige Ansätze schlagen vor, sich auf die Validierung durch Überprüfung der Übereinstimmung eines Systems mit den Sicherheitsrichtlinien zu konzentrieren. In [8] zeigen die Autoren, wie die Netzsicherheit einer Organisation auf hoher Ebene formal spezifiziert und wie diese Spezifikation dazu verwendet werden kann, Testfälle automatisch zu erzeugen, um

ein System zu prüfen. Im Gegensatz zu anderen Firewall-Testmethoden, wie Penetrationstests, prüft dieser Ansatz die Übereinstimmung mit einer spezifizierten Strategie.

Diese Testfälle sind organisationsspezifisch – d. h. sie hängen von den Sicherheitsanforderungen und von der Netzwerkarchitektur einer Organisation ab – und können Fehler sowohl in Firewall-Produkten als auch in deren Konfiguration aufdecken. Jedoch ist dieses Modell auf das Netzmanagement und speziell auf die Netzwerk- und Transport-Schicht des TCP/IP-Stacks beschränkt. Außerdem handelt es sich dabei um einen theoretischen Ansatz und es gibt noch keine Werkzeuge, um den Testprozess zu automatisieren und ihre Wirksamkeit in einer realen Fallstudie zu bewerten. In [44] wählen die Autoren einen anderen Ansatz, um Tests von Netzwerk-Sicherheitsregeln zu erreichen. Sie drücken das Netzwerkverhalten mit Hilfe von markierten Transitionssystemen aus. Dann schlagen sie für jedes Element ihrer Sprache und jeden Typ der Regel ein Testmuster genannt „Tile“ vor. Anschließend kombinieren sie diese „Tiles“ in kompletten Testfällen für die gesamte Regel, um deren Gültigkeit zu überprüfen.

Der in diesem Projekt vorgeschlagene Ansatz unterscheidet sich von diesen Vorschlägen in den Annahmen über die Strategie und in der Methode, Testfolgen zu erzeugen. Zunächst machen wir keine Annahme über die Beschreibungssprache der Strategie. Stattdessen schlagen wir einen Rahmen vor, um Regeln in einer formalisierten Form zu spezifizieren, so dass wir sie auf unser mathematisches Modell anwenden können. Dann erzeugen wir den ganzen Satz von Testfällen und zwar in automatisierter Form.

In Bezug auf Test und Fehler-Einbringung verbindet die in Avresky et al. [9] vorgestellte Arbeit Fehlereinbringung und formelle Prüfung. Sie verwenden Zusicherungen, um das Systemverhalten, besonders Fehlertoleranz-Mechanismen, zu spezifizieren. Von diesen Annahmen wird ein Ausführungsbaum abgeleitet, der alle möglichen Ausführungen des Systems darstellt, die sich aus der Menge der Zusicherungen ergeben. Um den Fall mehrerer vorhandener Kopien darzustellen, werden mehrere Ausführungsbäume betrachtet, für die verschiedene Pfade aktiviert werden können. Die Menge der Bedingungen, um diese Pfade zu aktivieren, führt den Testentwickler durch die Erstellung von Aktivitäten, die vom System durchgeführt werden müssen, um es in einen Zustand zu bringen, in dem Fehler eingebracht werden können. Diese Fehler werden auch vom Ausführungsbaum abgeleitet. Da das Modell auch die erwarteten Ergebnisse für die aktivierten Pfade enthält, ist die Ergebnisanalyse ebenso möglich. Der Ansatz wurde verwendet, um ein Inter-Replica-Protokoll eines fehlertoleranten Protokoll-Stacks auszuwerten.

Der Ansatz von Avresky et al., der unserem beabsichtigten Ansatz nahekommt, beruht nicht auf endlichen Zustandsautomaten. Soweit wir wissen, kombiniert keine andere Arbeit Konformitätstest und Fehlereinbringung für die Beurteilung der Funktionssicherheit. Ausgehend von der Verhaltensspezifizierung eines Dienstes und vor allem in Gegenwart von Fehlern, ist es die Idee, Fehler zu erzeugen und zu beobachten, ob die gewünschten Mechanismen der Fehlertoleranz zum Tragen kommen. Zunächst wird ein Kommunikations-Fehlermodell verwendet, da es für Dienste-Architekturen gut geeignet ist.

Penetrationstests

Penetrationstests bewerten Systemsicherheit in der Gestalt eines Angreifers. Sie analysieren Systeme auf jede mögliche Schwachstelle, die sich aus schlechter oder unpassender Systemkonfiguration, bekannten und/oder unbekannten Hardware- oder Softwarefehlern oder betriebliche Schwächen im Prozess oder den technischen Gegenmaßnahmen ergeben. Ziele von Penetrationstests sind, Durchführbarkeit eines Angriffs und die Auswirkung einer erfolgreichen Attacke zu bestimmen.

IT-Sicherheitstest-Standards

Verschiedene internationale und de facto Standards sind für DIAMONDS relevant und werden für die geplante Forschung und Entwicklung berücksichtigt:

Das Open Source Security Testing Methodology Manual (OSSTMM [42]) definiert eine Menge von Methoden für das Durchführen von Sicherheitstests und ihrer entsprechenden Metriken. OSSTMM

konzentriert sich auf die technischen Details genau der Elemente, die geprüft werden müssen, was vor, während und nach einem Sicherheitstest zu tun ist, und wie die Ergebnisse zu messen sind.

Das National Institute of Standards and Technology (NIST) SP 800-115 [33] definiert eine Methodik, die weniger ausführlich als OSSTMM ist, aber dasselbe Gebiet abdeckt, häufig durch direkten Verweis.

Das Information Systems Security Assessment Framework (ISSAF [19]) Penetration Testing Framework bewertet Gefahren durch Bewertung der Durchführbarkeit von Angriffen.

Die Common Criteria [4] definieren eine Menge von Anforderungen für die Zertifizierung von Informationstechnologieprodukten, die von den Bewertenden und den Entwicklern verwendet werden können, um zu demonstrieren, dass das Produkt eine Reihe von Sicherheitsstandards erfüllt. Dieser Standard definiert sieben Zusicherungslevel der Evaluierung (Evaluation Assurance Level, EAL), von EAL1 bis EAL7, und schließt eine Reihe von Anforderungen an die Testaktivitäten für das zu zertifizierende Produkt ein.

Modellbasiertes Sicherheitstesten

Obwohl mehrere Forschungsarbeiten über modellbasierte Sicherheit (siehe z.B. [25][7]) und modellbasiertes Testen (siehe z.B. [35]) existieren, gibt es wenig Vorarbeit über modellbasierte Sicherheitstests. Die einzigen Arbeiten, die wir kennen, sind [24][23][29][41][28][16].

Kaksonen et al. vom PROTOS-Projekt (1999-2001) diskutieren und implementieren einen modellbasierten Ansatz für das Sicherheitstesten unter Verwendung von Syntaxtests als Startpunkt und implementieren die Modelle durch Benutzung der angereicherten Backus-Naur Form (ABNF). Die Herangehensweise von PROTOS an das modellbasierte Testen liest kontextfreie Grammatik (BNF, ASN.1, und XML) für kritische Protokoll-Schnittstellen ein und erzeugt die Tests durch systematisches Durchgehen der Protokoll-Spezifikationen.

Jürjens und Wimmel [23][16] gehen das Problem an, Testsequenzen aus abstrakten Systemspezifikationen zu erzeugen, um mögliche Schwachstellen von sicherheitskritischen Systemen zu entdecken. Beide Papiere gehen davon aus, dass die Systemspezifikation, aus der die Tests erzeugt werden, formal in der Sprache Focus definiert ist. Der Beitrag [23] konzentriert sich auf das Testen von Firewalls, wohingegen sich [16] auf Transaktionssysteme konzentriert. In [24] erweitert Jürjens [16] durch die Betrachtung von Systemspezifikationen, die in der Sprache UMLsec geschrieben sind (im Gegensatz zu Focus).

In [29] fassen Blackburn et al. die Ergebnisse einer Anwendung eines modellbasierten Ansatzes zusammen, funktionale IT-Sicherheitstesten zu automatisieren. Der Ansatz beinhaltet die Entwicklung von Modellen von Sicherheitsanforderungen als Basis für die automatischen Testvektoren- und Testtreiber-Generierung. Insbesondere werden Sicherheitsanforderungen in dem sogenannten SCR-Tool geschrieben und übersetzt in Testspezifikationen, die ihrerseits in Testvektoren und Testtreiber transformiert werden. Der Ansatz zielt auf Java-Applikationen und Datenbankserver.

Mouelhi et al. [41] schlagen einen modellgetriebenen Ansatz für das Spezifizieren, Entwickeln und Testen von Zugangskontroll-Strategien in Java- Applikationen vor. Der Ansatz hat vier Hauptschritte. Der erste Schritt besteht darin, ein plattformunabhängiges Zugangskontroll-Modell für die Applikation zu entwerfen. Im zweiten Schritt wird das Modell in sogenannte Platform Specific Policy Decision Points (PDPs) umgewandelt. In Schritt drei wird der PDP vermittels Aspekt-orientierter Programmiertechniken in den funktionellen Code der Anwendung integriert. Schließlich, im vierten Schritt, wird die resultierende integrierte Anwendung gegen Tests geprüft, die vom plattform-unabhängigen Zugangskontroll-Modell erzeugt wurden. Ein anderer Ansatz, der die Spezifikation, die Entwicklung, das Testen und Überwachen von Sicherheitsstrategien abdeckt, ist im Politess-Projekt (Grenoble INP, ES, Smartesting) [44][46][3] vorgeschlagen worden.

In [28], präsentieren Wang et al. einen bedrohungsgesteuerten Ansatz für modellbasierte Sicherheitstests. Bei diesem Ansatz spezifizieren UML Sequenzdiagramme ein Bedrohungsmodell, d. h. Ereignis-Folgen, die während der Systemausführung nicht vorkommen sollten. Das Bedrohungsmodell wird dann als Basis für die

Codeinstrumentierung verwendet. Schließlich wird der instrumentierte Code wieder kompiliert und unter Benutzung zufällig erzeugter Testfälle ausgeführt. Wenn ein Ausführungsablauf mit einem von dem Bedrohungsmodell beschrieben Ablauf übereinstimmt, werden Sicherheitsverletzungen gemeldet und es sollten Aktionen vorgenommen werden, um die Bedrohung des Systems abzustellen.

Sicherheitstests in DIAMONDS-Anwendungsgebieten

Die Verifikation der Sicherheit im **Bereich der Banken** betrifft sowohl Anwendungen als auch die Infrastruktur. Gemäß diesen Zielen wird die Sicherheit erhöht durch das Design, das auf Normen und Anforderungen beruht, und durch die Zuverlässigkeit der Anwendung. Ansonsten sind die Testanwendungen für den Bankensektor dieselben wie für andere Bereiche. In Hinblick auf die Bewertung der Sicherheit und die Entdeckung von Sicherheitsmängeln gibt es einen Bedarf nach der folgenden Art von Methoden oder Werkzeugen: (1) Penetrationstests: Um effizient und genau zu sein, könnten diese Methoden an den Anforderungen für den Bankensektor ausgerichtet werden (z.B. PCI-DSS v1.2); (2) Development Tools: Sie werden verwendet, um Sicherheitslücken zu identifizieren, zu analysieren und zu beheben (z.B. HP WebInspect, Compuware DevPartner SecurityChecker 2.0); (3) Debugging Tools: Sie entdecken, analysieren und bewerten Abstürze der Applikation und bezeichnen einen Grad der Ausnutzbarkeit; und (4) Auditing and Review Tools: Diese Tools werden für die formelle Sicherheitsprüfung, die auf Normen oder Standards basiert, verwendet (z.B. Werkzeuge, die vom OWASP – Open Web Application Security Project zur Verfügung gestellt wird). Die Fallstudie setzt auf einigen der Hauptsicherheitsproblemen im Bankensektor auf: Schutz der Daten (Integrität, Verfügbarkeit, Vertraulichkeit, Prüffähigkeit), physischer Zugang und Daten-Zugriff (Verfügbarkeit, Vertraulichkeit, Prüffähigkeit), sichere Kommunikationskanäle (Integrität, Vertraulichkeit) und Sicherheitskontrollen (Integrität, Verfügbarkeit, Vertraulichkeit, Prüffähigkeit). Diese Sicherheitsthemen betreffen sowohl Kundenanwendungen (z.B. Onlinebanking, Onlinetransaktionen, usw.) als auch Bereiche des Kerngeschäftes (Konten-Management, Infrastruktur-Überwachung, Zugriffsmanagement).

Im **Transportation/Automotive Bereich** sollten drei Typen möglicher Schwachstellen in Fallstudien Berücksichtigung finden: (1) Verletzlichkeit der Sicherheit (Senden falscher Information an die Fahrzeuge bezüglich Verkehrsaufkommen, Geschwindigkeitsbegrenzungen usw.), (2) Verletzlichkeit des Komforts (Senden/Hacken von Diensten), und (3) Verletzlichkeit von Handel und kommerziellen Werten (Hacken von Maut-Systemen, Missbrauch von kommerziellen Daten). Für diese drei Klassen von Sicherheitsaspekten wurde eine passende Fallstudie für den Bereich Transportation/Automotive aus folgenden potenziellen Anwendungsgebieten ausgesucht: (1) Diagnose-Anwendungen (SW-Update oder -Flashen über Air-I/F, Ferndiagnose), (2) Infotainment-Anwendungen (Konnektivität mit Consumer-Elektronik im Fahrzeug) oder (3) kommerzielle Anwendungen (mobile Commerce, Handel, elektronische Maut, Telematik-Anwendungen, E-Call, Internetzugang im Fahrzeug).

5. FRAUNHOFER FOKUS, FÖRDERKENNZEICHEN 01 IS 100 31 A

5.1 ERREICHTE ERGEBNISSE

Fraunhofer FOKUS hat im DIAMONDS-Projekt Ergebnisse in den wesentlichen Arbeitspaketen erzielt. Im WP1 wurden die wissenschaftlich-technischen Ergebnisse der technischen Arbeitspakete in die Projekt-Fallstudien integriert und eine systematische Beschreibung der Fallstudienergebnisse für die interne und externe Verwertung erstellt.

Die Arbeiten in WP2 konzentrierten sich auf die Entwicklung von Techniken und Konzepten für den modellbasierten Sicherheitstest, entsprechender Evaluationstechniken und Risikoanalysekonzepten für den risikobasierten Test.

- Entwicklung von Konzepten und Spracherweiterungen für TTCN-3, sodass die Spezifikation und Ausführung von Fuzz-Tests möglich ist.
- Entwicklung eines Ansatzes zum Model-based Behavioural Fuzzing (MBBF) und deren Erweiterung um die Möglichkeit des Online Fuzzing (Online MBBF).
- Entwicklung eines Ansatzes zur kompositionellen Risikoanalyse auf Basis des CORAS Ansatzes

Die Arbeiten in WP3 zielten auf den Entwurf und die Implementierung von Werkzeugen, die die Techniken und Konzepte aus dem WP2 umsetzen. Schwerpunkt war die Entwicklung von RISKTest, einer Trace Management Plattform für den risikobasierten Test, sowie die Implementierung der Werkzeugunterstützung für MBBF und Daten-Fuzzing. Die Trace Management Plattform dient der Integration verschiedenartiger Modellierungs- und Testwerkzeuge, die alle zur Umsetzung der DIAMONDS-Methodik beitragen.

- Entwicklung von RISKTest, einer integrierten IT-Sicherheitstestplattform mit Integration der Werkzeuge CORAS, Papyrus-UML, ProR und der TTworkbench.
- Entwicklung der FOKUS Fuzzing-Bibliothek (Fuzzino).
- Integration von Fuzzino als Erweiterung der TTworkbench.

In WP4 wurden Sicherheitstestmuster entwickelt und in einem Sicherheitstestmusterkatalog zusammengefasst. Darüber hinaus wurde eine Methode für den risikobasierten Sicherheitstest entwickelt.

- Entwurf und Realisierung eines Katalogs von Sicherheitstestmustern mit Bezug auf bekannte Schwachstellen- und Angriffsmustern-Kataloge (WP4/M3-T3)
- Entwicklung einer Bibliothek von Sicherheitstestmustern auf Basis der ESG ISI IT-Sicherheitsheitsindikatoren
- Entwicklung einer modellbasierten Methodik für den risikobasierten Sicherheitstest (WP4/M3-T1&T2).

Die Arbeiten in WP5 befassten sich mit der Verbreitung und Verwertung der DIAMONDS-Arbeiten im Kontext akademischer und industrieller Interessenten sowie mit der Überführung der Ergebnisse in Standardisierungsgremien. Vor dem Hintergrund wurden die folgenden Tätigkeiten durchgeführt:

- Mitarbeit in ETSI-Arbeitsgruppen zum Thema IT-Sicherheitstest (ETSI MTS Interest Group (SIG) für "Security Testing" und ETSI „Industrial Specification Group (ISG) on Information Security Indicators (ISI)“).
- Veröffentlichung relevanter Projektergebnisse über Prä-Standardisierungsdokumente der ETSI.
- Aufbau und kontinuierliche Betreuung eines deutschen Industriebeirats, um die Kompatibilität der Projektergebnisse mit konkreten Industrieanforderungen zu wahren.

- Aufbau eines Vereins zur „Spezifikation Operationaler Sicherheit in Organisationen“ (SOSO), ähnlich dem Französischen „Club R2GS“1.

Im Folgenden werden die wichtigsten Ergebnisse noch einmal im Detail erläutert.

5.1.1 Model-based Behavioural Fuzzing

Um Sicherheitslücken in Software zu finden, hat sich Fuzzing als Testtechnik bewährt. Dazu werden die Schnittstellen des zu testenden Systems (SUT) mit ungültigen Eingabedaten stimuliert, um fehlende oder fehlerhafte Eingabevalidierungsmechanismen aufzudecken. Modellbasierte Fuzzer nutzen ein Modell des Schnittstellenprotokolls, um nicht per se ungültige Eingabedaten zu generieren, sondern sogenannte semi-valide Eingabedaten. Diese haben subtile Abweichungen zu gültigen Eingabedaten [34] und sind so in der Lage, effizient Fehler in der Eingabevalidierung aufzudecken. Diese Fehler können Verwundbarkeiten darstellen, wenn ungültige Eingabedaten nicht zurückgewiesen, sondern verarbeitet werden und so bspw. schadhafte Daten eingeschleust oder das System zum Absturz gebracht werden kann.

Protokolle spezifizieren zusätzlich zum Format gültiger Eingabedaten auch den gültigen Nachrichtenaustausch zwischen Komponenten, die über dieses Protokoll miteinander kommunizieren. Ungültige, d.h. im Sinne des Protokolls fehlerhafte Nachrichtensequenzen können ebenso Sicherheitslücken aufdecken, wenn bspw. Nachrichten, die gemäß Protokoll-Spezifikation nur einmal gesendet werden dürfen, wiederholt gesendet werden und so Schutzmechanismen gegen Pufferüberläufe aushebeln[38].

Das gezielte Auffinden von Sicherheitslücken mit Hilfe von ungültigen Nachrichtensequenzen nennen wir Behavioural Fuzzing (Verhaltensfuzzing). Während es viele Arbeiten zum Fuzzing von Eingabedaten gibt, ist das Behavioural Fuzzing ein Bereich, dem in der Forschung bislang nur wenig Beachtung geschenkt wurde. Ansätze zum Behavioural Fuzzing finden sich vereinzelt bei Becker et al. [2] in Form vom Entfernen, Einfügen oder Modifizieren einzelner Nachrichten mit Hilfe von Zustandsautomaten. Ähnliche Ansätze finden sich bei Hsu et al. [18], die die Typen von Nachrichten verändern und Nachrichten umordnen. Kitagawa et al. deckten mit Behavioural Fuzzing eine Sicherheitslücke im verbreiteten Apache Webserver auf[38].

Während die bisherigen Ansätze zum Behavioural Fuzzing sich auf Zustandsautomaten und einzelne Nachrichten bezogen, wurde im DIAMONDS-Projekt Behavioural Fuzzing für UML Sequenzdiagramme ausgearbeitet. Dazu wurde eine Reihe von Fuzzing-Operatoren entwickelt, die nicht nur einzelne Nachrichten innerhalb eines Sequenzdiagramms einfügen, löschen oder umordnen, sondern auch Kontrollstrukturen modifizieren, indem bspw. Schleifengrenzen erweitert bzw. Bedingungen für Verzweigungen negiert werden, um sicherheitsrelevante Fehler in der Protokollimplementierung aufzudecken.

Die Verwendung von Sequenzdiagrammen, die gültige Nachrichtensequenzen beschreiben, hat dabei den Vorteil, dass funktionale Testsuiten oder Protokollmitschnitte als Ausgangsbasis für IT-Sicherheitstests verwendet werden können, wenn kein Modell des Protokolls vorliegt. Indem ein oder mehrere Fuzzing-Operatoren auf ein gültiges Sequenzdiagramm angewendet werden, wird ein Sequenzdiagramm generiert, das eine ungültig Sequenz von Nachrichten beschreibt. Dieses Sequenzdiagramm stellt einen Behavioural Fuzz-Testfall dar. Abbildung 3 illustriert die Testfall-Generierung: Im linken Teil der Abbildung ist eine gültige Sequenz dargestellt, die aus der Eingabe einer Geheimzahl („enterPIN“) und dem Abheben von Geld („drawMoney“) besteht. Der Fuzzing-Operator „Remove Message“ wird auf die erste Nachricht („enterPIN“, mittig im Bild) angewendet. Dadurch entsteht die ungültige Sequenz (rechts im Bild), bei der ohne Eingabe einer PIN versucht wird, Geld abzuheben. Die Technik des Behavioural Fuzzing ist im DIAMONDS-Deliverable D2.WP2 beschrieben.

[1] Der Club R2GS ist ein gemeinnütziger Verein, der Akteure der IT und Kommunikationsindustrie in Frankreich um das gemeinsame Ziel verbindet, die Umsetzung etablierter Security Standards der operativen Sicherheitsinformations- und Eventmanagement bei großen Organisationen zu unterstützen

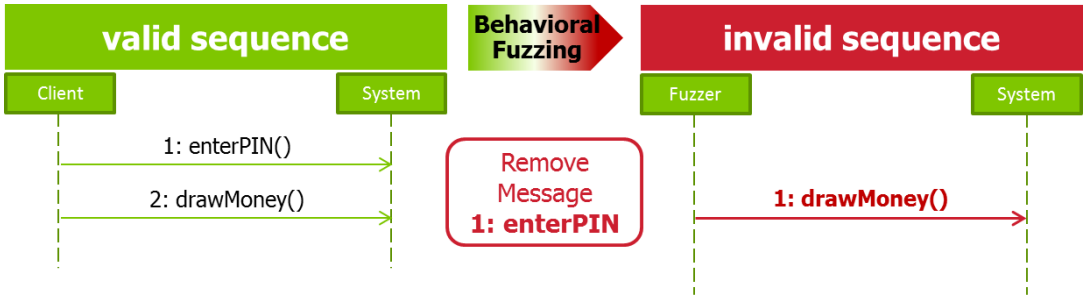

Abbildung 3: Anwendung eines Fuzzing-Operators auf ein gültiges Sequenzdiagramm

Da Fuzzing-Operatoren auf viele Elemente eines Sequenzdiagramms angewendet werden können, ist die Anzahl der Testfälle, die generiert werden können, so groß, dass in der Regel nicht alle Testfälle ausgeführt werden können. Wie man die Anzahl der Testfälle einschränken kann, um sich auf bestimmte Sicherheitsaspekte zu konzentrieren, wurde am Beispiel des Authentifizierungsmechanismus mit UMLsec gezeigt. UMLsec ist ein UML-Profil, mit dessen Hilfe sich unter anderem Informationen zur rollenbasierten Zugriffskontrolle im Modell hinterlegen lassen. Die entsprechenden UMLsec Stereotypen wurden so ergänzt, dass in einem Sequenzdiagramm die Informationen angegeben werden können, welche Nachrichten der Authentifizierung dienen und welche Nachrichten nur nach erfolgter Authentifizierung verarbeitet werden dürfen. Mit diesen Informationen lassen sich gezielt Testfälle generieren, die auf Sicherheitslücken im Authentifizierungsmechnismus testen, indem der Fuzzing-Operator „Remove Message“ wie in Abbildung 3 dargestellt, auf die Authentifizierungsnachricht angewendet wird, und der Fuzzing-Operator „MoveMessage“ die Abmeldenachricht vor die Nachrichten verschiebt, die eine vorherige Authentifizierung erfordern. Mit Hilfe des zweiten Fuzzing-Operators kann so getestet werden, ob der Abmeldemechanismus korrekt implementiert ist und Nachrichten, die eine Authentifizierung erfordern, nach der Abmeldung nicht mehr verarbeitet werden. Dies ist im DIAMONDS-Deliverable D3.WP2 beschrieben.

Die Technik des Behavioural Fuzzings wurde in einem Prototyp implementiert, der im DIAMONDS-Deliverable D3.WP3 beschrieben ist.

5.1.2 Fuzzing-Bibliothek Fuzzino

Fuzzing ist als Technik für Sicherheitstests verbreitet. Es gibt sowohl kommerzielle als auch Open Source-Fuzzing-Werkzeuge und Fuzzing-Frameworks zur Entwicklung von protokollspezifischen Fuzzing-Werkzeugen. Allen diesen Werkzeugen ist gemein, dass sie eigenständig arbeiten. Dies erfordert es, dass ein solches Werkzeug für das zu testende System konfiguriert wird. Dies bedeutet insbesondere für die Modellierung des Protokolls einen hohen Aufwand, wenn das Protokoll des zu testenden Systems nicht vom Werkzeug bereits unterstützt wird. Daher wäre es hilfreich, wenn bestehende Testwerkzeuge, die für funktionales Testen eingesetzt werden, auch für Sicherheitstests mit Fuzzing eingesetzt werden könnten. Diese Werkzeuge können bereits die Protokoll-Modelle des zu testenden Systems verarbeiten, wodurch der zusätzliche Aufwand für Modellierung des Protokolls in einem separaten Fuzzing-Werkzeug entfallen könnte.

Die im DIAMONDS-Projekt entwickelte Fuzzing-Bibliothek „Fuzzino“ ermöglicht es, bestehende Testwerkzeuge für funktionales Testen auch für Fuzz Testen einsetzen zu können. In Form einer Bibliothek stellt sie Fuzzing-Heuristiken der verbreiteten Open Source Fuzzing Frameworks Peach[2] und Sulley[3] bereit. Fuzzino stellt somit einen Testdatengenerator für Fuzz Tests dar, der bewährte Fuzzing-Heuristiken für weitere Testwerkzeuge nutzbar macht. Fuzzino stellt dazu verschiedene Schnittstellen zur Verfügung. Um unabhängig von der Programmiersprache des Testwerkzeugs, das die Fuzzing Library nutzen möchte, den

[2] Peach Fuzzing Platform http://peachfuzzer.com/

[3] Sulley Fuzzing Framework https://github.com/OpenRCE/sulley

Zugriff zu ermöglichen, wurde ein auf XML basierendes Datenaustauschformat definiert. Mit diesem können Abfragen an Fuzzino formuliert werden, die Fuzzing-Bibliothek antwortet dann selbst wiederum mit einer XML-Datei, die zur Abfrage passende Fuzz-Testdaten enthält. Der Nutzer übergibt dazu eine einfache Beschreibung des Datenformats. Für viele auf String basierte Datenformate, wie bspw. Dateinamen, Zeitformate oder Befehlszeilenkommandos, werden bereits Datenformate zur Verfügung gestellt. Zusätzlich kann der Nutzer gültige Werte, die mutiert werden sollen, Fuzzino zur Verfügung stellen und ggf. Fuzzing-Heuristiken auswählen, die bei der Erzeugung der Fuzz-Testdaten verwendet werden sollen. Andererseits wird der mit Fuzzing unerfahrene Nutzer von Fuzzino unterstützt, indem für ein gegebenes Datenformat geeignete Fuzzing-Heuristiken ausgewählt werden. Durch die Nutzbarmachung von Fuzzing für bestehende Testwerkzeuge und die Unterstützung des Nutzers bei der Auswahl geeigneter Fuzzing-Heuristiken wird die Einführungsschwelle für Fuzz Testing deutlich gesenkt. Details zur Implementierung der Fuzzing Library finden sich im DIAMONDS-Deliverable D3.WP3.

5.1.3 Traceability Platform für den risikobasierten Test (RISKTest)

Um eine sicherheitsorientierte Risikoanalyse, die Testentwicklung und das Testmanagement sinnvoll in einer Werkzeugumgebung integrieren zu können, haben wir die Plattform für werkzeugübergreifendes Abhängigkeitsmanagement *RISKTest* entwickelt, die speziell auf die Anforderungen des risikobasierten Testens zugeschnitten ist. Mit *RISKTest* lassen sich die Artefakte der Risikoanalyse, der Testspezifikation und des Testmanagements miteinander in Beziehung setzen. So können wir, ausgehend von den in der Risikoanalyse identifizierten Schwachstellen, diesen die Testmodelle zuordnen, die als Basis für die Testgenerierung eingesetzt werden sollen. Die Werkzeuge, die anschließend die Testgenerierung automatisieren, sorgen dafür, dass auch zwischen dem Testmodell und den generierten Testfällen die notwendigen Links erzeugt werden, sodass sich die Testfälle auf die ihnen zugrundeliegenden, potenziellen Schwachstellen zurückführen lassen. Dies erlaubt es uns, die Abdeckung der potenziellen Schwachstellen durch Testfälle im Verlauf der Entwicklung automatisch ermitteln und kontrollieren zu können. Da wir darüber hinaus auch die Testergebnisse mit den Testfällen verlinkt haben, können wir zusätzlich nach jeder Testausführung die Testergebnisse auf die den Test motivierenden Schwachstellen zurückführen. Für jede Schwachstelle lässt sich so ermitteln, wie intensiv und mit welchem Ergebnis nach ihr gesucht wurde. Betrachtet man mehrere Iterationen der Testausführung, lassen sich auf Basis des in *RISKTest* verwalteten Abhängigkeitsnetzes weitergehende Analysen durchführen, die weitere Rückschlüsse auf die Systemqualität und die IT-Sicherheitseigenschaften des Systems möglich machen. *RISKTest*, die in DIAMONDS entwickelte Plattform für Abhängigkeitsmanagement in modellbasierten Testumgebungen, basiert auf der Eclipse-Plattform und auf der durch die Firma ITEMIS im Projekt Verde [45] entwickelten Abhängigkeitsmanagementlösung CREMA [6]. *RISKTest* integriert derzeit das Risikomodellierungswerkzeug CORAS [5], das UML-Werkzeug PAPYRUS [11], die Anforderungsentwicklungsplattform ProR [36] sowie das kommerzielle Testwerkzeug TTworkbench [43]. Zu den Besonderheiten der Plattform gehören, neben ihrer Spezialisierung auf das risikobasierte Testen, die automatische Erzeugung der Abhängigkeitsinformationen im Zuge der Testgenerierung und die Möglichkeit, direkt in und mit den Oberflächen der Originalwerkzeuge zu arbeiten. Alle wichtigen Funktionen der Plattform, wie beispielsweise das Erstellen der Abhängigkeiten und das Navigieren zwischen Modellelementen mit Hilfe von Abhängigkeiten, sind in allen Benutzeroberflächen der Originalwerkzeuge erreichbar und können durch einfache Interaktion über die Kontextmenüs ausgelöst und visualisiert werden. So können die Entwickler und Testexperten mit *RISKTest* weiterhin in ihrer gewohnten Werkezugumgebung arbeiten, ohne dass sie für das Abhängigkeitsmanagement auf die ihnen bekannte, native Darstellung der Entwicklungsartefakte verzichten müssen.

5.1.4 Kompositionelle Risikoanalyse

Die Durchführung einer Risikoanalyse kann mit erheblichem Aufwand und Schwierigkeiten etwa bei der Einschätzung von Wahrscheinlichkeitswerten verbunden sein. Das gilt besonders für komplexe Systeme. Oftmals bestehen diese aus mehreren Komponenten von verschiedenen Herstellern. Eine Idee, um dennoch

eine akkurate Risikoanalyse bei vertretbaren Kosten zu ermöglichen, ist die kompositionelle Risikoanalyse. Dabei werden die einzelnen Komponenten jeweils für sich bezüglich ihrer Risiken analysiert. Die Resultate werden dann benutzt, um aus deren Kombination die Risiken eines aus den Komponenten bestehenden Systems analysieren zu können, ohne sich mit den Details der einzelnen Komponenten auseinandersetzen zu müssen. Im Idealfall muss so jede Komponente nur einmal bezüglich ihrer Risiken analysiert werden, und die Resultate können für alle die Komponente nutzenden Systeme wiederverwendet werden.

Im Rahmen von DIAMONDS wurden durch Fraunhofer FOKUS Konzepte und Methoden entwickelt, um in einer erweiterten Form der modellbasierten CORAS Methode auch eine kompositionelle Risikoanalyse zu ermöglichen [DIA-24].

Dabei werden zunächst einzelne Komponenten mit der CORAS-Methode analysiert und aus den Resultaten wiederverwendbare *Threat Interfaces* generiert.

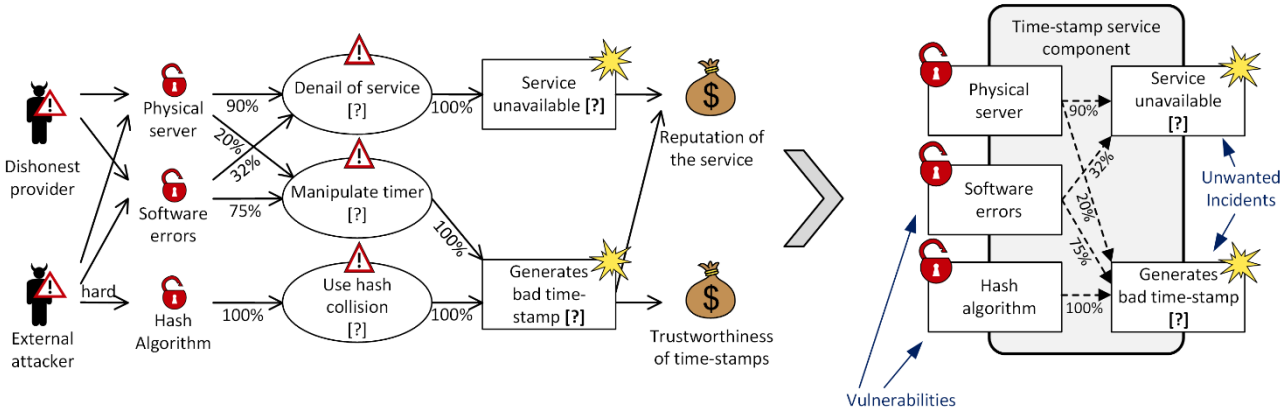

Abbildung 4: Generierung eines *Threat Interface* (rechts) aus einem konventionellen CORAS *Threat Diagram* (links) für eine einzelne Komponente

Mithilfe der *Threat Interfaces* wird dann in einem *Threat Composition Diagram* das Gefahrenpotenzial für das Gesamtsystem modelliert. Konkret werden die Auswirkungen von *Unwanted Incidents* auf *Vulnerabilities* in anderen Komponenten analysiert und durch Relationen sowie *Gates*, welche die Art der Abhängigkeit abbilden, in einem gerichteten Graphen abgebildet. Hat ein *Incident* verschiedene Trigger mit unterschiedlichen Abhängigkeiten, so werden diese in tabellenartigen *Dependency Sets* im *Threat Composition Diagram* erfasst. Durch Anwendung von Rechenregeln aus der Wahrscheinlichkeitstheorie lassen sich dann Wahrscheinlichkeiten für abhängige Ereignisse berechnen.

Das Abbilden von Abhängigkeiten zwischen ungewollten Ereignissen mit Gates zur Berechnung von Wahrscheinlichkeiten ist nicht neu, es wird beispielsweise in der *Fault Tree Analysis* FTA und *Event Tree Analysis* ETA verwendet. Nur die Ereignisse (Fehler) zu betrachten, wie es in ETA und FTA geschieht, ist jedoch für eine kompositionelle Risikoanalyse nicht zielführend, da sich mögliche Abhängigkeiten zwischen den Ereignissen verschiedener Komponenten nur bei erneuter Betrachtung der Komponenten selbst aufspüren lassen. In der erweiterten CORAS-Methode zur kompositionellen Risikoanalyse hingegen werden neben den Ereignissen auch die Stellen erfasst, an denen eine Komponente von anderen beeinflusst werden könnte. Die *Vulnerabilities* eines *Threat Interface* sind quasi die Eingänge für mögliche Trigger-Relationen. Mit ihnen wird es möglich, Verbindungen zwischen den Risiko-Analyse-Artefakten verschiedener Komponenten herzustellen, ohne dass dazu zusätzliche Informationen zu den internen Details einer Komponente (wie bestimmte *Unwanted Incidents* ausgelöst werden können) benötigt würden.

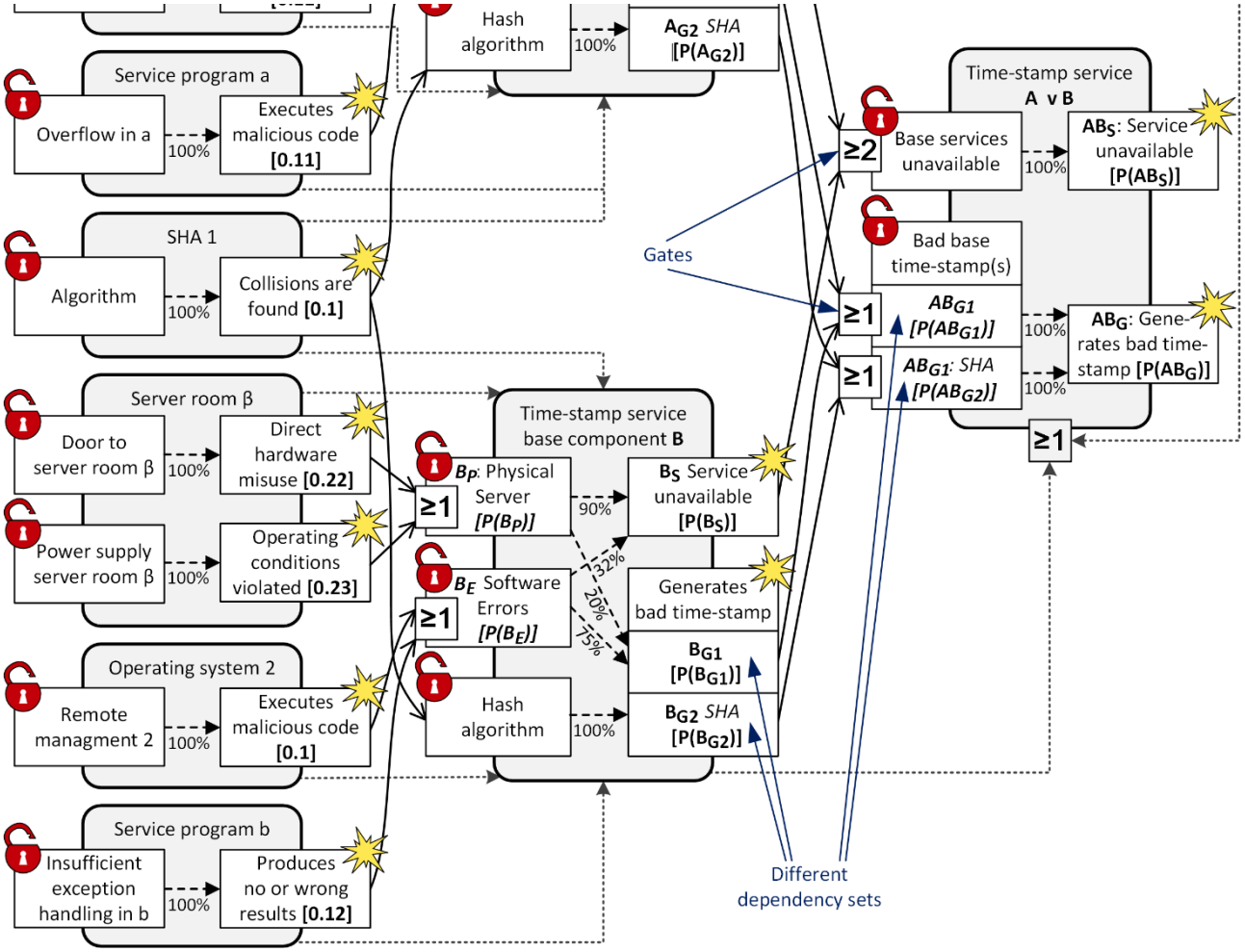

Abbildung 5: Ausschnitt aus einem *Threat Composition Diagram* mit *Gates* und *Dependency Sets*

5.1.5 Kombinierte TMSR und RMST Methode

Es gibt im Wesentlichen zwei verschiedene sinnvolle Möglichkeiten, Risikoanalyse und Sicherheitstesten miteinander zu verbinden: Bei *Test-driven Model-based Security Risk Analysis* (TMSR) wird das Testen eingesetzt, um Risiken zu identifizieren und um sie besser einschätzen zu können. Bei *Risk-driven Model-based Security Testing* (RMST) wird hingegen die Risikoanalyse genutzt, um die kritischen Stellen zu identifizieren, welche getestet werden sollen, und um die Testfälle zu priorisieren. Bei ersterem Verfahren wird die Risikoanalyse durch das Testen verfeinert und unterstützt, beim zweiten Verfahren ist es umgekehrt.

Im Rahmen von DIAMONDS wurde eine kombinierte iterative Methode aus beiden Verfahren entwickelt, um die Risikoanalyse und IT-Sicherheitstests so zu einem Kreislauf zu verbinden, dass eine inkrementelle Verbesserung des Gesamtbilds von der Sicherheit des zu analysierenden Systems entsteht.

Begonnen wird dabei mit der Risikoanalyse, und die Artefakte der Risikoanalyse werden auch als die finalen Ergebnisse interpretiert. Basierend auf den Ergebnissen der initialen Risikoanalyse wird das *Threat Scenario* identifiziert, das am kritischsten erscheint und für das die höchste Unsicherheit angenommen wird. Anschließend werden nur dafür Testfälle generiert. Um diesen Schritt zu unterstützen, wurde ein Mapping zwischen CORAS Risk Analysis Resultaten und Security Test Pattern erstellt.

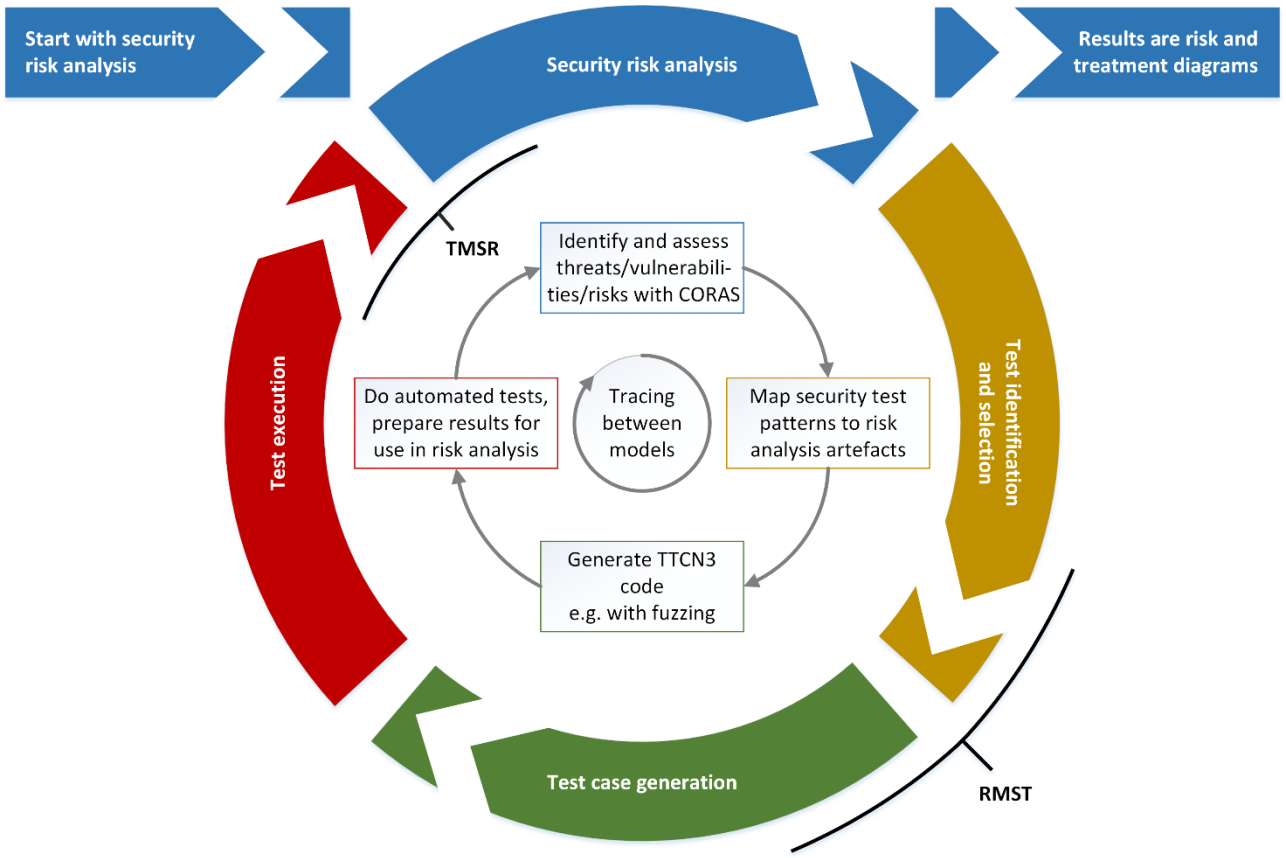

Abbildung 6: Prozesse der kombinierten TMSR und RMST Methode

Nach der Ausführung der Testfälle werden die Testergebnisse in die Risikoanalyse einbezogen. Dabei können neue Schwachstellen entdeckt werden und es können die Wahrscheinlichkeitswerte dafür, dass bestimmte *Unwanted Incidents* auftreten, durch die objektiven Testresultate korrigiert werden. Dazu wird eine neuerliche Risikoanalyse durchgeführt.

Basierend auf der durch Testergebnisse verbesserten Risikoanalyse kann die nächste kritische Stelle, an der es sich nun am meisten lohnen würde, weitere Tests durchzuführen, ausgewählt werden, und eine neue Runde mit Testen und Rückführung der Testergebnisse in die Risikoanalyse kann begonnen werden.

5.1.6 Security Testing Improvement Profile (STIP)

Im DIAMONDS-Projekt wurden neben den hier vorgestellten Techniken eine Vielzahl weiterer Techniken entwickelt und in den Fallstudien validiert. Um den Einfluss der Techniken auf die Reife der Testprozesse und speziell der modellbasierten IT-Sicherheitstestprozesse bewerten zu können, haben wir mit dem **Security Testing Improvement Profile (STIP)** ein Bewertungsprofil entwickelt, mit dem sich ebensolche Testprozesse systematisch evaluieren lassen. Vordergründig betrachtet gibt es bereits eine ganze Reihe von Methoden, um die Reife und Qualität von Testprozessen zu beurteilen. Die bekanntesten Vertreter sind TPI (Test Process Improvement) [40]bzw. dessen Nachfolger TPI NEXT und die Test Maturity Model Integration (TMMI)[12]. Obwohl wir davon ausgehen, dass diese Bewertungsverfahren auch auf die Prozesse für den IT-Sicherheitstest angewendet werden können, fehlt ihnen eine detaillierte Analyse der spezifischen Techniken aus dem Bereich des IT-Sicherheitstests, wie beispielsweise die Integration einer detaillierten Risikoanalyse für IT-Sicherheitsrisiken und die Betrachtung von speziellen

Sicherheitsprüftechniken wie dem Fuzzing (siehe vorausgehende Abschnitte). Das von uns entwickelte Bewertungsverfahren STIP kann sowohl eigenständig wie auch in Kombination mit den oben genannten etablierten Bewertungstechniken eingesetzt werden. Angelehnt an TPI und TMMI haben wir für das IT-Sicherheitstesten Schlüsselbereiche, sog. STIP Key Areas, identifiziert. Insgesamt haben wir für das DIAMONDS-Projekt neun Schlüsselbereiche definieren können, die wir als relevant für modellbasierte IT-Sicherheitstests angesehen haben und die jeweils eigenständig bewertet werden können. Die Schlüsselbereiche definieren wichtige Aspekte oder Aktivitäten in einem IT-Sicherheitstestprozess, wie z.B. die Reife der IT-Sicherheitsrisikoanalyse, den Grad der Automatisierung bei der Testgenerierung und -Ausführung, die Integration der Werkzeuge oder die Reife spezifischer Sicherheitstesttechniken wie z.B. das Fuzzing. Jedem Schlüsselbereich sind vier Bewertungsstufen zugeordnet, die hierarchisch angeordnet sind. Jede Stufe definiert einen spezifischen Reifegrad für den Schlüsselbereich, der durch eine informelle Beschreibung charakterisiert ist. Jede Stufe lässt sich von den anderen Stufen abgrenzen, sodass einem konkreten Testprozess genau eine Reifegradstufe pro Key Area zugeordnet werden kann. Tabelle 2 zeigt die Bewertungsstufen für den Schlüsselbereich *Fuzzing.*

STIP-Schlüsselbereich	**Fuzzing**
Level 1	Random data fuzzing
Level 2	Model-based data fuzzing
Level 3	Model-based evolutionary fuzzing
Level 4	Model-based data and behavioural fuzzing

Tabelle 3 Der STIP-Schlüsselbereich Fuzzing mit seinen Bewertungsstufen

Je höher der Reifegrad, desto besser bewerten wir den Prozess im Hinblick auf seine Leistungsfähigkeit bzw. Ergebnisqualität in dem gegebenen Schlüsselbereich. Für eine vollständige Prozessbewertung wird ein Prozess hinsichtlich aller neun Schlüsselbereiche bewertet. Eine vollständige Liste der STIP-Schlüsselbereiche inklusive der Definition und Beschreibung der einzelnen Reifegrade findet sich in der STIP-Darstellung [15] auf den Webseiten des DIAMONDS-Projekts.

In DIAMONDS haben wir STIP eingesetzt, um acht der DIAMONDS-Fallstudien zu bewerten und den Fortschritt zu messen, die die Fallstudien im Verlauf des DIAMONDS-Projekts gemacht haben. Zu diesem Zweck haben wir für jede Fallstudie zwei Bewertungen vorgenommen. Eine Bewertung, die die Fallstudien vor dem Start des Projekts DIAMONDS charakterisiert, und eine, die dieselbe Fallstudie zum Ende des Projekts nach Anwendung der DIAMONDS-Techniken betrachtet.

Abbildung 7 zeigt das Ergebnis der STIP-Evaluationen für eine DIAMONDS-Fallstudie. Der rot eingefärbte Bereich zeigt den Reifegrad der Fallstudie vor DIAMONDS und der blaue Bereich den Reifegrad zum Ende von DIAMONDS. Die Grafik zeigt deutlich, dass die bewertete Fallstudie insbesondere bei den Testtechniken zur Testgenerierung, beim Fuzzing sowie bei der Werkzeugintegration durch DIAMONDS profitieren konnte. Unter [15] finden sich weitere Bewertungsergebnisse der DIAMONDS-Fallstudien.

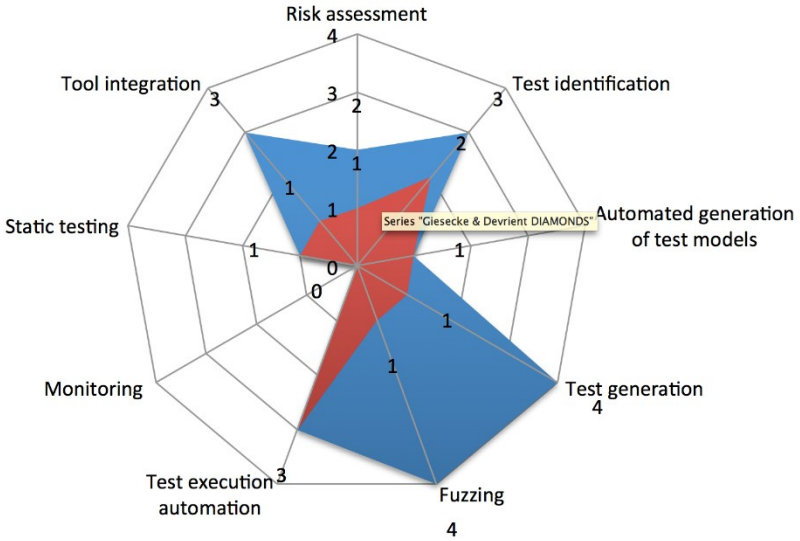

Abbildung 7: STIP Bewertung einer DIAMONDS-Fallstudie

5.2 VERWERTUNG DER ERGEBNISSE

	Ergebnisse (laut Arbeitsplan)	Verwertungsmöglichkeit, -aktivität nach Art und Wirkung/Nutzen	Zeithorizont/ geplante Realisierung	Erfüllungsgrad
1	Methode und prototypisches Werkzeug für den MBBF-Ansatz (WP2)	Langfristiger Aufbau von Kapazitäten (Know-how und Werkzeuge) im Bereich Modellbasiertes Security Testen	Bis 5 Jahre nach Projektende	Begonnen
2	Case Study Experience Report (Erfahrungen aus den Fallstudien werden in konsolidierter Form der ETSI zur Vorbereitung der Standardisierung Verfügung gestellt)	Transfer zu Nutzergruppen Vorbereitung der Standardisierung von Methoden und Techniken bei der ETSI	Bis Ende 2013	Begonnen

	Ergebnisse (laut Arbeitsplan)	Verwertungsmöglichkeit, -aktivität nach Art und Wirkung/Nutzen	Zeithorizont/ geplante Realisierung	Erfüllungsgrad
3	FOKUS Fuzzing-Bibliothek Fuzzino (WP3)	Kurzfristiger Aufbau von Security Testing-Reputation durch Open Source Lizensierung der Software	Bis Ende 2015	Umgesetzt
		Langfristiger Aufbau von Kapazitäten im Bereich Security Testing	Bis 5 Jahre nach Projektende	Begonnen
4	FOKUS RISKTest - Traceability Platform (WP3)	Langfristiger Aufbau von Kapazitäten im Bereich Security Test	Bis 5 Jahre nach Projektende	Begonnen
5	Risk-based Security Testing Method (WP4)	Grundlage für die Akquise von Folgeprojekten	Im Verlauf bzw. kurz nach Beendigung von DIAMONDS	Umgesetzt, das FP7-Projekt RASEN (http://www.rasenproject.eu/) wurde im Oktober 2012 gestartet. Weitere Projekte sind beantragt u.a. zum Thema Risikomanagement und Sicherheitstests für kritische Banken-infrastrukturen.
		Transfer zu Nutzergruppen Vorbereitung der Standardisierung von Methoden und Techniken bei der ETSI	Bis Ende 2015	Begonnen, ETSI Work Item zum Thema, Risk-based Security Testing Methodologies aufgesetzt und in Bearbeitung (siehe http://docbox.etsi.org/MTS/MTS/05-CONTRIBUTIONS/2013//MTS(13)000036_ETSI_EG_MTS-202793.zip)

	Ergebnisse (laut Arbeitsplan)	Verwertungsmöglichkeit, -aktivität nach Art und Wirkung/Nutzen	Zeithorizont/ geplante Realisierung	Erfüllungsgrad
6	Security Risk and Test Pattern Katalog (WP4)	Transfer zu Nutzergruppen	Im Verlauf bzw. kurz nach Beendigung von DIAMONDS	Begonnen, Ergebnisse werden im Rahmen der ETSI ISG ISI in die Standardisierung überführt.
7	Security Testing Terminology (WP5)	Standardisierung, Transfer zu Nutzergruppen	Im Verlauf bzw. kurz nach Beendigung von DIAMONDS	Begonnen, Terminologie wird derzeit im Rahmen der ETSI MTS-Security erarbeitet
8	Security Test Improvement Profile (WP4)	Standardisierung, Transfer zu Nutzergruppen	Kurz nach Beendigung von DIAMONDS	Begonnen, wird im Rahmen der ETSI MTS-Security Arbeiten als Teil der Fallstudienergebnisse publiziert.
		Langfristiger Aufbau von Kapazitäten im Bereich Security Test.	Bis 5 Jahre nach Projektende	Begonnen
9	Ausrichtung von Workshops und Tutorials, z.B. DIAMONDS STV Workshops auf der ICSSEA 2012, bzw. DIAMONDS Tutorial auf der ICST 13 (WP5)	Heranbildung von (wissenschaftlichem) Nachwuchs. Steigerung der wissenschaftlichen Konkurrenzfähigkeit	Im Verlauf bzw. kurz nach Beendigung von DIAMONDS	Umgesetzt: Tutorial bei der ICST 13 durchgeführt. SASSI 13 Workshop durchgeführt (http://s.fhg.de/Sassi 13)- RISKWorkshop auf der ICSST 13 geplant und durchgeführt (http://s.fhg.de/risk2013).

	Ergebnisse (laut Arbeitsplan)	Verwertungsmöglichkeit, -aktivität nach Art und Wirkung/Nutzen	Zeithorizont/ geplante Realisierung	Erfüllungsgrad
10	Veröffentlichungen auf akademischen und industrienahen Konferenzen und Workshops (z.B. MBTUC 12, SAM 12, SASSI13, RISK13) (WP5)	Steigerung der wissenschaftlichen Konkurrenzfähigkeit	Im Verlauf bzw. kurz nach Beendigung von DIAMONDS	Umgesetzt
11	Aufsetzen und Betreuen eines Industriebeirats (WP5)	Transfer von Know-how zu Nutzergruppen, potenziellen Kunden. Verstetigung eines Diskussionsrahmens zum Thema IT-Sicherheit in Deutschland Kontaktaufnahme mit potenziellen Verwertern/ Nutzern/Akteuren für die weitere Umsetzung der Ergebnisse auch in angrenzenden Wissensgebieten (spill-over)	Im Verlauf des DIAMONDS-Projekts und in den 5 Jahren danach	Umgesetzt, der Beirat existiert und es wird über eine Verstetigung des Zusammenhangs diskutiert. Mit Wincor-Nixdorf ist ein gemeinsames Forschungsprojekt beantragt
12	Gründung SOSO und Beitritt zur Cyber Allianz (WP5)	Transfer von Know-how zu Nutzergruppen, potentiellen Kunden. Verstetigung eines Diskussionsrahmens zum Thema IT-Sicherheit in Deutschland	Im Verlauf des DIAMONDS-Projekts und in den 5 Jahren danach	Teilweise umgesetzt

Tabelle 4 Verwertung FhG FOKUS

5.3 VORAUSSICHTLICHER NUTZEN

Fraunhofer FOKUS konnte sich durch das Projekt eine längerfristige Basis für die Etablierung von industrienahen Dienstleistungen und Produkten im Bereich Sicherheitstesten erarbeiten und ist nun dabei, die Ergebnisse zu industrialisieren. Projektergebnisse wie RISKTest, Fuzzino und die im Projekt erarbeitete MBBF-Methodik werden auf Messen und industrienahen Workshop und so einem breiterem industriellen Publikum vorgestellt. Fraunhofer FOKUS hat die technischen und wissenschaftlichen Ergebnisse aktiv über Konferenzen, Workshops und in Publikationen verbreitet und darüber das wissenschaftliche Profil des Instituts und einzelner Mitarbeiter im Bereich modellbasiertes Testen bzw. IT-Sicherheitstesten erweitern können. Um die nachhaltige Bereitstellung der DIAMONDS-Ergebnisse sicherzustellen, werden relevante

Ergebnisse des DIAMONDS-Projekts nach wie vor durch Fraunhofer FOKUS in Standardisierungsgremien eingebracht. Diese Form der Ergebnisverbreitung in Standardisierungsgremien wie der ETSI stellt für Fraunhofer FOKUS eine wichtige Grundlage dar, um die Ergebnisse längerfristig und nachhaltig einer Industrialisierung zuführen zu können und die industrielle Beratungskompetenz von Fraunhofer in diesen Bereichen zu stützen.

6. GIESECKE & DEVRIENT, FÖRDERKENNZEICHEN 01 IS 10 031 B

6.1 ERREICHTE ERGEBNISSE

Giesecke & Devrient ist Weltmarktführer bei den Banknotenbearbeitungssystemen. Solche Systeme werden von Landes- und Geschäftsbanken eingesetzt, um Banknoten zu bearbeiten. Dabei stehen Sicherheitsanforderungen im Vordergrund. Diese Anforderungen werden sich in den kommenden Jahren durch stärkere Vernetzung der Geräte untereinander und mit externen Systemen verändern und weiter an Bedeutung gewinnen. Da diese Geräte eine lange Lebenszeit haben und aktuelle Herangehensweisen keine Lösung der Probleme darstellen, hatte Giesecke & Devrient besonderes Interesse an der Entwicklung von neuen Technologien in diesem Umfeld.

Zu diesem Zweck hat Giesecke & Devrient eine Fallstudie zur Verfügung gestellt, an der die neuen Ideen ausprobiert und deren praktische Verwendung nachgewiesen werden kann. Die Fallstudie wurde so konzipiert, dass sie auch ohne kostspielige Hardware auskommt. Mehrere Softwarekomponenten wurden in virtuellen vernetzten Umgebungen zur Verfügung gestellt. Um die Tests automatisiert ausführen zu können, wurde eine verteile Testumgebung auf Basis von TTCN-3 in die Studie integriert. Giesecke & Devrient hat die Anforderung an das Testvorgehen und die Methoden definiert und nach einer ersten Überprüfung diese verfeinert oder präzisiert.

Die Studie wurde gemeinsam mit Fraunhofer FOKUS und Testing Technologies bearbeitet. Dabei wurden die Ergebnisse der Arbeitspakete an der Fallstudie gespiegelt bzw. die Methodik während der Durchführung der Studie erarbeitet. Die Themen des modellbasierten Verhaltens-Fuzzing wurden durch die Fallstudie vorangetrieben. Eine besondere Anwendungsform, die des Online-Fuzzing, wurde speziell auf Grund der Anforderungen der Fallstudie entwickelt. Sie umgeht unproduktive Totzeiten beim Testen unserer Banknotenbearbeitungsmaschinen durch Ausnutzung der maximalen Fuzz-Sequenz, wenn diese keine Fehler liefert.

Da Giesecke & Devrient schon lange den TTCN-3 Standard für die Testautomatisierung einsetzt, lag das Augenmerk auf die Wiederverwendung der Technologie auch für die Prüfung von Sicherheitsanforderungen. Damit sich die Erweiterung auch im Standard wiederfindet und die bisher proprietäre Lösung standardkonform wird, plant Giesecke & Devrient, als ETSI-Mitglied diese Aktivitäten weiter zu unterstützen.

Insbesondere hat sich Giesecke & Devrient auf die Methodik konzentriert. Als erstes Ergebnis ist die Integration der Methode in den Entwicklungsprozess zu nennen. Weitere Geschäftsbereiche von Giesecke & Devrient sind an der Methode interessiert oder schon aktuell dabei sie zu adaptieren.

Insgesamt wird Giesecke & Devrient von den Ergebnissen wie der Fuzzing-Bibliothek Fuzzino, der Traceability Platform RISKTest und dem Risk-based Security Testing in der Zukunft profitieren.

6.2 VERWERTUNG DER ERGEBNISSE

6.2.1 Interne Verwertung

Giesecke & Devrient konnte die Ergebnisse (Werkzeuge und Methoden) in die eigene Entwicklungslandschaft integrieren und wird damit einen technologischen Vorsprung bei den eigenen Produkten behalten. Dabei wird STIP (Security Testing Improvement Profile), ein wichtiges Ergebnis des Projekts, uns bei den weiteren Verbesserungen anleiten.

Die von Giesecke & Devrient durchgeführten bzw. geplanten Verwertungsmaßnahmen werden in der folgenden Tabelle noch einmal detailliert dargestellt:

	Ergebnisse (laut Arbeitsplan)	Verwertungsmöglichkeit, -aktivität nach Art und Wirkung/Nutzen	Zeithorizont/ geplante Realisierung	Erfüllungsgrad
1	Methode und prototypisches Werkzeug für den MBBF-Ansatz (WP2)	Langfristiger Aufbau von Kapazitäten (Know-how und Werkzeuge) im Bereich Modellbasiertes IT-Sicherheitstesten. Verankerung in den Entwicklungsprozessen des Unternehmens.	Bis 5 Jahre nach Projektende	Teilweise umgesetzt
2	FOKUS Fuzzing-Bibliothek (WP3)	Langfristiger Einsatz der Methodik für die Entwicklung und den Test von neuen Produkten.	Bis 5 Jahre nach Projektende	Teilweise umgesetzt, Wissen wird bei neuen Produkten angewandt
3	FOKUS Traceability Platform (WP3)	Langfristiger Einsatz der Methodik für die Entwicklung und den Test von neuen Produkten.	Bis 5 Jahre nach Projektende	Begonnen
4	Risk-based Security Testing Method (WP4)	Langfristiger Einsatz der Methodik für die Entwicklung und den Test von neuen Produkten.	Bis 5 Jahre nach Projektende	Teilweise umgesetzt, Testmethode in neuem Produkt pilotiert

Tabelle 5 Interne Verwertung Giesecke & Devrient

6.2.2 Externe Verwertung

Eine externe Verwertung der Ergebnisse findet durch Giesecke & Devrient nur mittelbar statt. Wir werden von der weiteren Verbreitung der Erkenntnisse und den Standardisierungsaktivitäten profitieren. Insbesondere hervorzuheben ist der Security Cluster, der in und um München eine Plattform für DIAMONDS-Ergebnisse für den Transfer zu Nutzergruppen darstellt. Dieser Cluster wurde von Giesecke & Devrient initiiert und soll in den Folgejahren weiter an Bedeutung gewinnen.

Die von Giesecke & Devrient durchgeführten bzw. geplanten Verwertungsmaßnahmen werden in der folgenden Tabelle noch einmal detailliert dargestellt:

	Ergebnisse (laut Arbeitsplan)	**Verwertungsmöglichkeit, -aktivität nach Art und Wirkung/Nutzen**	**Zeithorizont/ geplante Realisierung**	**Erfüllungsgrad**
1	Case Study Experience Report (Erfahrungen aus der Fallstudie werden in konsolidierter Form der ETSI zur Vorbereitung der Standardisierung Verfügung gestellt)	Transfer zu Nutzergruppen Vorbereitung der Standardisierung von Methoden und Techniken bei der ETSI für die Zertifizierung	Bis Ende 2013	Umgesetzt
2	Security Risk and Test Pattern Catalogue (WP4)	Vorbereitung für Standardisierung	Im Verlauf bzw. mittelfristig nach Beendigung von DIAMONDS	Teilweise umgesetzt, Ergebnisse werden im Rahmen der ETSI ISG ISI in die Standardisierung überführt.
3	Security Testing Terminology (WP5)	Standardisierung, Transfer zu Nutzergruppen	Im Verlauf bzw. kurz nach Beendigung von DIAMONDS	Begonnen, Terminologie wird derzeit im Rahmen der ETSI MTS-Security erarbeitet
4	Veröffentlichungen auf industrienahen Konferenzen und Workshops (z.B. MBTUC 12, SAM) (WP5)	Dissemination der Projektergebnisse	Im Verlauf bzw. kurz nach Beendigung von DIAMONDS	Umgesetzt
5	Unterstützung bei der Betreuung eines Industriebeirats (WP5)	Transfer von Know-how zu Nutzergruppen, potentiellen Partnern. Verstetigung eines Diskussionsrahmens zum Thema IT-Sicherheit in Deutschland	Im Verlauf des DIAMONDS-Projekts und in den 5 Jahren danach	Teilweise umgesetzt, der Beirat existiert und es wird über eine Verstetigung des Zusammenhangs diskutiert.
6	Beitritt zum Bavarian IT Security and Safety Cluster (WP5)	Transfer von Know-how zu Nutzergruppen. Verstetigung eines Diskussionsrahmens zum Thema IT-Sicherheit in Deutschland	Im Verlauf des DIAMONDS-Projekts und in den 5 Jahren danach	Teilweise umgesetzt, Präsentation der Ergebnisse auf dem Forum, Vertiefung wird fortgesetzt

Tabelle 6 Externe Verwertung Giesecke & Devrient

6.3 VORAUSSICHTLICHER NUTZEN

Giesecke & Devrient wird mit den Ergebnissen aus DIAMONDS seine marktführende Position bei den Banknotenbearbeitungssystemen halten oder weiter ausbauen können. Durch die konkrete Umsetzung der Methoden können Investitionen in die Testautomatisierung weiter genutzt werden. Dadurch entsteht ein finanzieller Vorteil gegenüber der Konkurrenz.

Die Verbreitung der Technologie unter den Nutzern und Anwendern führt zu einer weiteren Bekanntmachung der Möglichkeiten Sicherheitsrelevante Anforderungen zu testen. Diese wiederrum erleichtern die Akzeptanz von Produkten, die diesem Standard folgen und ihn erfolgreich umsetzten. Auch dafür wird sich Giesecke & Devrient einsetzen.

Insgesamt plant Giesecke & Devrient durch konsequente Nutzung der Ergebnisse in den Geschäftsgebieten weitere Nutzer seiner Technologien gewinnen.

7. DORNIER CONSULTING, FÖRDERKENNZEICHEN 01 IS 10 031 C

Die stetig steigende Komplexität von Komponenten in Kraftfahrzeugen und deren Integration stellt Automobilhersteller vor zunehmende Herausforderungen. Verschärft wird diese Situation noch zusätzlich durch immer kürzer werdende Entwicklungszyklen bei Zulieferer-Komponenten, v.a. im Bereich des Infotainments und der damit in Zusammenhang stehenden Services für den Endnutzer. Eine ähnliche Entwicklung ist auch im Bereich der Fahrerinformations- und Assistenzsysteme zu beobachten. Diese Umstände erfordern es nun, dass der Fahrzeughersteller möglichst effiziente Systemdesign-Methoden, wie z.B. das Modellieren auf Systemebene in UML/SysML, einsetzt. Das Systemmodell stellt die Grundlage dar, um darauf aufbauend formale Testmethoden zu entwickeln und durchzuführen, die eine schnelle und kosteneffiziente Absicherung der vernetzten Steuergeräte (Electronic Control Unit, ECU) ermöglichen. Das modellbasierte Testen stellt dabei einen der wichtigsten Ansätze dar, wie eine modellbasierte Systemabsicherung effizient realisiert werden kann.

Das Thema IT-Sicherheit wird, insbesondere mit Blick auf die steigende Vernetzung von Fahrzeugen sowie auf Grund der steigenden Durchdringung moderner Automobile mit IT-Technologie, auch für die Automobilindustrie zu einem immer brisanteren Thema. Dabei stellt sich u.a. die Frage, wie die IT-Sicherheit von Fahrzeugsystemen zukünftig analysiert und verifiziert werden kann. Es gibt zwei zentrale Aspekte, die hier eine wichtige Rolle spielen:

- Zur Bestimmung des Gefährdungspotenzials müssen die zu testenden Komponenten einer Risikoanalyse unterzogen werden, die neben dem Aspekt der funktionalen Sicherheit einen Schwerpunkt auf die IT-Sicherheit legt.
- Es muss geklärt werden, inwieweit bereits bekannte und etablierte (Test-)Methoden aus dem Bereich der IT-Sicherheit in die automobile Systementwicklung übertragen werden können, um dort in domänen-spezifischer Form zur Anwendung gebracht zu werden.

Untersuchungen, die ein realistisches Bedrohungsszenario im Automobilumfeld betrachten, sind im Gegensatz zu Studien in anderen Technologiebereichen (wie z.B. dem Internet) noch in der Minderheit. Einen wesentlichen Beitrag leisteten hierbei Koscher et al. [27] mit ihrer Fallstudie, die gezeigt hat, wie ein Angreifer die Kontrolle über nahezu alle ECUs eines Autos übernehmen und nach seinem Willen beeinflussen kann. Durch die vollständige Vernetzung der Onboard-Komponenten über Bussysteme (je nach Art und Wichtigkeit der Komponente in der Regel Low-Speed CAN, High-Speed CAN, LIN oder MOST) ist nach Infiltrierung einer einzelnen Einheit unter Umständen der Zugriff auf das Gesamtsystem oder zumindest dessen Störung möglich, wobei bestehende Sicherheitssysteme (z.B. das Body Control Module, BCM) umgangen werden konnten. Es wurde u.a. eine Vorgehensweise beschrieben, wie der

Schutzmechanismus, der ein unautorisiertes Firmware-Upgrade einer ECU verhindern soll, umgangen werden konnte, um diese ECU (im Beispiel die Telematik-Unit) mit einer neuen Firmware zu versehen, die Schadcode enthielt. Da diese ECU zudem auch noch eine Überbrückung der beiden CAN-Busse erlaubte, war ein Zugriff auf sicherheitskritische ECUs des High-Speed CANs über Komponenten des Low-Speed CANs möglich. Außerdem wurde gezeigt, dass viele ECUs anfällig gegenüber Fuzzing-Attacken sind. Dabei werden Nachrichten auf den CAN-Bus gelegt, deren Pakete zwar einen standard-konformen Aufbau aufweisen, die konkret übertragenen Werte sind aber bewusst zufällig gewählt. Viele Komponenten reagierten auf solche Nachrichten mit fehlerhaftem Verhalten oder sogar mit einem Totalausfall, da die interne Implementierung nicht robust genug war, um solche „Fehleingaben“ abzufangen. Kritisch daran ist, dass ein Angreifer mit dieser Technik relativ einfach und sehr effektiv Angriffe durchführen kann, da er diese nicht auf eine bestimmte ECU anpassen muss und auch sonst kein explizites Spezialwissen über die betroffene Komponente benötigt.

Im Rahmen von DIAMONDS hat Dornier Consulting eine Fallstudie zur Verfügung gestellt. Diese Fallstudie bestand aus einem Infotainmentsystem eines großen Automobilherstellers. Der Fokus lag bei der Studie auf dem Konnektivitätsmodul, welches dem Infotainmentsystem die Verbindung mit externen Geräten ermöglicht. Im speziellen wurde die Bluetooth-Verbindung untersucht, welche Mobiltelefone und Headsets in das Infotainmentsystem integriert. Das Konnektivitätsmodul ist des Weiteren mit dem Automobilbus verbunden und leitet bei Bedarf die Befehle vom Telefon an die entsprechenden Steuergeräte (ECU) weiter. Somit besteht die Möglichkeit, extern über eine Bluetooth-Verbindung die Steuergeräte eines modernen Automobils zu manipulieren.

Dornier Consulting hat, stellvertretend für den Automobilbereich, eine Fallstudie für DIAMONDS entwickelt, welche die zuvor hier allgemein beschriebenen Risiken beim Einsatz von vernetzten Steuergeräten und darauf aufbauenden Telematik-Anwendungen und -Dienstleistungen berücksichtigt. Die dadurch entstehende Komplexität stellt auch eine zusätzliche Herausforderung für System- und Integrationstests dar, die ein Hersteller für das Gesamtsystem durchführen muss. Dieses besteht aus einer Vielzahl von Komponenten, die von Zulieferern bezogen werden und die aus Sicht des OEMs eine Blackbox darstellen.

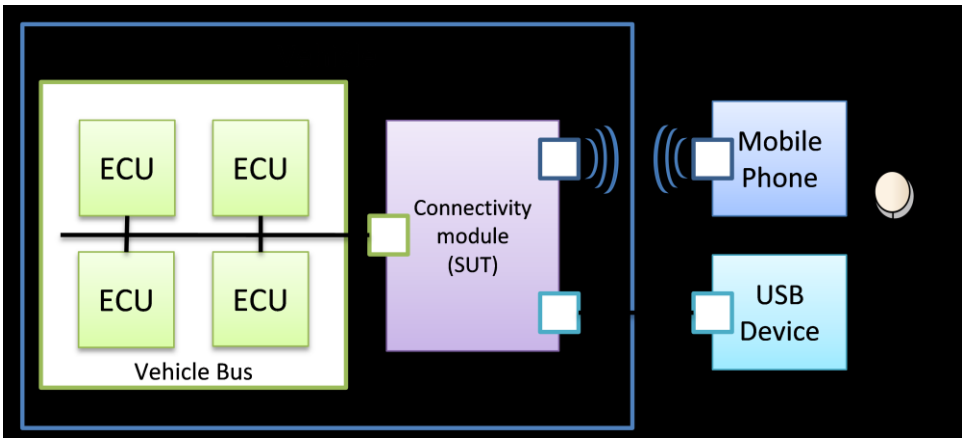

Abbildung 8: Schematische Übersicht über den Testfallaufbau

In Abbildung 8 ist das System Under Test (SUT) schematisch dargestellt. Konkret untersuchte die Fallstudie die Testbarkeit von Sicherheitsanforderungen am Beispiel des Speech Control-Moduls (Unit Under Test, UUT), welches u.a. die Verbindung zwischen Consumer-Endgeräten (Smartphones, MP3-Player, USB-Sticks) und dem Infotainment-Systems des Fahrzeugs herstellt. Diese Verbindung wird über ein entsprechendes Connectivity-Modul realisiert und kann zum einen, wie bei Mobiltelefonen und Smartphones

üblich, drahtlos über Bluetooth erfolgen oder über ein USB-Kabel bzw. über einen USB-Steckplatz direkt am Connectivity-Modul gelöst sein.

Das Infotainment-System besteht aus einer Vielzahl einzelner Electronic Control Units (ECUs), die über den CAN-Bus miteinander kommunizieren. Neben dem Connectivity-Modul zählen hierzu u.a.:

- Radio, Mediaplayer und Navigationssystem (Head Unit)
- weitere Cockpit-Instrumente (z.B. Radio-Steuerung über Knöpfe im Lenkrad, Displays im Bereich des Tachos)
- Klimaanlagensteuerung

Darüber hinaus sind diese Komponenten durch die Vernetzung über den CAN-Bus noch mit anderen, teils kritischen Systemen verbunden. Beispielsweise ist das Radio mit der Motorsteuerung gekoppelt, um z.B. in Abhängigkeit der Geschwindigkeit die Lautstärke der Wiedergabe zu erhöhen oder zu senken, damit Umgebungsgeräusche angemessen kompensiert werden können.

Der Testaufbau wird mit der eignen Werkzeugkette von Dornier Consulting angesteuert. Diese ermöglicht es, aus einem UML/SysML-Testmodell ausführbare Testfälle zu generieren. Diese Testfälle können nach der Generierung automatisch am zu testenden System ausgeführt werden. In folgender Abbildung, welche auf dem ITEA-2 Summit in Paris aufgenommen wurde, ist Dornier Consultings Testroboter zu sehen, der einen generierten Testfall an einer Testhardware ausführt.

Abbildung 9: Präsentation während des ITEA-2-Summits in Paris

7.1 ERREICHTE ERGEBNISSE

7.1.1 Risikoanalyse

Wie auch in anderen Testfällen wurde eine Risikoanalyse als zentrales Einstiegsvorgehen gewählt. Diese Risikoanalyse wurde mit der CORAS-Methode durchgeführt, welches von einem der Norwegischen Forschungspartnern entwickelt wird und auch im Rahmen dieses Forschungsprojekts weiterentwickelt wurde. Es ermöglicht eine abstrakte Betrachtung des Testsystems in Bezug auf mögliche Risiken. Als Ergebnis dieses Forschungsprojekts wurde eine Methodik entwickelt, diese Sicherheitsuntersuchung in numerischer Form in das Testmodell zu transferieren. Hierzu werden die einzelnen Pfade mit Wahrscheinlichkeiten bewertet. Hinzu kommen eine Einschätzung bezüglich des nötigen Aufwands einer erfolgreichen Attacke und eine Einschätzung der möglichen Folgen in monetärem Ausmaße. Diese Werte ergeben eine numerische Bewertung des Risikos, welche in das Testmodell übertragen werden kann. Somit ist es möglich, die manuell erstellten Risikoanalysen als Steuerungsgröße für die Testfallgenerierung zu benutzen. In Abbildung 10 ist eine durchgeführte Risikoanalyse abgebildet.

Zusätzlich sind diese Risikobewertungen auf die einzelne Aktivitäten eines Testfalls appliziert. Dies ermöglicht eine Rückverfolgung der Testergebnisse auf die analysierten Bedrohungsszenarien der Risikoanalyse.

Dornier Consulting hat aus diesem Ansatz der Risikobewertung sehr viel Motivation für zukünftige Projekte gezogen. Auch bei nicht sicherheitsrelevanten Projekten wird in Zukunft eine Risikobewertung in Betracht gezogen. Aber auch das Vorgehen der Übertragung der Analyseinformationen in das Testmodell hat sich als eine sehr vielversprechende Methode bewiesen.

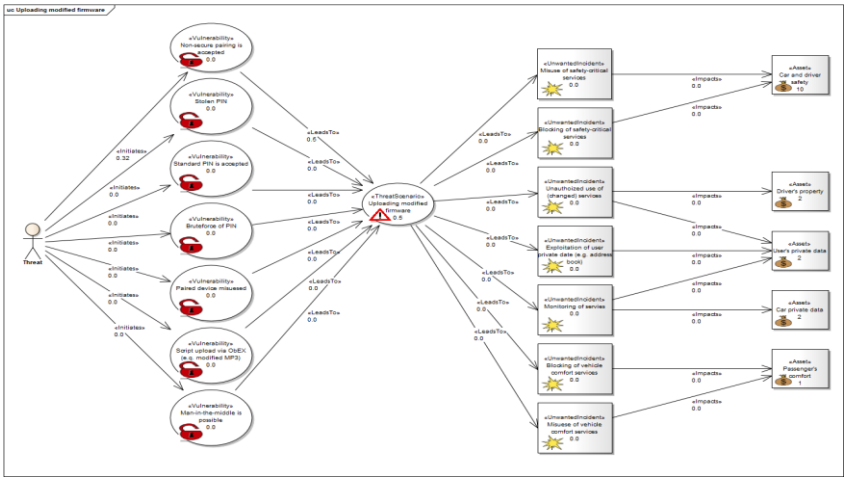

Abbildung 10: Risikoanalyse mit dem Werkzeug CORAS

7.1.2 Fuzzing

Zur Ausführung der generierten Testfälle wurde die entwickelte Fuzzing-Bibliothek von Fraunhofer FOKUS in das .NET-basierte Testwerkzeug do.ATOMS von Dornier Consulting eingebunden. Die Mithilfe der Risikoanalyse generierten Testfälle können mit der Fuzzing-Bibliothek weiter modifiziert werden, sodass das Testsystem auf Schwachstellen getestet werden kann.

Wie im spezifischen Absatz von Fraunhofer erläutert, wurde in diesem Forschungsprojekt zwei Arten von Fuzzing berücksichtigt. Zu einem das Daten-Fuzzing und zum anderen das Behavioural-Fuzzing. In der

Automotive-Fallstudie wurde die Form des Daten-Fuzzings gewählt. Die mit Hilfe der Risikoanalyse identifizierten Testschritte werden mit Hilfe der Fuzzing-Bibliothek weiter moduliert und so ein gezieltes Fehlverhalten des System-Under-Test (SUT) provoziert. Dies können zum Beispiel Integer-Grenzwerte sein, welche eventuell vom System nicht richtig abgefangen werden, oder auch Zeichen, welche vom System nicht erwartet werden.

In Abbildung 11 ist zu erkennen, dass die Fuzzing-Bibliothek aus einer Anfragedatei im XML-Format eine Liste an Ergebnisse generiert, die ebenfalls im XML-Format in das do.ATOMS-Werkzeug eingelesen werden kann. So wird eine Vielzahl an möglichen Daten generiert, die die erforderlichen Anforderungen erfüllen.

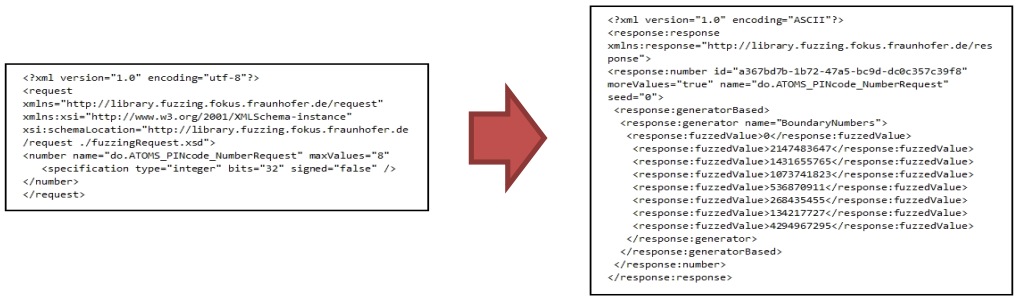

Abbildung 11: Input und Output der Fuzzing-Bibliothek

In dieser Fallstudie wurden mit Hilfe der Risikoanalyse zwei potenzielle Sicherheitslücken identifiziert:

- PIN-Übermittlung
- Discovery Namensaustausch

PIN-Übermittlung

Bei der PIN-Übermittlung wird per Systemdefinition eine vier-stellige Nummer gefordert. Diesen PIN wird im normalen Anwendungsfall in ein Mobiltelefon eingegeben und mit dem SUT abgeglichen. Eine andere Eingabe als diese vier-stellige Nummer ist in diesem Fall nicht möglich. Der Bluetooth Standard erlaubt es aber dennoch, dass auch andere alphanumerische Zeichen verwendet werden kann. Dieses Verhalten wurde mit Hilfe der Fuzzing-Bibliothek getestet.

Discovery Namensaustausch

Bevor zwei Bluetooth Geräte einen gezielten Datenaustausch starten können, müssen sich die Geräte finden. Hierzu startet eines der Geräte den Discovery-Service und sucht nach weiteren Geräten. Wenn ein Gerät gefunden wurde, wird ein Namensaustausch durchgeführt. Dabei wird gefragt, welchen Namen das andere Gerät führt, und der eigene Name wird ebenso mitgeteilt. Dieser Name wird dann in der Suchliste aufgeführt, beziehungsweise bei einer Paarungsanfrage angezeigt. Dieser Namensaustausch findet absolut unverschlüsselt statt und jegliche Zeichenfolge ist hier erlaubt. Diese Stelle eignet sich im speziellen zum Beispiel für eine SQL Injection.

Für beide potenziellen Sicherheitslücken wurden je fast 2000 modulierte Testfälle generiert, welche dann automatisch am SUT ausgeführt wurden. Dabei wurden gerade beim Namensaustausch leichte Anomalien aufgedeckt.

7.1.3 Weitere Ergebnisse

Unabhängig des deutschen Konsortiums haben sich noch IT SudParis und Montimage aus Frankreich an der von Dornier Consulting zur Verfügung gestellten Fallstudie beteiligt. Bei beiden Partnern ging es um eine Postanalyse der mitgezeichneten Bluetooth-Traces. In Abbildung 12 ist eine schematische Abbildung der Arbeit von IT SudParis zu sehen. Dabei wird mit Hilfe eines abstrakten Modells der Bluetooth-Kommunikation eine Sprachlogik erstellt. Die von Dornier Consulting aufgezeichneten Bluetooth-Traces werden mit dieser Sprachlogik analysiert. Dabei wird jeder einzelne Schritt im Bluetooth Trace mit der Sprachlogik abgeglichen und eventuelle Anomalien bewertet.

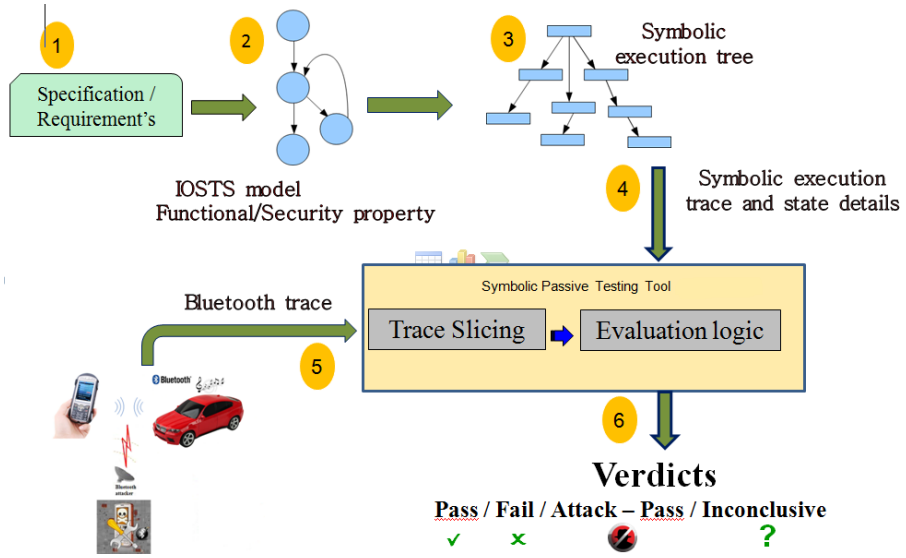

Abbildung 12: Schematische Abbildung der Arbeit von IT SudParis

Diese Methodik erlaubt eine Postanalyse der aufgezeichneten Kommunikation zwischen einem Angreifer und dem SUT. In einem nächsten Schritt soll diese Methodik bei einer laufenden Messung Anomalien während der Aufzeichnung erkennen.

7.2 VERWERTUNG DER ERGEBNISSE

7.2.1 Interne Verwertung

	Ergebnisse (laut Arbeitsplan)	**Verwertungsmöglichkeit, -aktivität nach Art und Wirkung/Nutzen**	**Zeithorizont/ geplante Realisierung**	**Erfüllungsgrad**
1	Risk-based Security Testing Method (WP4)	Grundlage für die Akquise von Folgeprojekten	Im Verlauf bzw. kurz nach Beendigung von DIAMONDS	Umgesetzt
2	Erstellung und Umsetzung eines Testfallgenerierungs-konzepts	Mit Hilfe der Erfahrungen aus dem DIAMONDS-Projekt soll ein Konzept zur Testfallgenerierung erstellt und	Im Verlauf des DIAMONDS-Projekts und in den 2 Jahren	Teilweise umgesetzt, Konzept fertig.

	Ergebnisse (laut Arbeitsplan)	**Verwertungsmöglichkeit, -aktivität nach Art und Wirkung/Nutzen**	**Zeithorizont/ geplante Realisierung**	**Erfüllungsgrad**
		umgesetzt werden.	danach	
3	Risikoanalyse	Risikoanalyse mit den DIAMONDS-Werkzeugen bei neuen Projekte, zur Analyse von Schwachstellen	Nach dem Ende von DIAMONDS	Teilweise umgesetzt

7.2.2 Externe Verwertung

	Ergebnisse (laut Arbeitsplan)	**Verwertungsmöglichkeit, -aktivität nach Art und Wirkung/Nutzen**	**Zeithorizont/ geplante Realisierung**	**Erfüllungsgrad**
1	Veröffentlichungen auf akademischen und industrienahen Konferenzen und Workshops (z.B. MBTUC 12, Automotive 2012) (WP5)	Steigerung der Wahrnehmung des Unternehmens im domainspezifischen Testumfeld	Im Verlauf bzw. kurz nach Beendigung von DIAMONDS	Umgesetzt
2	Aufsetzen und Betreuen eines Industriebeirats (WP5)	Transfer von Know-how zu Nutzergruppen, potenziellen Kunden. Verstetigung eines Diskussionsrahmens zum Thema IT-Sicherheit in Deutschland Kontaktaufnahme mit potenziellen Verwertern/ Nutzern/Akteuren für die weitere Umsetzung der Ergebnisse auch in angrenzenden Wissensgebieten (spill-over)	Im Verlauf des DIAMONDS-Projekts und in den 5 Jahren danach	Teilweise umgesetzt, der Beirat existiert und es wird über eine Verstetigung des Zusammenhangs diskutiert.
3	Ausbau des Security Testing Moduls im Testframework do.ATOMS	Durch das zusätzliche Security Testing Modul steigt das Portfolio des Testframeworks und ergänzt durch den Know-how-Ausbau die Beratungskompetenz des Unternehmens	Im Verlauf des DIAMONDS-Projekts und in den 5 Jahren danach	Umgesetzt in erster Version

	Ergebnisse (laut Arbeitsplan)	**Verwertungsmöglichkeit, -aktivität nach Art und Wirkung/Nutzen**	**Zeithorizont/ geplante Realisierung**	**Erfüllungsgrad**
4	Integration der FOKUS Fuzzing-Bibliothek in das Testfallgenerierungs-konzept des nächsten do.ATOMS Release	In einem neuen Ansatz zur Testfallgenerierung soll eine reibungslose Integration der FOKUS Fuzzing-Bibliothek erfolgen, um dem Kunden eine breite Palette zu Daten-Fuzzing-Ansätze in einem bedienungsfreundlichen Werkzeug zu ermöglichen	Im Verlauf des DIAMONDS-Projekts und in den 2 Jahren danach	Umgesetzt

Tabelle 7 Verwertung Dornier Consulting

7.3 VORAUSSICHTLICHER NUTZEN

Dornier Consulting hat aus DIAMONDS eine sehr interessante Methodik gewonnen, welche in zukünftige Kundenprojekte angewendet werden soll. Erste Pilotprojekte werden zur Zeit definiert. Insbesondere ist folgender Nutzen hervorzuheben:

- Im Umfeld des modellbasierten Testens ist die Kombination einer Analyse (Risiko, FMEA, ISO 26262, etc.) mit einem Testmodell sehr interessant, insbesondere wenn die Informationen der Analyse als Steuerungsgrößen im Generierungsprozess verwendet wird.
- Die Einbindung der Fuzzing-Bibliothek, um bereits generierte Testfälle zu modulieren. Somit lässt sich eine intelligente Variation der Testfälle generieren.

8. TESTING TECHNOLOGIES, FÖRDERKENNZEICHEN 01 IS 100 31 D

8.1 ERREICHTE ERGEBNISSE

Fuzz Testing oder auch Fuzzing ist eine etablierte Testautomatisierungsmethode um Softwarefehler auf effiziente Art und Weise aufzudecken. Sie wird insbesondere im Bereich der Sicherheitstests für Software oder Computersysteme eingesetzt. Es handelt sich dabei um eine Black-Box-Testtechnologie, bei welcher das System unter Test (SUT) mit ungültigen, unerwarteten oder auch zufälligen Daten an seinen Schnittstellen belastet wird. Das Ziel des Fuzzings ist das Aufdecken von Schwachstellen im System unter Test, indem dieses in Fehlerzustände versetzt wird. Dabei wird das System unter Test mit unerwarteten Daten, die aus modifizierten gültigen Daten generiert werden, stimuliert und gleichzeitig das Systemverhalten beobachtet.

TTCN-3 wiederum ist eine weit verbreitete Testautomatisierungstechnologie im Bereich des funktionalen Testens im Bereich der Telekommunikation und wird zunehmend auch in anderen kommunikationsintensivem Bereichen, wie Intelligent Transport Systems (ist) oder Internet of Things (IoT), eingesetzt. Im Bereich des Testens sicherheitskritischer Systeme wurde TTCN-3 bislang noch nicht eingesetzt. TTCN-3 wurde und wird vom European Telecommunications Standards Institute (ETSI) standardisiert.

Testing Technologies hat in Zusammenarbeit mit Fraunhofer FOKUS eine leichtgewichtige Erweiterung des aktuellen TTCN-3 Standards entwickelt, um Fuzzing auch in TTCN-3 zu unterstützen und somit die Anwendbarkeit von TTCN-3 auf den Bereich des Testens sicherheitskritischer Systeme zu erweitern. Fuzzing Operationen wurden dabei auf Basis des TTCN-3 Typensystems definiert und formal als spezielle Fuzz-Funktionen spezifiziert. Das Fuzzing selber, d.h. die Generierung der gefuzzten Daten, wird dabei on-the-fly während der Sendeoperationen durchgeführt. Die wiederholte Ausführung des Daten-Fuzzing, also die Generierung von verschiedenen Varianten der zu sendenden Daten, kann dabei über die klassischen Schleifenkonstrukte von TTCN-3 umgesetzt werden. Um die Wiederholbarkeit des Testverhaltens zu garantieren, nutzt der Ansatz eine Pseudozufälligkeit der Daten auf Basis eines Seeds. Da das simple zufällige Generieren von Daten nur selten zu verwertbaren Ergebnissen führt, unterstützt unser Ansatz auch das applikationsspezifische und protokollspezifische Daten-Fuzzing über die Einbindung von externen Daten-Fuzzern. Im Projekt wurde hier die Fraunhofer FOKUS Fuzzing-Bibliothek eingebunden.

Testing Technologies bietet mit der TTworkbench eine integrierte Testentwicklungs- und Testausführungsumgebung für die Technologie TTCN-3 an. TTworkbench wird in Testautomatisierungsprojekten im Bereich Telekommunikation, Automotive, Avionics, eHealth, Energy und anderen eingesetzt. TTworkbench unterstützt den gesamten Lebenszyklus von TTCN-3 basierten Testprojekten mit textuellen und grafischen Editoren, einem TTCN-3 Compiler und einem Testausführungsmanagement, zur Analyse, Debugging und Reporting für zentralisierte und verteilte Testausführungen.

Der im Projekt entwickelte Fuzz Testing-Ansatz wurde in der TTworkbench prototypisch implementiert und den Projektpartnern für die Fallstudien zur Verfügung gestellt. Innerhalb des deutschen Konsortiums hat die Fallstudie von Giesecke & Devrient diesen Ansatz genutzt und seinen praktische Einsatzfähigkeit nachgewiesen. Dabei kam auch die Integration mit der Fraunhofer FOKUS Fuzzing-Bibliothek zum Einsatz. Umgekehrt wurde die TTworkbench in das Fraunhofer FOKUS Traceability-Framework integriert, um somit eine geschlossene Werkzeugkette über den gesamten Testprozess zu erreichen.

Die TTCN-3-Erweiterung wurde dem zuständigen Standardisierungsgremium ETSI TC MTS präsentiert und dort als Work Item angenommen. Testing Technologies fungiert als Reporter für dieses Work Item. Eine Integration dieser Erweiterung der TTCN-3-Standards wird für den Juni nächsten Jahres mit der TTCN-3-Version 4.6.1 erwartet.

8.2 VERWERTUNG DER ERGEBNISSE

Testing Technologies plant die Ergebnisse des Projektes in Form von Erweiterungen der TTCN-3 Testentwicklungs- und -ausführungsplattform TTworkbench für das Testen von sicherheitskritischen Anwendungen zu vermarkten. Hier ist insbesondere der im Projekt entwickelte Ansatz zum Fuzz-Testen mit TTCN-3 zu nennen, für den bereits eine prototypische Implementierung umgesetzt wurde. Diese wird eng verzahnt mit den entsprechenden Standardisierungsaktivitäten zur Produktreife weiterentwickelt. Darüber hinaus wird das Thema Eingang in die aktuellen Schulungsunterlagen für TTCN-3 Trainings finden.

Die von uns durchgeführten bzw. geplanten Verwertungsmaßnahmen werden in der folgenden Tabelle noch einmal detailliert dargestellt:

	Ergebnisse (laut Arbeitsplan)	**Verwertungsmöglichkeit, -aktivität nach Art und Wirkung/Nutzen**	**Zeithorizont/ geplante Realisierung**	**Erfüllungsgrad**
1	Implementierung der TTCN-3 Erweiterung "Security Testing" Data Fuzzing	Erweiterung der existierenden TTCN-3 Testplattform TTworkbench, um IT-Sicherheitstest-Markt zu adressieren	Bis Mitte 2014	Teilweise umgesetzt
2	Standardisierung der TTCN-3 Erweiterung "Security Testing"	Existierender Standard als Unterstützung der Vertriebsaktivitäten der TTworkbench Erweiterung "Security Testing"	Bis Mitte 2014	Teilweise umgesetzt
3	Schulungsmaterial "Security Testing"	Erweiterung der existierenden TTCN-3 Trainingskurse um das Thema IT-Sicherheitstest	Bis Mitte 2014	Begonnen
4	Integration in die Traceability Platform	Erweiterung des Produktangebots um Komponenten der Traceability Platform-Werkzeugkette	Bis Ende 2013	Teilweise umgesetzt
5	Aufsetzen und Betreuen eines Industriebeirates	Transfer von Know-how zu Nutzergruppen, potenziellen Kunden, Verstetigung eines Diskussionsrahmens zum Thema IT-Sicherheit in Deutschland	Im Verlauf des DIAMONDS-Projekts und in den 5 Jahren danach	Teilweise umgesetzt, der Beirat existiert und es wird über eine Verstetigung der Zusammenarbeit diskutiert

Tabelle 8 Verwertung Testing Technologies IST GmbH

8.3 VORAUSSICHTLICHER NUTZEN

Durch maßgebliche Unterstützung der DIAMONDS-Projektpartner wird das Testen sicherheitskritischer Systeme im Rahmen von ETSI MTS in der Security SIG weiterverfolgt. Darüber hinaus wurde im Rahmen des deutschen Konsortiums ein Projektbeirat installiert, der sich aus interessierten Industrievertretern zusammensetzt. Mit den Mitgliedern wird aktuell über eine Verstetigung der Zusammenarbeit über das Projektende hinaus diskutiert.

9. WICHTIGSTE POSITIONEN DES ZAHLENMÄßIGEN NACHWEISES

Die Projektpartner haben im Projekt Kosten geltend gemacht, die fast ausschließlich aus Personalkosten (ca. 95%), zum geringeren Teil aus Reisekosten (ca. 4%) und sonstigen unmittelbaren Vorhabenskosten (ca. 1%) bestanden. Die Personalkosten wurden für die fachliche Arbeit in den Arbeitspaketen AP1 – AP5 sowie für die Projektleitung eingesetzt. Die Arbeiten hatten einen Umfang von ca. 200 Personenmonaten. Die Reisekosten wurden für die Teilnahme an den das Projekt begleitenden nationalen und europäischen Projekttreffen und für die Teilnahme an Fachkonferenzen im Rahmen des Projektes verwendet. Die sonstigen unmittelbaren Vorhabenskosten wurden für Hilfsmittel zur Präsentation der Projektergebnisse bei öffentlichen Projektauftritten im Rahmen der jährlich stattfindenden ITEA Symposien verwendet.

10. NOTWENDIGKEIT UND ANGEMESSENHEIT DER GELEISTETEN ARBEIT

Das Projekt DIAMONDS ist, sowohl im europäischem Rahmen wie auch im nationalen Rahmen, seiner Rolle als Initiator nachhaltiger Partnerschaften von Firmen und Organisationen zur Erforschung und Entwicklung industrietauglicher Lösungen für den Test sicherheitskritischer Systeme voll gerecht geworden. Die innovativen Forschungs- und Entwicklungsergebnisse des Projekt stärken Europa und den Standort Deutschland durch Bereitstellung von industrietauglichen Test- und Prüflösungen, die effiziente Maßnahmen zur Steigerung der Softwarequalität und somit ein probates Gegenmittel zur wachsenden Gefährdung vernetzter Systeme und Infrastrukturen durch Cyber-Angriffe darstellen. Darüber hinaus ermächtigen die in DIAMONDS erarbeiten Verfahren die beteiligten Firmen dazu, ihre Rolle als Vorreiter bei der Bereitstellung innovativer Testlösungen im Umfeld sicherheitskritischer Systeme einnehmen zu können. Ein solcher Vorstoß birgt nicht nur ein großes Potenzial, sondern er bringt auch wirtschaftliche und technische Risiken mit sich. Insbesondere für umfassende Ansätze, wie dem modell-basierten Sicherheitstest, die hohe Voraussetzung auch an die Begleitprozesse beim Kunden stellen, ist davon auszugehen, dass ein tragfähiger Markt erst mittelfristig bis langfristig entsteht. Die intensive Kooperation der Projektpartner mit Standardisierungsgremien zeigen jedoch bereits heute, wie ausgesuchte Forschungsergebnisse aus DIAMONDS auf direktem Weg in eine industrielle Nutzung überführt werden können.

Die durchgeführten Arbeiten sowie die dafür aufgewandten Ressourcen waren darüber hinaus notwendig und angemessen, um die trotz des vorhersehbaren Erfolg des Projekt die Risiken für den Einstieg in die Entwicklung neuer Testansätze auszugleichen. Die Arbeiten entsprachen der im Projektantrag detailliert dargelegten Planung und alle im Arbeitsplan formulierten Aufgaben wurden erfolgreich bearbeitet und mit industrietauglichen Ergebnissen abgeschlossen. Darüber hinaus mussten keine zusätzlichen Ressourcen zur Durchführung des Vorhabens aufgewandt werden. Die in DIAMONDS erzielten Resultate in Wissenschaft und Technik wären aufgrund der hohen Forschungs- und Entwicklungsrisiken ohne die Förderung nicht in der Form erzielt worden.

11. ZUSAMMENARBEIT MIT ANDEREN STELLEN

Der deutsche Teil des DIAMONDS-Projekts konstituierte sich aus den Partnern Fraunhofer FOKUS, Dornier Consulting, Giesecke & Devrient sowie der Testing Technologies IST GmbH. Dieser Teil war zusammen mit finnischen, französischen, norwegischen und österreichischen Partnern in das europäische ITEA-2-Projekt DIAMONDS eingebettet. Innerhalb des deutschen Projekts wurde ein Industriebeirat aufgesetzt, der die wissenschaftlichen Aktivtäten der deutschen Projektpartner beratend begleitet hat. Mit den europäischen Projektpartnern wurde auf wissenschaftlicher Ebene, in den Fallstudien sowie zur Konsolidierung und Verbreitung der Projektergebnisse zusammengearbeitet. Zudem wurden die Ergebnisse des Projekts bei Standardisierungsgremien wie beispielsweise der ETSI eingereicht, um eine langfristige und nachhaltige Nutzung der Projektergebnisse zu ermöglichen. Schlussendlich konnte eine stabile Kooperation mit andern

Forschungsprojekten etabliert werden, welche sich durch die Ausrichtung gemeinsamer Workshops und Veranstaltungen manifestieren konnte.

11.1 ZUSAMMENARBEIT MIT STANDARDISERUNGSGREMIEN

Die Partner im DIAMONDS-Projekt verfolgten das gemeinsame Ziel, die Projektergebnisse rund um das Thema IT-Sicherheitstest an Standardisierungsorganisationen heranzutragen. Da Standardisierung an sich ein langwieriger und häufig von Kompromissen geprägter Prozess ist, wurde diesbezüglich ein mehrstufiger Ansatz gewählt. In einem ersten Schritt wurden die Ergebnisse an Standardisierungsorganisationen herangetragen, mit denen die Partner bereits eine langjährige Kooperation pflegen. Hierzu gehört insbesondere das Europäische Institut für Telekommunikationsnormen (ETSI), bei dem es langjährige Mitgliedschaften von DIAMONDS-Partnern gibt und das sich insbesondere durch den relativ kurzen Zeitrahmen bis zur Etablierung von Standards auszeichnet. In einem zweiten Schritt sollten die Ergebnisse auch an andere internationale Standardisierungsgremien wie z. B. der ISO oder ITU-T weitergeleitet werden.

Im Rahmen des DIAMONDS-Projekts konzentrierten sich die Aktivitäten der Projektpartner auf die Mitarbeit in den folgenden ETSI-Arbeitsgruppen:

- ETSI MTS Interest Group (SIG) für “Security Testing”
- ETSI „Industrial Specification Group (ISG) on Information Security Indicators (ISI) “.

Die ETSI MTS Interest Group (SIG) für „Security Testing“ ist eine im Jahr 2011 gegründete Arbeitsgruppe, die sich innerhalb von ETSI MTS speziell dem Thema Security Testing widmet. Im Rahmen der Arbeitsgruppe wurden insgesamt vier Arbeitsdokumente aufgesetzt. Drei der Dokumente enthalten maßgebliche Ergebnisse aus dem DIAMONDS-Projekt.

- Das Dokument *DTS MTS-101582 Case Study Experiences* enthält Erfahrungsberichte zu Fallstudien aus dem DIAMONDS- und dem SPACIOS-Projekt. Der Erfahrungsbericht beschreibt und bewertet die Sicherheitstesttechniken und Best Practices, die im Rahmen der DIAMONDS- und SPACIOS-Fallstudien eingesetzt und gesammelt werden konnten. Das Dokument enthält Erfahrungsberichte zu Fallstudien aus den Bereichen Smart Cards, Industrial Automation, Radio Protokolle, Automotive, eHealth und Telekommunikation.
- Das Dokument *DTS MTS-101583 Security Testing Terminology and Concepts* definiert eine Terminologie und eine Ontologie, die zusammen die Grundlage für ein gemeinsames Verständnis von Sicherheitstesttechniken ermöglichen. Terminologie und Ontologie basieren auf aktuellen Standards sowie auf den Erfahrungen und Best Practices, die im Rahmen der DIAMONDS-Fallstudien gesammelt werden konnten. Terminologie und Ontologie bieten einen Leitfaden für Praktiker aus den Bereichen Sicherheitstesten und Robustheitstest über den gesamten Entwicklungs- und Produktlebenszyklus.
- Das Dokument *DTS MTS-202793 Risk-based Security Testing Methodologies* definiert Methoden zum risikobasierten IT-Sicherheitstest. Es enthält Ergebnisse aus dem DIAMONDS- und dem RASEN-Projekt.

ETSI MTS ist darüber hinaus für die Erarbeitung und Pflege des TTCN-3-Standards zuständig. Im Rahmen des DIAMONDS-Projekts wurde eine Spracherweiterung für TTCN-3 entwickelt, die eine Integration von Fuzz-Testtechniken in die Testbeschreibungssprache erlaubt. Die Spracherweiterung ist im ETSI-Dokument TR 202790 beschrieben. Eine Integration dieser Erweiterung in den TTCN-3-Standard wird für den Juni nächsten Jahres mit der TTCN-3-Version 4.6.1 erwartet.

Die ETSI „Industrial Specification Group (ISG) on Information Security Indicators (ISI)“ definiert eine Katalog von IT-Sicherheitsindikatoren, mit dem eine qualitative und quantitative Bewertung der Sicherheit von Organisationen ermöglicht wird. Sicherheitsvorfälle und bekannte Schwachstellen werden systematisch und detailliert erfasst und separat bewertet. Die Systematik dieses Katalogs erfolgt analog zu den ISO Common Criteria in Klassen, Familien und Komponenten. Für alle Komponenten werden Berechnungsformeln,

Grenzwerte und Gewichtungen (maturity) zur Erkennung von Gefahrensituationen aufgestellt. In Zusammenarbeit mit dem DIAMONDS-Projekt wurden insbesondere Methoden und Techniken für den systematischen Sicherheitstest in die Arbeitsgruppe transferiert.

Im April 2013 wurden die Arbeiten der ETSI im Rahmen des ETSI Security Workshop 2013 Vertretern der ISO SC27 vorgestellt. Es wurde beschlossen, eine formelle Verbindungen zwischen den ISO SC27 WGs (WG3, WG4) und den ETSI Arbeitsgruppen (ETSI TC MTS, ETSI ISG ISI) zu initiieren [5] und darüber einen Abgleich und einen langfristigen Austausch der Standardisierungsergebnisse zu gewährleisten.

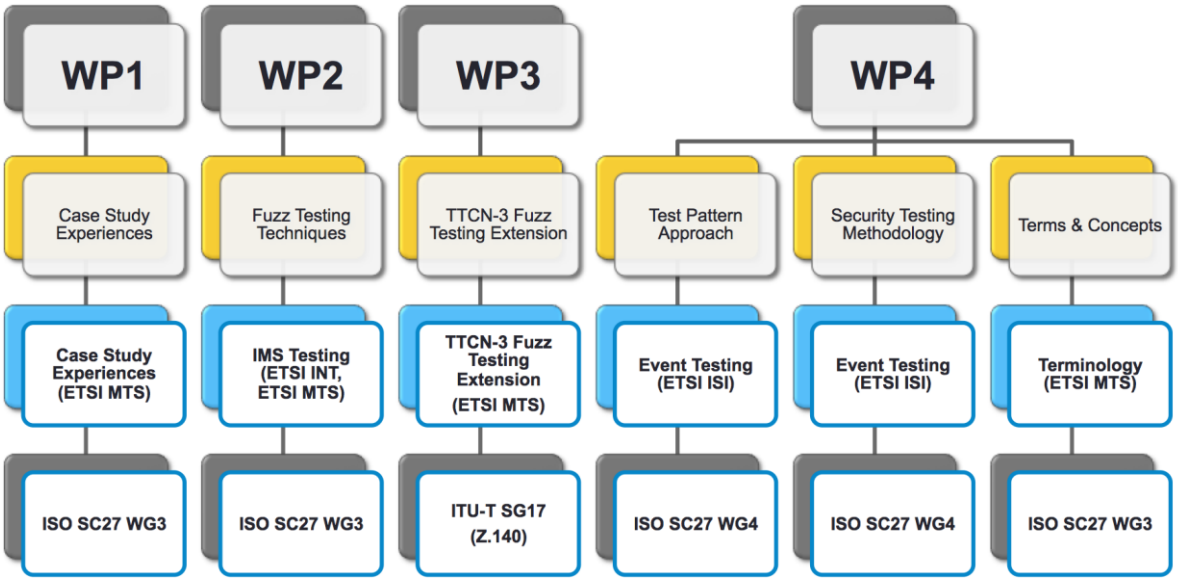

Abbildung 13: Standardisierungsbeiträge aus DIAMONDS

Abbildung 13 zeigt den Transfer der DIAMONDS-Arbeitsergebnisse in die ETSI Arbeitsgruppen sowie in die Arbeitsgruppen der ISO SC27.

11.2 ZUSAMMENARBEIT IM INDUSTRIEBEIRAT

Der DIAMONDS-Industriebeirat diente dem Projektteam als beratendes Gremium mit dem Ziel, konkrete Industrieanforderungen in das Projekt einzuspeisen, sodass diese in der Projektsteuerung berücksichtigt werden konnten. Der Industriebeirat setzte sich aus Vertretern namhafter Industriefirmen wie der Telekom, Wincor-Nixdorf, EADS, Siemens etc. zusammen. Den Mitgliedern des Beirates standen die Arbeitsergebnisse des DIAMONDS-Projektes über den gesamten Projektverlauf zur Verfügung. Auf halbjährlich stattfindenden Beiratssitzungen wurden die Projektergebnisse sowie der Projektfortschritt erörtert. Die Beiratsmitglieder hatten so die Möglichkeit, dem Projektverlauf sowohl im deutschen wie auch im europäischen DIAMONDS-Projekt zu folgen und Vorschläge zur fachlichen Projektausrichtung in das Projekt mit einzubringen. Anregungen des Beirats flossen insbesondere in die Standardisierungsaktivitäten und die Planung der Verwertungs- und Verbreitungsaktivitäten mit ein. Darüber hinaus diente der Beirat als eine von allen Partnern genutzte Plattform zur Etablierung bilateraler Kooperationen zwischen einzelnen Projektpartnern und der Industrie.

11.3 ZUSAMMENARBEIT MIT ANDEREN FORSCHUNGSPROJEKTEN

Für den Austausch auf wissenschaftlicher Ebene haben die DIAMONDS-Partner eine enge Kooperation mit anderen Forschungsprojekten gesucht. Diese Kooperation wurde insbesondere durch die Organisation

gemeinsamer Workshops realisiert. Insgesamt wurden über diesen Weg Kooperationen mit den folgenden Forschungsprojekten etabliert:

- SPaCIoS: Secure Provision and Consumption in the Internet of Services, http://www.spacios.eu/
- NESSOS: Network of Excellence on Engineering Secure Future Internet Software Services and Systems, http://www.nessos-project.eu/
- RASEN: Compositional Risk Assessment and Security Testing of Networked Systems, http://www.rasen-project.eu/
- INTER-TRUST: Interoperable Trust Assurance Infrastructure http://www.inter-trust.eu/

Die folgenden Workshops wurden im Rahmen von DIAMONDS und mit einem oder mehreren der Kooperationspartner umgesetzt:

- SecTest 2012, http://www.spacios.eu/sectest2012
- SecTest 2013, http://www.spacios.eu/sectest2013
- SASSI 2013, http://www.fokus.fraunhofer.de/en/fokus_events/motion/sassi_2013/index.html
- RISK 2013, http://www.fokus.fraunhofer.de/en/fokus_events/motion/risk_2013/index.html

11.4 ZUSAMMENARBEIT MIT ANDEREN GREMIEN

Für den industriellen Austausch und den Transfer der Ergebnisse diente auch das „Sicherheitsnetzwerk München“. In den vergangenen Jahren hat in München eine einzigartige Konzentration von Unternehmen, Startups und Forschungseinrichtungen stattgefunden, die sich auf IT-Sicherheitslösungen und -konzepte spezialisiert haben. Mittlerweile hat sich die bayerische Landeshauptstadt zu einer Art „Hot-Spot“ für Cyber-Sicherheit entwickelt. Um diese geballte Kompetenz für Kooperationen im Bereich Forschung und Entwicklung, aber auch als Branchennetzwerk der IT-Sicherheitsindustrie zu nutzen, hat G&D vor etwas mehr als zwei Jahren zusammen mit dem Fraunhofer AISEC das „Sicherheitsnetzwerk München“ gegründet.

Die Ergebnisse von DIAMONDS wurden in der Vergangenheit auf Workshops des Netzwerks präsentiert und konnten dadurch weitere Verbreitung finden.

Eine weitere nationale Initiative aus DIAMONDS heraus entstand in Zusammenhang mit der Standardisierungsarbeit zu Information Security Indicators (ETSI ISG ISI). Hierbei handelt es sich um die Gründung einer deutschen Gruppe im Rahmen der europäischen Initiative R2GS (Operational Security Management Thought and Research Club).

11.5 FORTSCHRITT BEI ANDEREN STELLEN

Während der Projektlaufzeit hat es neben den Ergebnissen der nationalen Projektpartner eine weitere Verbreitung und Fortentwicklung der modellbasierte Sicherheitsprüfung gegeben. An dieser Stelle gilt es insbesondere die Ergebnisse der Projektpartner in Frankreich, Norwegen, Finnland, Österreich und Luxemburg zu benennen, da diese Beiträge durch die deutschen Arbeiten ergänzt wurden. Darüber hinaus haben weitere Forschungsprojekte, wie beispielsweise die oben genannten Projekte SPaCIoS, NESSOS und RASEN, die Arbeiten in DIAMONDS komplementär ergänzt. Während in den Projekten SPaCIoS und NESSOS insbesondere Ansätze entwickelt wurden, die modellbasierte Prüfverfahren auf Basis von automatisierten Verifikationstechniken realisieren, entwickelt das RASEN Projekt erweiterte Verfahren zum risikobasierten Sicherheitstest. Die in RASEN entwickelten Ansätze basieren auf den Ergebnissen des DAIMONDS Projekts und erweitern diese um die Aspekte Kompositionalität sowie die Berücksichtigung rechtlicher Risiken. Durch enge Kooperation zwischen den Projekten konnte sichergestellt werden, dass die Forschungsergebnisse aufeinander abgestimmt erarbeitet werden konnten.

Neben dem von DIAMONDS initiierten Workshops SecTest, SASSI und RISK entstanden im Projektzeitraum keine weiteren uns bekannten akademischen Workshops, die explizit die DIAMONS Thematik adressieren.

Im und unmittelbar im Anschluss an den Projektzeitraum entfalteten sich Aktivitäten im Rahmen der ETSI, die das Ziel hatten, Verfahren zur modellbasierten Sicherheitsprüfung zu standardisieren. Diese Aktivitäten sind noch nicht abgeschlossen und erlauben es, die Ergebnisse aus dem DIAMONDS und dem SPaCIoS Projekt zusammenführen und nachhaltig verfügbar zu machen.

12. DELIVERABLES, VERÖFFENTLICHUNGEN UND WORKSHOPS

12.1 ÖFFENTLICHE DELIVERABLES

[D1.WP2] Jean-Luc Richier (ed.); Security Testing Techniques; D1.WP2; DIAMONDS Consortium, 2011-05-31

[D1.WP3] Ilkka Uusitalo (ed.); Review of Security Testing Tools; D1.WP3; DIAMONDS Consortium, 2011-06-27

[D2.WP2] Wissam Mallouli (ed.); Concepts for Model-Based Security Testing; D2.WP2; DIAMONDS Consortium, 2011-09-26

[D2.WP4] Fredrik Seehusen (ed.); Methodologies for risk- and model-based security testing; D2.WP4; DIAMONDS Consortium, 2011-12-12

[D3.WP2] Franz Wotawa (ed.); Initial Model-Based Security Testing Methods; D3.WP2; DIAMONDS Consortium, 2012-07-06

[D3.WP3] Matti Mantere (ed.); Initial Design of Security Testing Tools; D3.WP3; DIAMONDS Consortium, 2012-06-29

[D3.WP4a] Alain-Georges Vouffo Feudjio (ed.); Initial Security Test Patterns Catalogue; D3.WP4.T1; DIAMONDS Consortium, 2012-06-30

[D3.WP4b] Fredrik Seehusen (ed.); Initial methodologies for model-based security testing and risk-based security testing; D3.WP4.T2_T3; DIAMONDS Consortium, 2012-07-02

[D5.WP2] Stephane Maag (ed.); Final Security Testing Techniques; D5.WP2.v10.FINAL; DIAMONDS Consortium, 2013-05-23

[D5.WP3] Matti Mantere, (ed.); Final Security Testing Tools; D5.WP3.v10.FINAL; DIAMONDS Consortium, 2013-05-22

[D5.WP4] Fredrik Seehusen (ed.); DIAMONDS Security Testing Methodology; D5.WP4.T1-T3v1.0; DIAMONDS Consortium, 2013-05-16

[D5.WP5] Teemu Tokola (ed.); Security Testing Training Description; D5.WP5.T3.FINAL; DIAMONDS Consortium, 2013-05-21

12.2 VERÖFFENTLICHUNGEN

[DIA-1] Organization, chairing and preparation of proceedings for the DIAMONDS workshop STV at October 24th in Paris: http://www.fokus.fraunhofer.de/en/fokus_events/motion/stv_2012/index.html

[DIA-2] DIAMONDS Consortium, Methods for Testing and Specification (MTS); Security Testing; Security testing terminology, concepts and lifecycle, http://docbox.etsi.org/MTS/MTS/07-Drafts/00101583_SecTest_Terms/MTS-101583%20SecTest_Termsv002.docx

[DIA-3] DIAMONDS Consortium, Information Security Indicators (ISI); Event Testing; Part 5: Event Testing; Guidelines for event testinghttp://docbox.etsi.org/ISG/ISI/05-CONTRIBUTIONS/2013/ISI(13)M09007_Information_Security_Indicators__ISI___Part_5___A_security_e.zip

[DIA-4] Wolfgang Kremer, Andreas Schulze, Jürgen Großmann; How can we test security? Model-based Security Testing for the automotive industry; conference paper and presentation; 27. VDI/VW-Gemeinschaftstagung Automotive Security (01TA102011) 2011; Berlin, Germany; 11. Oct 11

[DIA-5] Ina Schieferdecker, Axel Rennoch, Jürgen Großmann; Security Testing Approaches in Industry and Standardization; conference paper; 23. International Conference on Software & Systems Engineering and their Applications (ICSSEA'11) 2011; Paris, France; 29. Nov 11; published in: ISSN-0295-6322

[DIA-6] Ina Schieferdecker, Axel Rennoch, Jürgen Großmann; DIAMONDS do IT with MODELS: Innovative Security Testing Approaches; article; ERCIM News #88 2012, ERCIM EEIG; Sophia-Antipolis, France; Jan 12; published in: ISSN 0926-4981

[DIA-7] Stephan Pietsch, Bogdan Stanca-Kaposta, Jürgen Großmann, Martin Schneider, Dirk Tepelmann, Jacob Wieland; Data Fuzzing with TTCN-3; Presentation; ETSI TTCN-3 User Conference & Model Based Testing Workshop 2012; Bangalore, India; 11. Jun 12

[DIA-8] Felix Jakob, Andreas Schulze; Risk-based testing of Bluetooth functionality; Presentation; MBT User Conference 2012; Tallinn, Estonia; 25. Sep 12

[DIA-9] Ina Schieferdecker, Jürgen Großmann, Martin Schneider Model-Based Security Testing; Keynote & Conference Paper; MBT Workshop at ETAPS 2012; Tallinn, Estonia; 25. Mar 12; published in: Conf. Proceeding EPTCS; Mar 12

[DIA-10] Ina Schieferdecker, Jürgen Großmann, Martin Schneider; Model-Based Fuzzing for Security Testing; Keynote & Extended Abstract; SECTEST Workshop at ICST 2012; Montreal, Canada; 21. Apr 12; published in: Conf. Proceeding IEEE; Apr 12

[DIA-11] Ina Schieferdecker; Test Automation; Keynote; AST Workshop at COMPSAC 2012; Izmir, Turkey; 16. Jul 12

[DIA-12] Ina Schieferdecker; DIAMONDS Poster; conference poster; ITEA Symposium 2010; Ghent, Belgium; 26. Oct 10

[DIA-13] Ina Schieferdecker; DIAMONDS Poster; conference poster; 4. IEEE International Conference on Software Testing, Verification and Validation 2011; Berlin, Germany; 21. Mar 11

[DIA-14] Ina Schieferdecker; Model Based Security Testing: Selected Considerations (Keynote); keynote; Sectest 2011, Workshop on the 4th IEEE International Conference on Software Testing, Verification and Validation 2011; Berlin, Germany; 21. Mar 11

[DIA-15] Jan-Bernhard Demeer, Axel Rennoch; The ETSI TVRA Security Measurement Methodology by Means of TTCN-3 Notation; conference paper; TTCN-3 User Conference 2011; Bled, Slovenia; 3. Jun 11

[DIA-16] Ina Schieferdecker, Axel Rennoch, Jürgen Großmann; Security Testing Approaches in Industry and Standardization; conference paper; 23. International Conference on Software & Systems Engineering and their Applications (ICSSEA'11) 2011; Paris, France; 29. Nov 11; published in: ISSN-0295-6322

[DIA-17] Ina Schieferdecker, Axel Rennoch, Jürgen Großmann; DIAMONDS do IT with MODELS: Innovative Security Testing Approaches; article; ERCIM News #88 2012; ERCIM EEIG; Sophia-Antipolis, France; Jan 12; published in: ISSN 0926-4981

[DIA-18] Stephan Pietsch, Bogdan Stanca-Kaposta, Jürgen Großmann, Martin Schneider, Dirk Tepelmann, Jacob Wieland; Data Fuzzing with TTCN-3; Presentation; ETSI TTCN-3 User Conference & Model Based Testing Workshop 2012; Bangalore, India; 11. Jun 12

[DIA-19] Felix Jakob, Andreas Schulze; Risk-based testing of Bluetooth functionality; Presentation; MBT User Conference 2012; Tallinn, Estonia; 25. Sep 12

[DIA-20] Martin Schneider, Nikolay Tcholtchev, Jürgen Großmann, Andrej Pietschker, Alain-Georges Vouffo Feudjio, Ina Schieferdecker; Model based Behavioural Fuzzing for the Banking Domain; Presentation; MBT User Conference 2012; Tallinn, Estonia; 25. Sep 12

[DIA-21] Martin Schneider, Jürgen Großmann, Nikolay Tcholtchev, Ina Schieferdecker, Andrej Pietschker; Behavioral Fuzzing Operators for UML Sequence Diagrams; Workshop Paper; 7. System Analysis and Modeling Workshop (SAM 2012) 2012; Innsbruck, Austria; 1. Oct 12; published in: LNCS vol. 7744; Springer; 25. Oct 12

[DIA-22] Martin Schneider; Model based Behavioural Fuzzing; Workshop Paper; System Testing and Validation (STV 2012) 2012; Paris, France; 24. Oct 12; published in: Workshop Proceedings; Fraunhofer; 25. Oct 12

[DIA-23] Michael Berger; A Traceability Managing Tool as an Integration Platform for Risk-based Security Testing; Workshop Paper; System Testing and Validation (STV 2012) 2012; Paris , France; 24. Oct 12; published in: Workshop Proceedings; Fraunhofer; 25. Oct 12

[DIA-24] Johannes Viehmann; Reusing Risk Analysis Results - An Extension for the CORAS Risk Analysis Method; Conference Paper; PASSAT 2012; Amsterdam, Netherlands; 3. Sep 12; published in: Conf. Proceeding; IEEE; Sep 12

[DIA-25] Felix Jakob, Andreas Schulze, Wolfgang Kremer, Jürgen Großmann, Nadja Menz, Martin Schneider, Alain-Georges Vouffo Feudjio; Risk-based testing of Bluetooth functionality in an automotive environment; Conference paper and presentation; Automotive 2012; Karlsruhe, Germany; 14. Nov 12

[DIA-26] Felix Jakob, Markus Rath, Andreas Schulze; A case study report on security testing of Bluetooth functionality in an automotive environment; Conference paper and presentation; Embedded Security in Cars Conference (ESCAR) 2012; Berlin, Germany; 28. Nov 12

[DIA-27] Felix Jakob, Markus Rath, Andreas Schulze; A case study report on security testing of Bluetooth functionality in an automotive environment; Conference paper and presentation; Embedded Security in Cars Conference (ESCAR) 2013; Berlin, Germany; 28. Nov 13

[DIA-28] Martin Schneider, Jürgen Großmann, Ina Schieferdecker, Andrej Pietschker; Online Model-Based Behavioral Fuzzing; Workshop paper; The 4. International Workshop on Security Testing SECTEST2013 (in association with ICST 2013) 2013; Luxembourg; Mar 13; published in: Conf. Proceedings; IEEE

[DIA-29] Ina Schieferdecker; Case Study Experiences from the DIAMONDS Project; Presentation; 8. ETSI Security Conference 2013; Sophia Antipolis, France; 16. Jan 13

[DIA-30] Martin Schneider, Nikolay Tcholtchev, Jürgen Großmann, Ina Schieferdecker; Model-Based Fuzzing Testing for Security; Presentation; ATAMI Workshop 2012; Berlin, Germany; 27. Sep 12

[DIA-31] Ina Schieferdecker; Recent Advances in Test Automation; Keynote; AST Workshop at COMPSAC 2013; Izmir, Turkey; 1. Jul 12

[DIA-32] Ina Schieferdecker; Dependability of Connected Systems; Keynote; Connected Products, Management Circle 2012; Stuttgart, Germany; 19. Nov 13

[DIA-33] Ina Schieferdecker; DIAMONDS – Development and Industrial Application of Multi-Domain Security Testing Technologies; Presentation; ITEA ARTEMIS Co-Summit 2012; Paris, France; 30. Oct 13

[DIA-34] Martin Schneider, Jürgen Großmann, Ina Schieferdecker; Model-Based Security Testing; Presentation; MBT Workshop at ETAPS 2012; Tallinn, Estonia

[DIA-35] Axel Rennoch, Ina Schieferdecker, Jürgen Großmann; Security testing achievements and benefits by European research; Talk; InfoTech 2013; Dalian, China; Jun 13

[DIA-36] Marc-Florian Wendland, Martin Schneider, Øystein Haugen; Evolution of the UML Interactions Metamodel; Conference Paper; Models 2013; Sep 13; published in: Conf. Proceeding; to appear LNCS vol. 8107, pp. 405-421

[DIA-37] Axel Rennoch; Model-based security testing and its applications to industrial case studies Security risk analysis and smart fuzzing in practice; Keynote; ASQT 2013; Graz, Austria; Sep 13

[DIA-38] Jürgen Großmann; Security Testing Improvement Profile (STIP) Presentation; SASSI 2013; Berlin, Germany; Sep 13; published in: Conf. Proceeding; to appear in Dec 2013

[DIA-39] Ina Schieferdecker; Model- based testing with the UML testing profile; Tutorial; Softec 2013; Malaysia; Sep 13

[DIA-40] Ina Schieferdecker; Model-based security testing and its application to industrial case studies; Keynote; Softec 2013; Malaysia; Sep 13

[DIA-41] Ina Schieferdecker, Axel Rennoch; Model-Based Security Testing – Results from Industrial Case Studies; Presentation; German Testing Days 2013; Munich; Germany; Nov 13; published in: to appear in Nov 13

[DIA-42] Axel Rennoch, Ina Schieferdecker, Jürgen Großmann: Security Testing Approaches - for Research, Industry and Standardization: International Standard Conference on Trustworthy Computing and Services (ISCTCS), 29.-30. November 2013, Beijing, China

12.3 WORKSHOPS

[W-1] Organisation des SecTest Workshops 2011, http://www.avantssar.eu/sectest2011/

[W-2] Organisation des SecTest Workshops 2012, http://www.spacios.eu/sectest2012

[W-3] Organisation des SecTest Workshops 2013, http://www.spacios.eu/sectest2013

[W-4] Organisation und Leitung des DIAMONDS Workshop STV auf der ICSSEA12 in Paris: http://www.fokus.fraunhofer.de/en/fokus_events/motion/stv_2012/index.html

[W-5] Organisation und Leitung des DIAMONDS-Tracks bei der MBT UC 2012 in Tallin: http://www.elvior.com/model-based-testing-uc-2012

[W-6] Organisation und Leitung des DIAMONDS Tutorials auf der ICST 2013 in Luxembourg, http://www.icst.lu/site/icst2013/program/tutorial

[W-7] Organisation und Leitung des SASSI Workshops SASSI 2013, http://www.fokus.fraunhofer.de/en/fokus_events/motion/sassi_2013/index.html

[W-8] Organisation und Leitung des RISK Workshops RISK 2013, http://www.fokus.fraunhofer.de/en/fokus_events/motion/risk_2013/index.html

13. EXTERNE REFERENZEN

[1] A. Abou El Kalam, R. E. Baida, P. Balbiani, S. Benferhat, F. Cuppens, Y. Deswarte, A. Mi`ege, C. Saurel, G. Trouessin; Organization Based Access Control; 4. IEEE International Workshop on Policies for Distributed Systems and Networks (Policy'03), Jun 03.

[2] Becker, S., Abdelnur, H., State, R., Engel, T.; An Autonomic Testing Framework for IPv6 Configuration Protocols; In: Mechanisms for Autonomous Management of Networks and Services. Lecture Notes in Computer Science; Stiller, B., De Turck, F. (Eds.); Springer Berlin / Heidelberg. 2010

[3] Bruno Legeard, Marteen Rits; Model-based testing of SAP systems; Software and Systems Quality Conferences; Zurich, Oct 07.

[4] Common Criteria for Information Technology security; Part I: Introduction and General Model; Version 3.1. CCMB-2006-09-001; http://www.commoncriteriaportal.org/

[5] CORAS Werkzeug und Methode: www.sourceforge.com/coras

[6] CREMA at guersoy.net: http://www.guersoy.net/knowledge/crema

[7] D. Basin, J.Doser, T. Lodderstedt; Model driven security: From UML models to access control infrastructures; ACT Transactions on Software Engineering Methodologies; 15(1):39-91; 2006.

[8] D. Senn, D. A. Basin, G. Caronni; Firewall conformance testing; In: TestCom, pages 226–241; 2005.

[9] D.Avresky, J.Arlat, J.-C.Laprie, Y.Crouzet; Fault injection for the formal testing of fault tolerance; In. Proc. of Twenty-Second International Symposium on Fault-Tolerant Computing (FTCS-22), Jul 92, pp345-354; Boston, MA, USA.

[10] E. Jarvi; Security Testing and Risk Management, testing experience; Mar 09

[11] Eclipse Papyrus UML: http://www.eclipse.org/papyrus/

[12] Erik van Veenendaal; Test Maturity Model integration: http://www.tmmi.org/pdf/TMMi.Framework.pdf

[13] ETSI RES/MTS-203150 Methods for Testing and Specification (MTS); The Testing and Test Control Notation version 3; TTCN-3 language extensions: Support for Security Testing

[14] Frost&Sullivan; World Security Testing Equipment Markets; 2007

[15] Fuzzing-Bibliothek Fuzzino
http://www.fokus.fraunhofer.de/de/motion/ueber_motion/technologien/fuzzing

[16] G. Wimmel, J. Jürjens; Specification-based Test Generation for Security-Critical Systems Using Mutations; In: proceedings of the International Conference on Formal Engineering Methods (ICFEM'02), Springer; 2002

[17] Gartner RAS Core Research Note G00164100; Joseph Feiman, Neil MacDonald; Magic Quadrant for Static Application Security Testing; 6 Feb 09

[18] Hsu, Y., Shu, G., Lee, D.; A model-based approach to security flaw detection of network protocol implementations; In: IEEE International Conference on Network Protocols 2008; ICNP 2008. pp.114-123; 19-22 Oct 08

[19] Information Systems Security Assessment Framework (ISSAF) Penetration Testing Framework; Available from http://www.oissg.org/issaf.

[20] J. A. Arnedo, A. Cavalli, M. Nunez; Fast Testing of Critical Properties through Passive Testing; Lecture Notes on Computer Science, vol 2644/2003, Pages 295-310; Springer; 2003

[21] J. Garcıa-Alfaro, F. Cuppens, N. Cuppens-Boulahia; Analysis of policy anomalies on distributed network security setups; In: ESORICS, Pages 496–511; 2006

[22] J. Garcıa-Alfaro, F. Cuppens, N. Cuppens-Boulahia; Towards filtering and alerting rule rewriting on single-component policies; In: SAFECOMP, Pages 182–194; 2006.

[23] J. Jürjens, G. Wimmel; Specification-Based Testing of Firewalls; In: proceedings of the 4th International Andrei Ershov Memorial Conference on Perspectives of System Informatics (PSI'01), pages 308-316; Springer; 2001

[24] J. Jürjens; Model-based security Testing Using UMLSec; Electronic Notes in Theoretical Computer Science 220(1): 93-104; 2008

[25] J. Jürjens; Secure systems development with UML; Springer; 2005

[26] J. Lobo, R. Bhatia, S. A. Naqvi; A policy description language; In: AAAI/IAAI, pages 291–298; 1999

[27] K. Koscher, A. Czeskis, F. Roesner, S. Patel, T. Kohno, S. Checkoway, D. McCoy, B. Kantor, D. Anderson. H. Shacham, S. Savage: Experimental Security Analysis of a Modern Automobile. In: *IEEE Symposium on Security and Privacy (SP'10), pages 447-462; IEEE Computer Society*

[28] L. Wang, E. Wong, D. Xu; A Threat Model Driven Approach for Security Testing; In: proceedings of the 3rd International Workshop on Software Engineering for Secure Systems (SESS'07); pages 10-16; IEEE Computer Society; 2007

[29] M. Blackburn, R. Busser, A. Nauman; Model-based Approach to Security Test Automation; In: proceedings of the 13th International Symposium on Software Reliability Engineering (ISSRE'02); 1999

[30] M. Broy, B. Jonsson, J.-P. Katoen, M. Leucker, A. Pretschner; Model-Based Testing of Reactive Systems; LNCS 3472; Springer; 2005

[31] N. Damianou, A. Bandara, M. Sloman, E. Lupu; Handbook of Network and System Administration, chapter: A Survey of Policy Specification Approaches; Elsevier; 2007

[32] N. Damianou, N. Dulay, E. Lupu, M. Sloman; The ponder policy specification language; In POLICY '01: Proceedings of the International Workshop on Policies for Distributed Systems and Networks; pages 18–38; London, UK; 2001; Springer

[33] NIST Special Publication 800-115: Technical Guide to Information Security Testing and Assessment

[34] Oehlert, P.; Violating assumptions with fuzzing; In: IEEE Security & Privacy, vol. 3, no. 2; pp. 58-62; Mar-Apr 05

[35] P. Baker, Z. R. Dai, J. Grabowski, Ø. Haugen, I. Schieferdecker, C. Williams; Model-driven testing. Using the UML Testing Profile; Springer; 2008

[36] ProR: http://www.eclipse.org/rmf/pror/

[37] R.M. Hierons, J.P. Bowen, Mark Harman (Eds.); Formal Methods and Testing, LNCS 4949; Springer; 2008

[38] Software Engineering Institute: Research outline, http://www.sei.cmu.edu/tsp/research/index.html, Last visited Apr. 2009

[39] T. Kitagawa, M. Hanaoka, K. Kono: AspFuzz: A state-aware protocol fuzzer based on application-layer protocols; In: IEEE Symposium on Computers and Communications (ISCC'10), pp. 202-208, 22-25 June 2010

[40] T. Koomen, M. Pool; Test process improvement – A practical step-by-step guide to structured testing; Addison Wesley; 1999

[41] T. Mouelhi, F. Fleurey, B. Baudry, Y. L. Traon; A Model-Based Framework for Security Policy Specification, Deployment and Testing; In: proceedings of the 11th International Conference on Model Driven Engineering Languages and Systems; pages 537-552; Springer; 2008

[42] The Institute for Security and Open Methodologies; "The Open Source Security Testing Methodology Manual"

[43] TTworkbench: http://www.testingtech.com/products/ttworkbench.php

[44] V. Darmaillacq, J.-C. Fernandez, R. Groz, L. Mounier, J.-L. Richier; Test generation for network security rules; In: TestCom; pages 341–356; 2006

[45] VERDE Forschungsprojekt http://www.itea-verde.org/

[46] W. Mallouli, F. Bessayah, A. R. Cavalli, Azzedine Benameur; Security Rules Specification and Analysis Based on Passive Testing; The IEEE Global Communications Conference (GLOBECOM 2008); New Orleans, USA; 30 Nov – 4 Dec 08

[47] URL http://www.bicc-net.de/kontakt/unternehmen/sicherheitsnetzwerk-muenchen/ ; retrieved: 1 Dec 03

Berichtsblatt

1. ISBN oder ISSN	2. Berichtsart (Schlussbericht oder Veröffentlichung) Schlussbericht
3. Titel DIAMONDS Schlussbericht	
4. Autor(en) [Name(n), Vorname(n)] Schieferdecker, Ina; Großmann, Jürgen; Schneider, Martin; Viehmann, Johannes *(FhG FOKUS)* Pietsch, Stephan *(Testing Technologies IST GmbH)* Pietschker, Andrej *(Giesecke & Devrient)* Jakob, Felix; Schulze, Andreas *(Dornier Consulting)*	5. Abschlussdatum des Vorhabens 30.06.2013 6. Veröffentlichungsdatum 31.12.2013 7. Form der Publikation Report
8. Durchführende Institution(en) (Name, Adresse) Fraunhofer FOKUS, Kaiserin-Augusta-Allee 31, 10589 Berlin Testing Technologies IST GmbH, Michaelkirchstr. 17/18 Giesecke & Devrient, Prinzregentenstr. 159, 81607 München Dornier Consulting, Kolumbusstr. 27, 71063 Sindelfingen	9. Ber. Nr. Durchführende Institution 10. Förderkennzeichen 01 IS 100 31A, 01 IS 100 31B, 01 IS 100 31C, 01 IS 100 31D 11. Seitenzahl 67
12. Fördernde Institution (Name, Adresse) Bundesministerium für Bildung und Forschung (BMBF) 53170 Bonn	13. Literaturangaben 101 14. Tabellen 8 15. Abbildungen 13
16. Zusätzliche Angaben	
17. Vorgelegt bei (Titel, Ort, Datum)	
18. Kurzfassung Dieser Schlussbericht dokumentiert die Ergebnisse des deutschen DIAMONDS Projekt, das bestehend aus den Partnern Fraunhofer FOKUS, Giesecke & Devrient, Dornier Consulting und Testing Technologies Fallstudien aus den Bereichen des Bankwesens und Automobilindustrie bearbeitet haben. Das deutsche DIAMONDS Projekt ist in das europäische DIAMONDS Projekt integriert gewesen. Zu den innovativen Ergebnissen des Projekts zählen Techniken für den modellbasierten Robustheitstest durch Smart Behavioural Fuzzing, die Dokumentation und Wiederverwendung von Know-how in Form von Security Test Pattern, eine Methode für den Risiko-basierten IT-Sicherheitstest und eine Open Source Plattform zur Integration von IT-Sicherheitstestwerkzeugen. Die Überführung von ausgesuchten Projektergebnissen in Standardisierungsaktivitäten bei der ETSI (MTS-SIG, ISG-ISI) sorgen für eine nachhaltige Konsolidierung und Verfügbarkeit der Projektergebnisse.	
19. Schlagwörter Modellbasierter IT-Sicherheitstest, IT-Sicherheitstest, Risikobasiertes Testen, IT-Sicherheit, Security	
20. Verlag	21. Preis

Document Control Sheet

1. ISBN or ISSN	2. type of document (e.g. report, publication) final report
3. title DIAMONDS final report	
4. author(s) (family name, first name(s)) Schieferdecker, Ina; Großmann, Jürgen; Schneider, Martin; Viehmann, Johannes *(FhG FOKUS)* Pietsch, Stephan *(Testing Technologies IST GmbH)* Pietschker, Andrej *(Giesecke & Devrient)* Jakob, Felix; Schulze, Andreas *(Dornier Consulting)*	5. end of project June 30, 2013 6. publication date December 31, 2013 7. form of publication report
8. performing organization(s) (name, address) Fraunhofer FOKUS, Kaiserin-Augusta-Allee 31, 10589 Berlin Testing Technologies IST GmbH, Michaelkirchstr. 17/18 Giesecke & Devrient, Prinzregentenstr. 159, 81607 München Dornier Consulting, Kolumbusstr. 27, 71063 Sindelfingen	9. originator's report no. 10. reference no. 01 IS 100 31A, 01 IS 100 31B, 01 IS 100 31C, 01 IS 100 31D 11. no. of pages 67
12. sponsoring agency (name, address) Bundesministerium für Bildung und Forschung (BMBF) 53170 Bonn	13. no. of references 101 14. no. of tables 8 15. no. of figures 13
16. supplementary notes	
17. presented at (title, place, date)	
18. abstract This final report documents the results of the German DIAMONDS project. The German DIAMONDS project has been integrated into the European DIAMONDS project. The German DIAMONDS project has been carried out by Fraunhofer FOKUS, Giesecke & Devrient, Dornier Consulting and Testing Technologies. The partners have worked on case studies in the fields of banking and the automotive industry. Among others, the innovative results of the project include new techniques for model-based robustness testing, a new method for smart behavioural fuzz testing, the documentation and reuse of know-how in the form of Security Test Pattern, a methodology risk-based security testing and an open source platform for security test tool integration. The transfer of selected project results in standardization activities in ETSI (MTS-SIG, ISG-ISI) provide for a sustainable consolidation and availability of project results.	
19. keywords model based security testing, security testing, risk based security testing, security	
20. publisher	21. price