technische universität
dortmund

Seminararbeit

Compliance Analyse mit dem Riskfinder Werkzeug

Christian Riest
31. Januar 2014

Gutachter: Prof. Dr. Jan Jürjens
Dipl.-Inf. Jens Bürger

Prof. Dr. Jan Jürjens Lehrstuhl 14 Software Engineering
Fakultät Informatik
Technische Universität Dortmund
Otto-Hahn-Straße 14
44227 Dortmund
http://www-jj.cs.uni-dortmund.de/secse

Christian Riest
christian.riest@tu-dortmund.de
Matrikelnummer: 147446
Studiengang: Bachelor Informatik

Proseminar Werkzeugunterstützung für sichere Software
Thema: Compliance Analyse mit dem Riskfinder Werkzeug

Eingereicht: 31. Januar 2014

Betreuer: Jens Bürger

Prof. Dr. Jan Jürjens Lehrstuhl 14 Software Engineering
Fakultät Informatik
Technische Universität Dortmund
Otto-Hahn-Straße 14
44227 Dortmund

Ehrenwörtliche Erklärung

Ich erkläre hiermit ehrenwörtlich, dass ich die vorliegende Arbeit selbstständig angefertigt habe; die aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht.

Die Arbeit wurde bisher keiner anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

Dortmund, den 31. Januar 2014

Christian Riest

Inhaltsverzeichnis

1 Einleitung

1.1 Thema dieser Arbeit

Die Ausarbeitung „Compliance Analyse mit dem Riskfinder Werkzeug“ befasst sich im Kontext des Proseminars „Werkzeugunterstützung für sichere Software“ mit der automatisierten Analyse von Softwaremodellen. Zentraler Aspekt dabei ist, dass an die Software gestellte Anforderungen schon während der Entwicklungsphase eingehalten werden sollen. Dadurch wird *Compliance*, also das Einhalten von Regeln und Gesetzen seitens der Software, hergestellt.
Dazu wird das im Rahmen der Diplomarbeit „Werkzeuggestützte Modell-basierte Sicherheitsanalyse für IT-Sicherheitsmanagement“ [Pes10] entstandene Werkzeug *Riskfinder* vorgestellt, welches die Analyse eines Softwaremodells auf der Grundlage eines *Gefahren- und Maßnahmenkatalogs* ermöglicht. Somit können eventuelle Gefahren- und Risikobereiche schon während der Modellierungsphase erkannt und vom Entwickler mit Hilfe der empfohlenen Maßnahmen behoben werden, so dass die Software die sicherheitsspezifischen Anforderungen erfüllt. Dadurch wird dann *Compliance* erreicht.

1.2 Einführung in die Problematik

Ein grundliegendes Problem in der heutigen Softwareentwicklung ist die Erfüllung von festgelegten Anforderungen durch die zu entwickelnde Software. Diese können je nach Art bzw. Einsatzort der Software in Form von Bestimmungen, Regeln und Gesetzen auftreten. Damit sichergestellt ist, dass die Software die an sie gestellten Anforderungen erfüllt, ist es deshalb sinnvoll, dies schon während der Entwicklungsphase automatisiert zu überprüfen, da eine nachträgliche Modifikation sehr aufwendig bzw. unmöglich sein kann.
Dabei erweist sich als Problem, dass die Software in der Entwicklungsphase üblicherweise durch eine Modellierungssprache, zum Beispiel UML, spezifiziert und beschrieben wird, die Anforderungen jedoch nicht notwendigerweise in dieser Form vorliegen. So werden Gesetze üblicherweise als Fließtext formuliert und dabei ein anderes Vokabular verwendet, als in der Softwareentwicklung. Daraus ergibt sich gerade für die automatisierte Überprüfung die Schwierigkeit, dass in einem Softwaremodell nicht erkennbar ist, ob eine Anforderung erfüllt ist oder nicht. Gründe dafür sind zum Beispiel, dass die textuelle Beschreibung der Anforderung nicht mit der in der Modellkomponente übereinstimmt oder das die Anforderung durch die Struktur des Modells umgesetzt wird.

Deshalb wird im Rahmen dieser Ausarbeitung das Riskfinder Werkzeug vorgestellt, welches den Entwickler dabei unterstützt, ein in UML spezifiziertes Softwaremodell auf der Grundlage textuell formulierter Anforderungen, die in Form eines Gefahren- und Maßnahmenkatalogs vorliegen, zu analysieren und somit die Komponenten zu ermitteln, die die Anforderungen nicht erfüllen.
Im Mittelpunkt steht dabei die Betrachtung des dem Riskfinder zugrundeliegenden Analysealgorithmus, durch den es möglich wird, Softwaremodelle im Kontext textuell formulierter Anforderungen zu analysieren.
Somit kann schon während der Entwicklungsphase ermittelt werden, ob die Software die an sie gestellten Anforderungen erfüllt und compliant ist.

1.3 Aufbau

Diese Ausarbeitung unterteilt sich neben dem Kapitel Einleitung, in welchem in die Thematik eingeführt wird, in drei weitere Kapitel.
In „Kapitel 2 Grundlagen“ wird der für das Verständnis wichtige Begriff „Compliance“ betrachtet und näher erläutert. Besonders wird der Einsatz im Bereich Software hervorgehoben und am Beispiel des *IT-Grundschutz-Katalogs* verdeutlicht.
Außerdem werden die Projekte *Wortschatz* und *SecRec* vorgestellt, da im Riskfinder Tools, die im Rahmen dieser Projekte entstanden sind, verwendet werden. Außerdem wird der Begriff *Stopword* eingeführt und an Beispielen verdeutlicht.
Riskfinder wird dann in „Kapitel 3 Riskfinder“ vorgestellt, wobei neben der Beschreibung des Analysealgorithmus mögliche Anpassungsmöglichkeiten und Optionen, die Riskfinder anbietet, behandelt werden. Abschließend wird ein Beispiel in Form eines UML-Aktivitätsdiagramms betrachtet. Mögliche Erweiterungen und weitere Einsatzgebiete sowie eine kritische Stellungnahme zu Riskfinder im Bezug auf die Entwicklung sicherer Software und Compliance werden in „Kapitel 4 Kritische Auseinandersetzung und Ausblick“ behandelt und ein abschließendes Fazit gezogen.

2 Grundlagen

2.1 Compliance

2.1.1 Begriffsdefinition Compliance

Um die Problematik von Compliance in der Softwareentwicklung besser zu verstehen, ist es sinnvoll, sich näher mit dem Begriff auseinanderzusetzen, da er in der Literatur nicht eindeutig definiert wird.
So ist die Definition im Oxford Dictionary
„compliance: noun ∼ the practice of obeying rules or requests made by people in authority“ [Oxf05].
Eine weitere Definition liefert Helma Quentmeier im „Praxishandbuch Compliance“: „Compliance bezeichnet die Gesamtheit aller zumutbaren Maßnahmen, die das gesetzes- und regelkonforme Verhalten eines Unternehmens, seiner Organisationsmitglieder und seiner Mitarbeiter im Hinblick auf alle gesetzlichen Ge- und Verbote begründet. [...] Es geht also um die Erfüllung und Konformität mit Gesetzen sowie mit Regeln, Grundsätzen und Spezifikationen. Compliance umfasst ebenfalls Standards und Konventionen, die klar definiert worden sind.“ [Que12].
Ebenfalls ist es schwierig eine aussagekräftige Übersetzung zu finden. Deshalb wurde durch die „Initiative für IT-Compliance in der Informations- und Datenverarbeitung in Deutschland“, als passendes deutsches Synonym für Compliance „Regeltreue“ von einer Jury ausgewählt [Mue].
Der Begriff Compliance umfasst also das Verhalten und die Maßnahmen, die notwendig sind, um die gestellten Anforderungen, die in verschiedenen Formen wie Regeln, Gesetzen aber auch Grundsätzen und Konventionen auftreten, zu erfüllen. Wird dies gewährleistet, so spricht man von Compliance bzw. Regeltreue. Dieses Konzept findet neben der Softwareentwicklung auch in anderen Bereichen Anwendung, wie zum Beispiel in der Rechts- und Wirtschaftswissenschaft, der Chemie, der Medizin und der Landwirtschaft [Com].

2.1.2 Compliance in der Softwareentwicklung

Jedoch wird gerade in der Softwareentwicklung die Bedeutung von Compliance immer wichtiger, da in immer mehr Anwendungs- und Lebensbereichen Software zum Einsatz kommt und diese entsprechend reguliert und reglementiert werden muss. Deshalb ist es nötig, passende Anforderungen zu formulieren, um Kriterien wie Sicherheit, Kompatibilität oder Bedienbarkeit zu gewährleisten.

Als Beispiel ist das *Bundesdatenschutzgesetz (BDGS)* zu nennen, welches die Erhebung, Verarbeitung und Nutzung von Daten regelt, um beispielsweise Missbrauch zu verhindern [SW]. So heißt es in §9:
„Öffentliche und nicht-öffentliche Stellen, die selbst oder im Auftrag personenbezogene Daten erheben, verarbeiten oder nutzen, haben die technischen und organisatorischen Maßnahmen zu treffen, die erforderlich sind, um die Ausführung der Vorschriften dieses Gesetzes, insbesondere die in der Anlage zu diesem Gesetz genannten Anforderungen, zu gewährleisten [...]“ [Jus].
Dies wird in den Anlagen zu §9 weiter spezifiziert:
„Dabei sind insbesondere Maßnahmen zu treffen, die [...] geeignet sind [...] zu gewährleisten, dass personenbezogene Daten bei der elektronischen Übertragung oder während ihres Transports oder ihrer Speicherung auf Datenträger nicht unbefugt gelesen, kopiert, verändert oder entfernt werden können [...]“ [Jus].
Das bedeutet für die Entwicklung von Software, dass, wenn die Software personenbezogene Daten verwendet, diese durch zum Beispiel kryptographische Verfahren schützen muss.
Da es tendenziell immer mehr Anforderungen aus verschieden Bereichen geben wird, die die Software regulieren sollen, ist es sinnvoll, sich mit der automatisierten Compliance-Analyse zu beschäftigen, da es für einen einzelner Entwickler bzw. eine Gruppe von Entwicklern schwierig ist, alle Anforderungen im Blick zu behalten.
Nur so kann eine Software entwickelt werden, die am Ende als Compliance bezeichnet werden kann.

2.1.3 IT-Grundschutz-Katalog

Um eine Compliance-Analyse durchführen zu können, muss ein Sammlung von Anforderungen als Referenz vorhanden sein. Im Falle des Riskfinder Werkzeugs wird dabei auf den *IT-Grundschutz-Katalog* des *Bundesamtes für Sicherheit in der Informationstechnik (BSI)* zurückgegriffen. Dabei handelt es sich um eine Sammlung von Regeln und Empfehlungen für typische Geschäftsprozesse, Anwendungen und IT-Systeme, um den Schutz von Informationen, Daten und Kontrollen in Instituten oder Unternehmen zu gewährleisten.
Der IT-Grundschutz-Katalog ist dabei in so genannte Bausteine eingeteilt, die je einer der fünf Schichten des *IT-Grundschutz-Modells* (Abbildung 2.1) zugeordnet sind. Das Modell klassifiziert die verschiedenen Risiken je nach Schwere anhand des Auftrittsorts. So ist ein Risiko, dass der „Schicht 5: Anwendungen“ zugeordnet ist und somit nur eine Anwendung betrifft weit weniger schwerwiegend als ein Risiko, dass die gesamte Infrastruktur tangieren und somit zur „Schicht 2: Infrastruktur“ zählt.
Dabei besteht jeder Baustein aus einer Beschreibung in Textform, einer Auflistung der einzelnen Gefährdungslagen und den empfohlenen Maßnahmen (Abbildung 2.2). Die Gefährdungslagen und Maßnahmen bestehen ebenfalls aus einer textuellen Beschreibung sowie Beispielen zum besseren Verständnis. Sie werden jeweils in einem Gefährdungs- bzw. Maßnahmenkatalogs zusammengefasst.

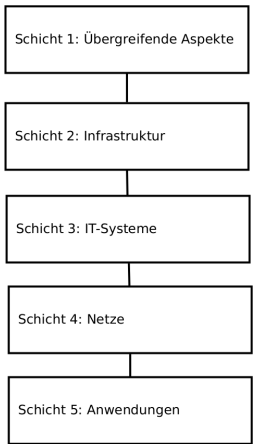
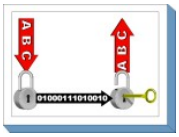


Abbildung 2.1: von [Sic], IT-Grundschutz-Modell

Bundesamt für Sicherheit in der Informationstechnik

Sie sind hier: Startseite Themen IT-Grundschutz-Kataloge Inhalt Bausteine Übergreifende Aspekte **B 1.7 Kryptokonzept**

B 1.7 Kryptokonzept



Beschreibung

Dieser Baustein beschreibt eine Vorgehensweise, wie in einer heterogenen Umgebung sowohl die lokal gespeicherten Daten als auch die zu übertragenen Daten wirkungsvoll durch kryptographische Verfahren und Techniken geschützt werden können. Dazu wird beschrieben, wie und wo in einer heterogenen Umgebung kryptographische Verfahren und die entsprechenden Komponenten eingesetzt werden können. Da beim Einsatz kryptographischer Verfahren sehr viele komplexe Einflussfaktoren zu betrachten sind, sollte hierfür ein Kryptokonzept erstellt werden.

In diesem Baustein wird daher beschrieben, wie ein Kryptokonzept erstellt werden kann. Beginnend mit der Bedarfsermittlung und der Erhebung der Einflussfaktoren geht es über die Auswahl geeigneter kryptographischer Lösungen und Produkte bis hin zur Sensibilisierung und Schulung der Anwender und zur Krypto-Notfallvorsorge.

Dieser Baustein kann auch herangezogen werden, wenn nur ein kryptographisches Produkt für eines der möglichen Einsatzfelder ausgewählt werden soll. Dann können einige der im folgenden beschriebenen Schritte ausgelassen werden und nur die für das jeweilige Einsatzfeld relevanten Teile bearbeitet werden.

Für die Umsetzung dieses Bausteins sollte ein elementares Verständnis der grundlegenden kryptographischen Mechanismen vorhanden sein. Ein Überblick über kryptographische Grundbegriffe findet sich in M 3.23 Einführung in kryptographische Grundbegriffe .

Gefährdungslage

Kryptographische Verfahren werden eingesetzt zur Gewährleistung von

- Vertraulichkeit,
- Integrität,
- Authentizität und
- Nichtabstreitbarkeit.

Daher werden für den IT-Grundschutz primär die folgenden Gefährdungen für kryptographische Verfahren betrachtet:

Organisatorische Mängel

G 2.1	Fehlende oder unzureichende Regelungen
G 2.2	Unzureichende Kenntnis über Regelungen
G 2.4	Unzureichende Kontrolle der Sicherheitsmaßnahmen
G 2.19	Unzureichendes Schlüsselmanagement bei Verschlüsselung

Menschliche Fehlhandlungen

G 3.1	Vertraulichkeits- oder Integritätsverlust von Daten durch Fehlverhalten
G 3.32	Verstoß gegen rechtliche Rahmenbedingungen beim Einsatz von kryptographischen Verfahren

Abbildung 2.2: von [Sic], Baustein B 1.7 Kryptokonzept

2.2 Wortschatz Universität Leipzig

Das *Wortschatz-Portal* ist ein Projekt der Abteilung Sprachverarbeitung des Instituts Informatik an der Universität Leipzig. Kern des Projekts ist es, ein Nachschlagewerk für Wörter zu erstellen. Darin soll ihr Gebrauch, Wortassoziationen sowie eine graphische Darstellung von Zusammenhängen zu anderen Worten bereitgestellt werden. Dafür werden die zur Zeit zugänglichen maschinenlesbaren Texte verwendet und analysiert. Ebenso werden verschiedene Webdienste angeboten, die es ermöglichen, zum Beispiel die Grundform eines Wortes zu ermitteln oder auch einen Beispielsatz, um die Bedeutung des Wortes sowie seinen Gebrauch zu veranschaulichen [Wor].
Im Rahmen der Modellanalyse nutzt das Riskfinder Werkzeug einen dieser Webservices, den *Synonyms Webservice*, um für ein Wort Synonyme zu ermitteln. So wird zum Beispiel für das Wort „Daten“ eine Liste von Synonymen (Angaben, Testdaten, Zahlen, Dateien, Fakten) bestimmt (Abbildung 2.3).
Dies ist Notwendig, da bei der Beschreibung von Modell und Anforderungen für die selbe Aussage unterschiedliche Begriffe verwendet werden können. Damit bei der Analyse trotzdem die Gleichheit erkannt wird, wird die Menge der Begriffe mit Hilfe von Synonymen erweitert. Bei einem Wortvergleich zwischen den Mengen wird dann durch eine Reihe von identischen Worten in beiden Mengen mit hoher Wahrscheinlichkeit die Gleichheit der Aussagen erkannt.
Somit kann ein korrektes Analyseergebnis erzielt werden, ohne dass beispielsweise der Entwickler bei der Modellierung auf den Wortschatz, der bei der Formulierung der Anforderungen verwendet wurde, beschränkt ist.

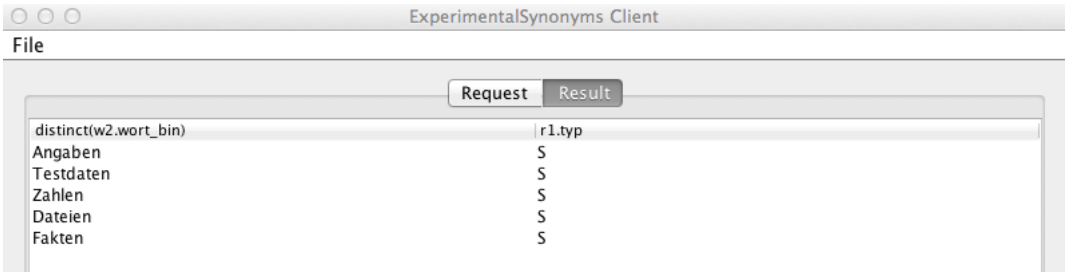


Abbildung 2.3: Ergebnis: Synonyme für das Wort Daten

2.3 Stopwords

Ebenfalls ist es sinnvoll, so genannte *Stopwords* (dt. Stoppworte), also die Menge aller Worte, die für die Bedeutung einer Aussage keine Information liefern, herauszufiltern, da sie durch ihr häufiges Auftreten den Analyseaufwand unnötig vergrößern oder sogar verfälschen können. So liefern zum Beispiel die Worte „die“, „das“, „und“, „eine“ sowie „den“ im Satz „Die Daten werden über das Netzwerk verschickt und eine Bestätigung an den Nutzer gesendet.“ keine relevante Information über die Aussage des Satzes. Diese wäre nach dem Entfernen der Stopwords mit „Daten werden über Netzwerk verschickt, Bestätigung an Nutzer gesendet“ immer noch die gleiche. Durch das Entfernen unnötiger Worte wird somit eine verbesserte Analysequalität erreicht,

da weniger Worte zu betrachten sind sowie die Aussagekraft auf das Notwendigste reduziert ist [Pes10].

2.4 SecReq

Neben den Stopword ist außerden eine Differenzierung zwischen den einzelnen Worten in der Analyse sinnvoll. So sind einige Worte für die Bedeutung wichtiger als andere. Zu diesem Zweck findet das *SecRec Projekt* der Universität Hannover Verwendung in Riskfinder.
Ziel des SecRec Projekts ist es, die besten Sicherheitskonzepte und -praktiken für Programmierer und Entwickler verständlich zu machen und sie bei der Umsetzung zu unterstützen. Dies ist notwendig, weil ständig neue Konzepte entwickelt werden, diese jedoch meist nicht von den Entwicklern umgesetzt werden. Gründe dafür sind beispielsweise fehlende Kenntnisse oder nicht vorhandene Unterstützung durch Software.
Ein in diesem Rahmen entwickeltes Tool bietet die Möglichkeit, Worte anhand ihrer Relevanz für den Bereich Sicherheit zu bewerten. Es findet im Riskfinder Anwendung, da somit wie bei den Stopwords für die Analyse nicht relevante Worte herausgefiltert werden können, wodurch die Analysequalität verbessert wird [Hou+].

3 Riskfinder

3.1 Einleitung Riskfinder

Das Werkzeug Riskfinder ist ein von Marc Peschke im Rahmen seiner Diplomarbeit [Pes10] entwickelte Erweiterung für das UMLsec-Tool.
Es ermöglicht es ein, Modell, welches in UML spezifiziert ist, zu laden und dieses auf Grundlage einer Sammlung von sicherheitsrelevanten Eigenschaften zu analysieren. Es unterstützt somit den Entwickler dabei, potenziellen Gefahrenbereiche zu ermitteln und durch die jeweils empfohlenen Maßnahmen zu beheben. Des Weiteren werden die betroffenen Modellkomponenten je nach Relevanz der zugeordneten Gefahren farblich markiert, sodass ein schneller Überblick möglich ist.
Dabei besteht die Sammlung der sicherheitsrelevanten Eigenschaften, im folgenden *Pattern-Repository*, aus so genannten *Sicherheits-Pattern's*, die auf der Basis des IT-Grundschutz-Katalogs vom Nutzer selbstständig zusammengestellt werden können.

3.2 Analysealgorithmus

Wie bereits im ersten Kapitel beschrieben, liegt die Hauptschwierigkeit bei der Analyse eines Modells unter dem Aspekt textueller Anforderungen darin, dass es nicht direkt ersichtlich ist, ob diese eingehalten werden.
Der Analysealgorithmus hat also die Aufgabe, auf der Grundlage des Pattern-Repositories zu einem Modell alle passenden Patterns zu ermitteln und den jeweiligen Komponenten zuzuordnen. Dies wird durch die separate Betrachtung der einzelnen Komponenten des Modells ermöglicht. Der Algorithmus bekommt als Eingabe die jeweilige textuelle Beschreibung der einzelnen Komponenten und generiert als Ausgabe eine Liste der der jeweiligen Komponente zugeordneten Patterns, welche nach Abschluss der Analyse in einem separaten Fenster angezeigt werden. Außerdem werden die Modellkomponenten passend eingefärbt.
Dabei wird die textuelle Beschreibung der einzelnen Modellkomponenten sowie die der Anforderungen, in diesem Fall die Sicherheits-Pattern, durch Synonyme erweitert und durch Filtermechanismen verkleinert. Die daraus entstehenden Wortvektoren werden dann jeweils miteinander verglichen und bei genügend großer Übereinstimmung zwischen eines Komponenten- und eines Patternvektors das jeweilige Sicherheits-Pattern der Komponente zugeordnet.

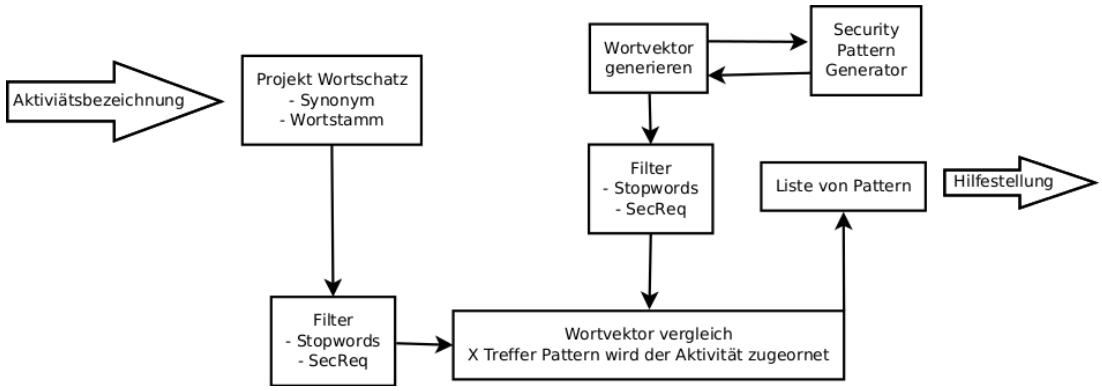


Abbildung 3.1: aus [Pes10], Riskfinder Analyse Algorithmus

Die genaue Vorgehensweise (Abbildung 3.1) wird nun näher betrachtet.
Im ersten Schritt werden zu jeder textuellen Beschreibung einer Modellkomponente mit Hilfe des Projekts „Wortschatz“ zu allen Worten der Wortstamm und Synonyme ermittelt.
Die dadurch entstandene Menge an Worten wird dann durch zwei verschiedene Filtermechanismen auf die für die Beschreibung der Aktion notwendigen Worte reduziert. Zum einen werden Stopworte entfernt, zum anderen durch das Projekt „SecReq“ ermittelte nicht sicherheitsrelevante Worte herausgefiltert. Somit entsteht für jede Modellkomponente ein Komponenten-Wortvektor.
Die für den Vergleich notwendigen Pattern-Wortvektoren werden aus den einzelnen Patterns des Repositories generiert. Sie enthalten die textuellen Beschreibungen der dem Pattern zugeordneten Gefahren und Maßnahmen. Die Pattern-Wortvektoren durchlaufen die selben Filter wie die Komponenten-Wortvektoren, um nicht relevante Worte zu entfernen, da sonst eine zu große Anzahl an Übereinstimmungen zu erwarten wäre.
Im nächsten Schritt wird ermittelt, welche Patterns den einzelnen Komponenten zugeordnet werden. Dazu wird jeweils ein Wortvektor einer Modellkomponente mit dem Wortvektor eines Patterns verglichen, indem die Anzahl der Worte, die in beiden Vektoren vorhanden sind, ermittelt wird. Wird dabei eine bestimmte Anzahl überschritten, kann angenommen werden, dass das Pattern eine Relevanz für die aktuell betrachtete Komponente hat und wird somit der Komponente zugeordnet.
Dies wird für alle Komponenten im Modell und alle Patterns wiederholt. Zum Abschluss wird eine Liste ausgegeben, die alle Komponenten sowie die dazu gefundenen Patterns auflistet. Außerdem werden die Komponente je nach Einstufung der ihr zugeordneten Patterns noch entsprechend eingefärbt.

3.3 Anpassungsmöglichkeiten und Optionen

Die für die Analyse notwendigen Patterns können, wie bereits erwähnt, vom Nutzer selbstständig erstellt werden. Dafür wird von Riskfinder eine Eingabemaske zur Verfügung gestellt (Abbildung 3.2), mit der man ein neues Pattern erstellen kann. Dabei wird dem Pattern neben einem Namen auch ein Bereich des IT-Grundschutz-Modells (Abbildung 2.1) zugewiesen. Außerdem können aus G1 bis G5 verschiedene

Risiken aus dem Gefahrenkatalog aufgelisteten und durch Markierung dem Pattern zugeordnet werden. Ebenso werden die Maßnahmen aus M1 bis M6 ausgewählt. Mit „add to Repository“ wird das Pattern dem Repository hinzugefügt.
Dadurch können die gewünschten Anforderungen vom Nutzer selbstständig spezifiziert werden. Weiter können bereits bestehende Pattern gelöscht, ein Backup des Repositories erstellt oder das gesamte Repository gelöscht werden, wobei zuvor ein Backup im Home-Verzeichnis des Benutzers angelegt wird.

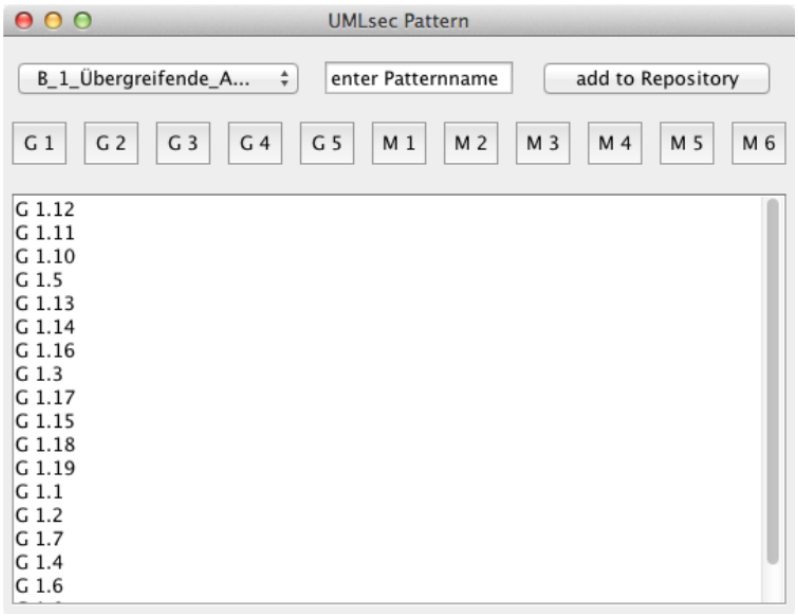


Abbildung 3.2: aus [Pes10], Neues Pattern erstellen

3.4 Beispiel

Zur Veranschaulichung wird als Beispiel ein UML Aktivitätsdiagramm (Abbildung 3.3) analysiert. Dies geschieht auf der Grundlage eines Sicherheits-Repositories, dass auf den Bausteinen des IT-Grundschutz-Katalogs basiert.
Das Aktivitätsdiagramm behandelt das „Verhalten einer Cloud“ bei Anfrage einer Dienstleistung durch den Kunden. Bei positiver Authentifikation wird die Dienstleistung erbracht und eine Mail als Antwort versendet.
Nach Abschluss der Analyse ergibt sich als Ergebnis das eingefärbte UML Modell (Abbildung 3.4)

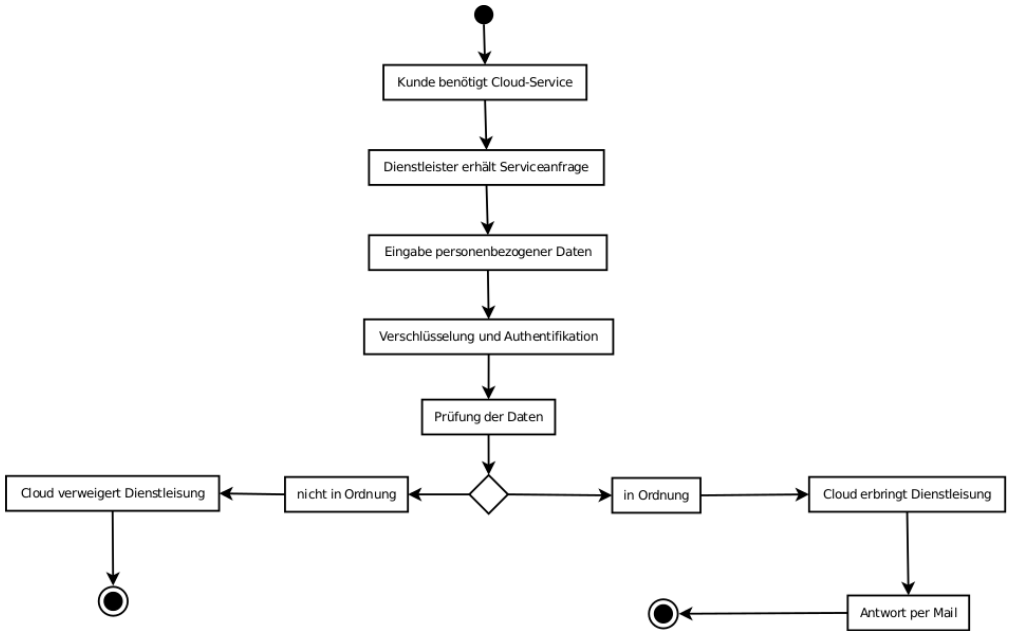


Abbildung 3.3: aus [Pes10], Beispiel UML Aktivitätsdiagramm Cloud

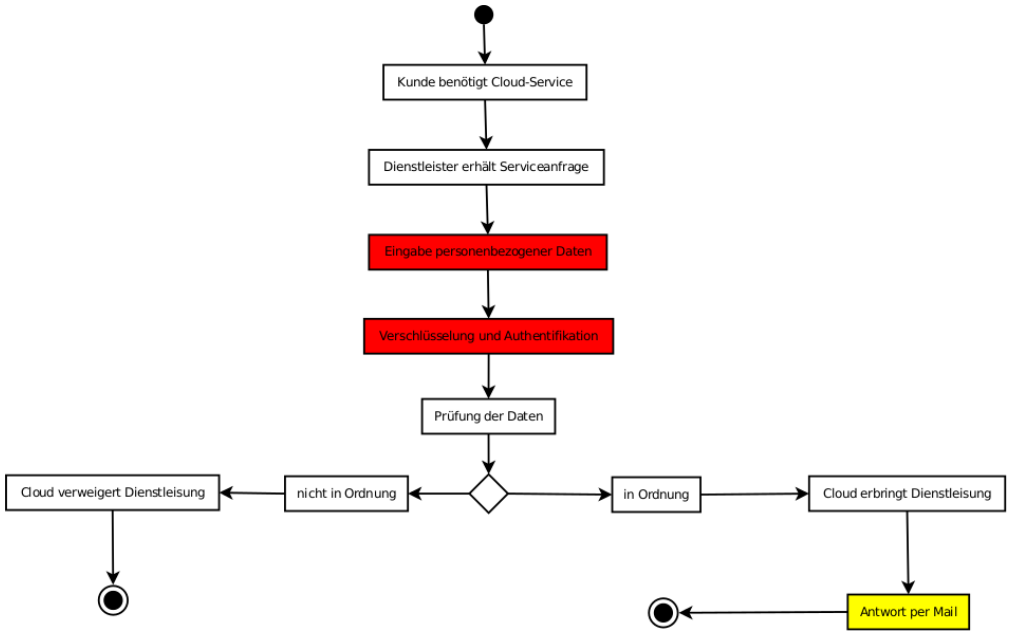


Abbildung 3.4: aus [Pes10], Ergebnis der Analyse

und eine Liste mit jeweils den Aktionen zugeordneten Patterns (Abbildung 3.5) wird angezeigt. Im Beispiel entsprechen diese gerade den Bausteinen des IT-Grundschutz-Katalogs.
So wird zum Beispiel bei der Aktion „Eingabe personenbezogener Daten“ als potenzielles Risiko das Pattern „Kryptokonzept“ angegeben, da in IT-Grundschutz-Katalog bei der Eingabe personenbezogener Daten als Richtlinie empfohlen wird, diese nur verschlüsselt zu übertragen, um das Abfangen und Missbrauchen von Da-

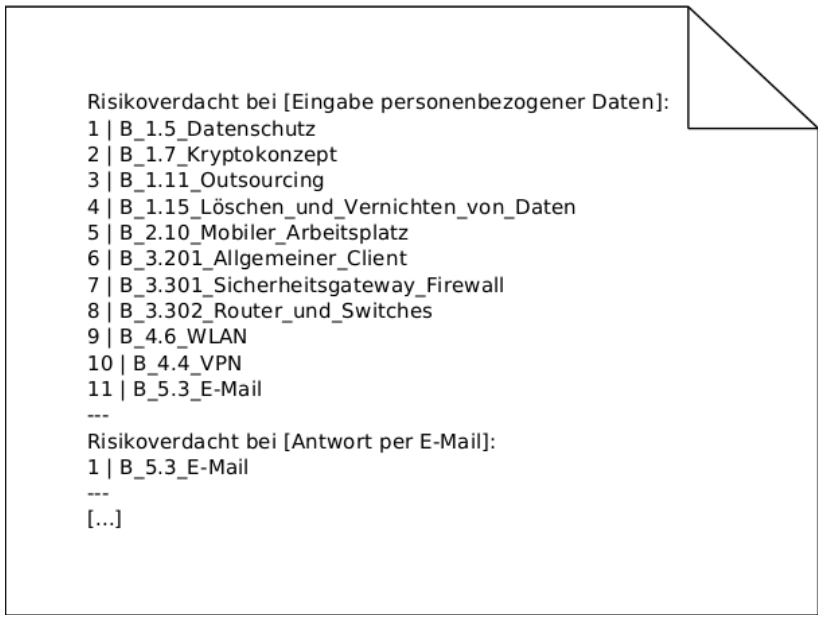


Abbildung 3.5: aus [Pes10], Ergebnisausgabe

ten zu verhindern. Weitere Risikobereiche betreffen die Datenschutzrichtlinien, die beim Umgang mit Nutzerdaten eingehalten werden müssen oder auch spezielle Anforderungen, die den Umgang bei mobilen Arbeitsplätzen betreffen. Durch die rote Einfärbung wird deutlich, dass es notwendig ist, sich mit dem Risiken und Anforderungen auseinanderzusetzen.
Somit bekommt der Entwickler einen guten Überblick darüber, welche Bereiche seines Modells durch Risiken und Anforderungen betroffen sind und er kann sich mit den jeweils in den Patterns spezifizierten Maßnahmen darum kümmern. Somit ist schon während der Entwicklung sichergestellt, dass Risiken und Anforderungen beachtet und behandelt werden und dadurch Compliance erreicht.

4 Kritische Auseinandersetzung und Ausblick

4.1 Kritische Auseinandersetzung

Die Analyse eines Software-Modells unter dem Aspekt von textuellen Anforderungen wird durch das Riskfinder Werkzeug elegant umgesetzt. Vor allem die Problematik der Vergleichbarkeit von Text und Modell wird durch die Separation des Modells und die Einzelanalyse der Kompotten gelöst und ermöglicht so eine effiziente Analyse.
Positiv ist dabei die Umsetzung mittels der recht simplen Idee, die betroffenen Patterns mittels dem Vergleich von Wortvektoren zu ermitteln. Ebenso ist die Erweiterung dieser durch Synonyme sinnvoll, da der Entwickler so bei der Modellierung nicht in der Wortwahl beschränkt ist.
Kritisch ist anzumerken, dass sich die Analyse ausschließlich auf die einzelnen Komponenten eines Modells bezieht und nicht die Gesamtstruktur berücksichtigt. Dies kann zu redundanten oder falschen Ergebnissen führen, zum Beispiel wenn eine Komponente, die eine Datenübertragung charakterisiert, als Risiko bezüglich Datenschutz eingestuft wird, jedoch der Aufbau der Verbindung in einer vorherigen Komponente bereits diesen Risikoaspekt unterbindet.
Ebenso werden durch textuelle Beschreibungen sicherheitsfördernde Maßnahmen, die in einer Komponente beschrieben werden, als Übereinstimmung gewertet, da nur das Wort und nicht der Kontext betrachtet wird. So wird zum Beispiel die Formulierung „Daten verschlüsseln“ so interpretiert, dass sowohl das Wort „Daten“ als auch das Wort „Verschlüsseln“ sowohl in der Komponente als auch im Pattern Kryptokonzept vorkommt und somit beide Worte als Übereinstimmung gewertet werden, obwohl das Sicherheitsrisiko, welches dieses Pattern behandelt, durch diese Komponente minimiert wird. Somit wird fälschlicherweise das Pattern der Komponente zugewiesen.
Dennoch ist Riskfinder für eine erste Analyse eines Modells durchaus ein sinnvolles und nützliches Werkzeug, um potentielle Risiken in einem Modell zu ermitteln und somit die notwendigen Maßnahmen schon während der Entwicklung umsetzen zu können.

4.2 Mögliche Erweiterungen

Jedoch kann durch einige Erweiterungen der Nutzen von Riskfinder verbessert sowie das Einsatzfeld vergrößert werden.
So ist es zwar möglich, ein Repository als Backup zu speichern, ein bereits vorhan-

denes zu laden ist jedoch nicht möglich. Da Riskfinder außerdem für die Identifikation sicherheitskritischer Bereiche entwickelt wurde, Compliance jedoch auch andere Bereiche in der Softwareentwicklung wie Nutzungsverhalten oder Schnittstellendefinition betreffen, sollte es durch eine Erweiterung möglich sein, auch Pattern's unabhängig von IT-Grundschutzkatalog zu erstellen.
Ebenfalls sollte es möglich sein, neben UML weitere Modellformate wie BPMN zu analysieren.

4.3 Weitere Einsatzgebiete

Neben den möglichen Erweiterungen kann man Riskfinder auch in anderen Kontexten und Gebieten nutzbringend verwenden.
So kann es neben der Sicherheitsanalyse auch generell bei der Erkennung von Risiken eingesetzt werden, sowie bei der Analyse von Abläufen und Konstruktionen helfen, denen je nach Kontext spezielle Abläufe bzw. Konstruktionsarten zugewiesen sind. Als Beispiel wäre zu nennen, dass Bedienelemente in einem Userinterface auch für Menschen mit Handicap bedienbar sein müssen. Dies könnte durch eine Anforderung für das Erstellen von Interfaces ermöglicht werden, so dass der Entwickler bei der Modellierung darauf hingewiesen wird und dementsprechende Voraussetzungen schafft.
Ebenso kann es eingesetzt werden, um bestimmte Komponenten in Software einheitlich zu modellieren. Dies ist dadurch möglich, dass man als Regel den gewünschten Standard spezifiziert. Bei der Analyse wird dies dann bei den beschriebenen Komponenten angezeigt, so dass der Entwickler dies berücksichtigen kann. Dadurch wird eine homogenere Softwarestruktur ermöglicht.

4.4 Fazit

Als Abschluss kann man sagen, dass Compliance in der heutigen Softwareentwicklung einen immer größeren Raum in Anspruch nimmt. Deshalb ist es wichtig, durch automatisierte Analyse die betroffenen Komponenten zu ermitteln und somit die Sicherheit und die Nutzbarkeit von Software zu erhöhen und gleichzeitig den Entwicklungsprozess zu vereinfachen. Werkzeuge wie Riskfinder können dafür gerade in der Entwicklungsphase der Software eine erste Grundlage bilden. Jedoch muss es noch viele Erweiterungen und Verbesserungen geben, damit auch in Zukunft Software entwickelt wird, die Compliance ist.

Literatur

[Com] Compliance-Magazin. *Compliance-Lexikon*. URL: http://www.compliancemagazin.de/compliancelexikon/ (besucht am 24. 01. 2014).

[Hou+] Siv Houmb u. a. *SecReq*. URL: http://www.se.uni-hannover.de/pages/en:projekte_re_secreq/ (besucht am 28. 01. 2014).

[Jus] Bundesministerium der Justiz und für Verbraucherschutz. *Bundesdatenschutzgesetz*. URL: http://www.gesetze-im-internet.de/bdsg_1990/ (besucht am 27. 01. 2014).

[Mue] Dietmar Mueller. *Regeltreue statt Compliance*. URL: http://www.silicon.de/41502486/regeltreue-statt-compliance/ (besucht am 24. 01. 2014).

[Oxf05] Oxford. *Oxford Advanced Learner's Dictionary*. 7. Auflage. Oxford Universität, Mai 2005.

[Pes10] Marc Peschke. „Werkzeuggestützte Modell-basierte Sicherheitsanalyse für IT- Sicherheitsmanagementbasierte Sicherheitsanalyse für IT-Sicherheitsmanagement". Magisterarb. Technische Universität Dortmund, Fakultät für Informatik, 2010.

[Que12] Helma Quentmeier. *Praxishandbuch Compliance: Grundlagen, Ziele und Praxistipps für Nicht-Juristen*. Gabler, 2012.

[Sic] Bundesamt für Sicherheit in der Informationstechnik. *IT-Grundschutz-Kataloge*. URL: https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/itgrundschutzkataloge_node.html (besucht am 24. 01. 2014).

[SW] Artur Strasser und Michael Wittek. *IT-Compliance*. URL: http://www.gi.de/nc/service/informatiklexikon/detailansicht/article/it-compliance.html (besucht am 27. 01. 2014).

[Wor] Universität Leipzig Wortschatz. *Projekt Wortschatz*. URL: http://wortschatz.uni-leipzig.de (besucht am 24. 01. 2014).