IT Compliance mit kontextuellen Sicherheitsanforderungen

Michael Brunner · Ruth Breu

Universität Innsbruck, Institut für Informatik
Forschungsgruppe Quality Engineering
{michael.brunner | ruth.breu}@uibk.ac.at

Zusammenfassung

In diesem Beitrag stellen wir ein Framework und dessen prototypische Umsetzung zur kontextuellen Verwaltung von Sicherheitsanforderungen in Unternehmen vor. Unser Ansatz verbindet Sicherheitsanforderungen mit Unternehmensmodellen und vereinfacht durch automatische Workflows die Einhaltung komplexer Rahmenbedingungen in sich stetig weiter entwickelnden Systemen. High-Level Anforderungen aus der IT Compliance werden in einem kontinuierlichen, kooperativen Prozess weiter detailliert bis hin zur Definition konkreter technischer Anforderungen für benötigte IT-Services und Infrastruktur-Komponenten. Unser Framework vereinfacht die laufende Erfassung des Erfüllungsgrades einzelner Anforderungen sowie das systematische Reporting bei gleichzeitiger Einbindung aller Beteiligten aus unterschiedlichen Unternehmensbereichen und Domänen. Die prototypische Implementierung unseres Ansatzes legt großes Augenmerk auf größtmögliche Automatisierung sowie die systematische Unterstützung von IT-Sicherheitsstandards und wurde bereits erfolgreich bei einem IT Service Provider und einem international tätigen Auditor evaluiert.

1 Einleitung

Die systematische Erfassung und Verwaltung von Sicherheitsanforderungen stellt für viele Unternehmen eine große Herausforderung dar. Zum einen, da eine Vielzahl gesetzlicher Regulatorien, vertraglicher Vereinbarungen und branchenüblicher Standards zu berücksichtigen sind, zum anderen da diese Sicherheitsanforderungen stets auf sich ständig verändernde unternehmerische Rahmenbedingungen angepasst werden müssen. Geschäftsprozesse werden optimiert, die Systemlandschaft wird entsprechend angepasst und laufend müssen neu identifizierte Bedrohungsszenarien berücksichtigt werden. Zudem stellen die stetig wachsende Einbindung von Cloud-Diensten, die Abhängigkeit von externen Dienstleistern oder Datenverarbeitern sowie die zunehmende Integration unterschiedlicher Systemkomponenten im Bereich Cyber-Physical Systems (CPS) erhöhte Ansprüche an die IT Sicherheit. Standards aus dem Bereich des IT Sicherheitsmanagements können Unternehmen durch ihren eher generischen Charakter, die meist stark auf Sicherheitsprozesse beschränkte Sicht und den geringen Fokus auf die für Unternehmen relevanten, meist sehr unterschiedlichen Einsatzgebiete, nur bedingt unterstützen [Sipo06, SiWi09]. Die effiziente Einhaltung von heterogenen Sicherheitsanforderungen in diesem komplexen Spannungsfeld wird ebenso wie die wiederkehrenden Kosten für Compliance Audits in [BaFo10, SPOA$^+$12, TBDM12] als große Herausforderung im Bereich IT Sicherheitsmanagement angesehen.

P. Schartner · P. Lipp (Hrsg.) · D•A•CH Security 2014 · syssec (2014) 136-147.

Dieser Beitrag beschreibt einen Ansatz zur kontextuellen Verwaltung von Sicherheitsanforderungen auf Basis eines modell-unterstützten kooperativen Management-Prozesses, der Wissensträger aus den unterschiedlichen Unternehmensbereichen zur Erfassung und laufenden Überwachung von Sicherheitsanforderungen einbindet. Damit adressieren wir die in [SpBa10] präsentierten Erkenntnisse, die eine Qualitätssteigerung im Sicherheitsmanagement (erhöhte Bewusstseinsbildung, besserer Abgleich zwischen Geschäfts- und Sicherheitszielen, Verbesserung von Sicherheitsmaßnahmen) durch eine breite Stakeholder Beteiligung prognostiziert. Durch entsprechende Tool-Unterstützung wird sicher gestellt, dass unmittelbar auf nicht erfüllte Anforderungen sowie Veränderungen im Unternehmenskontext reagiert wird. Interne und externe Audits werden durch umfassende Berichtsmöglichkeiten und eine nachvollziehbare Dokumentation von Sicherheitsanforderungen während ihres gesamten Lebenszyklus unterstützt.

Bei der Entwicklung unseres Frameworks haben wir uns an Hevner et al's [HMPR04] Prinzipien zu Design Science orientiert, ein Tool-getriebenes Framework spezifiziert und den zugehörigen Prototypen implementiert. Das Framework wurde bereits in frühen Entwicklungsphasen von Industrie-Partnern und potentiellen Anwendern evaluiert, um sicher zu stellen, dass durch unseren Ansatz relevante bis dato unzureichend gelöste Probleme im Bereich IT Sicherheitsmanagement adressiert werden. Durch die Weiterentwicklung und eine abschließende Evaluierung unseres Prototypen im Rahmen des FP7 EU-Projektes *PoSecCo*[1] können wir zeigen, dass der Einsatz von ADAMANT das Potential besitzt, die Qualität und Effizienz bei der Erhebung und Verwaltung von Sicherheitsanforderungen sowie bei Zertifizierungsverfahren und Audits zu verbessern.

In Abschnitt 2 stellen wir das Framework anhand des entwickelten Modells für Sicherheitsanforderungen vor. Wir erläutern den Lebenszyklus von Sicherheitsanforderungen, die Interaktion zwischen dem Sicherheitsmodell und der Unternehmenslandschaft sowie das zugrunde liegende Vorgehensmodell. Abschnitt 3 beschreibt den entwickelten Prototypen, der unsere Konzepte umsetzt. Abschließend folgt eine Abgrenzung unseres Ansatzes hinsichtlich bestehnder Arbeiten sowie ein Ausblick auf zukünftige Entwicklungen.

2 Das ADAMANT Framework

Das in diesem Bericht vorgestellte ADAMANT Framework wurde mit Hinblick auf gängige Standards und Vorgehensmodelle im Bereich IT Sicherheitsmanagement entwickelt und soll deren Umsetzung verbessern. Das hierfür entwickelte Meta-Modell für Sicherheitsanforderungen garantiert die Definition kompatibler und transparenter Modelle, die Unternehmen bei der Erfüllung der jeweils relevanten Sicherheitsstandards unterstützen. Im Rahmen eines Top-Down-Ansatzes werden Sicherheitsanforderungen hierarchisch definiert. Dabei können die jeweils verantwortlichen Stakeholder die Anforderungen ihrer Domäne weiter konkretisieren und neue untergeordnete Anforderungen mit entsprechend angepassten Verantwortlichkeiten definieren. Zudem wird jede Sicherheitsanforderung mit den zu schützenden Elementen aus der Unternehmenslandschaft verknüpft. Damit werden Sicherheitsanforderungen im tatsächlichen Kontext der Unternehmenslandschaft betrachtet und das iterativ erarbeitete Modell ist auf die jeweiligen Rahmenbedingungen und Bedürfnisse des Unternehmens fokussiert. Im folgenden referenziert die Bezeichnung *Sicherheitsmodell* auf die Definition von Sicherheitsanforderun-

[1] Mehr Informationen zum EU-Projekt PoSecCo können auf der offiziellen Homepage http://www.posecco.eu eingesehen werden. Unser Framework wird im Rahmen der zugehörigen Dokumente unter dem Arbeitstitel CoSeRMaS (Collaborative Security Requirements Management System) geführt.

gen und deren Verknüpfungen und mit *Unternehmensmodell* verweisen wir auf die dokumentierte Unternehmenslandschaft.

Durch die gleichzeitige Einbindung von Unternehmensmodellen und die Definition formaler Regelwerke kann ADAMANT das Sicherheitsmodell automatisch erweitern und adaptieren, um so dynamisch auf veränderte Rahmenbedingungen zu reagieren und die Einhaltung von Sicherheitsanforderungen auch im evolutionären Unternehmenskontext dauerhaft und zeitnah zu garantieren. Dabei unterstützt das ADAMANT Framework bestehende Informationsquellen in Unternehmen und erlaubt durch generische Schnittstellen die (gleichzeitige) Einbindung beliebiger Tools aus den Bereichen Enterprise Architecture Management (EAM) oder dem Konfigurationsmanagement. Das Sicherheitsmodell erlaubt in Folge die Analyse von Abhängigkeiten zwischen einzelnen Elementen oder Element-Gruppen aus der angebundenen Unternehmensarchitektur und unterstützt verantwortliche Stakeholder nicht nur bei der Erfassung von Sicherheitsanforderungen sondern zusätzlich auch bei deren qualitativen Bewertung im tatsächlichen Unternehmens-Kontext.

2.1 Meta-Modell für Sicherheitsanforderungen

Das für unseren Ansatz entwickelte Meta-Modell für Sicherheitsanforderungen ist das Ergebnis einer Analyse von IT-Sicherheitsstandards und Best Practices aus dem Bereich IT Governance. Ziel war die Entwicklung eines Modells, das in der Lage ist, bestehende Standards abzubilden und diese gleichzeitig mit den betroffenen Elementen der Unternehmenslandschaft zu verknüpfen. Die hier vorgestellte Version unseres Meta-Modells wurde unter Berücksichtigung von ISO 27001 [ISO13], den Control Objectives for Information and Related Technology (COBIT) [IT 07], der IT Infrastructure Library (ITIL) [Long12], dem IT Grundschutzkatalog [BSI05] und dem Payment Card Industry Data Security Standard (PCI-DSS) [PCI10] definiert.

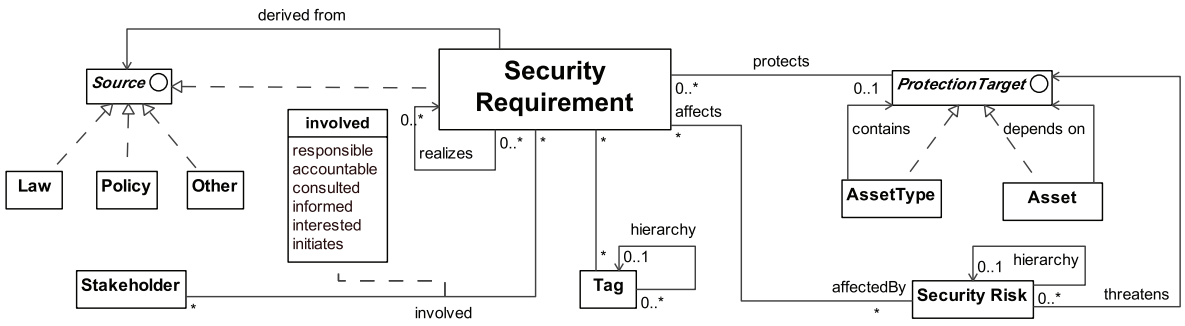

Abb. 1: Meta-Modell für Sicherheitsanforderungen

Abbildung 1 zeigt unser Meta-Modell für Sicherheitsanforderungen als UML Klassendiagramm. Die einzelnen Klassen unseres Meta-Modells und deren Assoziationen werden im Folgenden kurz erläutert:

- Die Klasse *SecurityRequirement* stellt das zentrale Element unseres Modells dar und dient der Modellierung des eigentlichen Sicherheitsmodells. Sicherheitsanforderungen sind prinzipiell hierarchisch organisiert, wobei eine Sicherheitsanforderung eine beliebige Anzahl an untergeordneten sowie übergeordneten Anforderungen besitzen kann. Jede

Sicherheitsanforderung ist mit maximal einem Protection Target verbunden, wobei eine temporale Abhängigkeit in der Assoziation *protectsDuring* hinterlegt werden kann.

- Das Interface *Protection Target* stellt die Verbindung zur Unternehmenslandschaft dar. Es können damit Gruppen oder Typen von Elementen via *AssetType* oder einzelne Elemente via *Asset* definiert werden.
- Die Klasse *Stakeholder* dient der Modellierung relevanter Stakeholder. Sicherheitsanforderungen können mit einer beliebigen Anzahl von Stakeholdern in unterschiedlichen Rollen verknüpft werden.
- Über die Klasse *Sources* können unterschiedliche Quellen für Sicherheitsanforderungen definiert werden. Jede Sicherheitsanforderung ist entweder direkt oder indirekt (über die hierarchische Vererbung von übergeordneten Anforderungen) mit zumindest einer Quelle verknüpft.
- Mit Hilfe der Klasse *Tag* können hierarchisch organisierte Tags definiert werden. Damit kann eine orthogonale Hierarchie für Sicherheitsanforderungen realisiert werden.
- Die Klasse *Security Risk* erlaubt die Einbindung von Risk Management Ansätzen in das Sicherheitsmodell. Einzelne Risiken können Sicherheitsanforderungen beeinflussen und sind über die Assoziation *threatens* mit dem Unternehmensmodell verknüpft.

Die Definition von Sicherheitsanforderungen erfolgt in Anlehnung an die geschäftliche Praxis und die bestehenden Standards in Form von natürlichsprachlichen Texten. Der Einsatz einer formalen Sprache zur Beschreibung von Sicherheitsanforderungen wird zu Gunsten der breiten Beteiligung von Stakeholdern aus unterschiedlichen Unternehmensbereichen und Domänen bewusst vermieden. Eine Schnittstelle zur Einbindung unterschiedlicher Policy-Sprachen ist konzipiert und kann ergänzend eingesetzt werden. Zudem verzichten wir auf eine dezidierte Unterscheidung zwischen Sicherheitszielen, Sicherheitsanforderungen und umsetzenden Maßnahmen – wir gehen davon aus, dass die Sicherheitsanforderungen auf oberster Ebene abstrakt als Ziele und dass Sicherheitsanforderungen auf unterster Ebene bereits als konkrete Maßnahmen formuliert werden. Eine zusätzliche Unterscheidung würde das unter Abschnitt 2.5 präsentierte Vorgehensmodell unnötig erschweren.

Für die Einbindung von Unternehmensmodellen sind automatische Schnittstellen vorgesehen. Allerdings erfolgt lediglich die Definition von geeigneten Element-Gruppen (vgl. *AssetType* in Abbildung 1) im Wirkungsbereich von ADAMANT, die einzelnen Elemente hingegen (vgl. *Asset* in Abbildung 1) werden aus bestehenden Systemen importiert. Dabei kann ein heterogener Ansatz verfolgt werden, indem je nach Asset Typ die eigentlichen Assets aus verschiedenen Datenquellen importiert werden.

2.2 Sicherheitsstatus und State Propagation

Bei der Verwaltung von Sicherheitsanforderungen liegt der Fokus auf der Erfassung ihres Status – ob sie erfüllt sind oder nicht. Um diesen Status nachvollziehbar zu erheben, arbeitet das ADAMANT Framework mit explizit erfassten Bestätigungen für die Erfüllung von Sicherheitsanforderungen. Diese Bestätigungen können entweder durch die verantwortlichen Stakeholder selbst oder auch durch automatische Regeln (siehe Abschnitt 2.4) erteilt werden. Bestätigungen besitzen eine festgelegte Gültigkeitsdauer, nach deren Ablauf der Status der Sicherheitsanforderung automatisch geändert wird sofern keine erneute Bestätigung rechtzeitig erteilt wird (siehe Abschnitt 2.5). Zudem wird durch automatische, formal definierte State Propagation sicher gestellt, dass diese Status-Information auf übergeordnete Sicherheitsanforderungen übertragen

werden. Die State Propagation basiert auf in Object Constraint Language (OCL) [OMG12] formulierten Bedingungen für die Statusänderung von Sicherheitsanforderungen basierend auf ihren untergeordneten Anforderungen und deren Status. Damit werden Statusänderungen entlang der Hierarchie im Sicherheitsmodell nach oben eskaliert und Stakeholder können unmittelbar Maßnahmen ergreifen. Abbildung 2 zeigt ein UML Klassendiagramm mit den relevanten Klassen für die State Propagation und die Bestätigung von Sicherheitsanforderungen:

- Die Klasse *Confirmation* modelliert die Bestätigung von Sicherheitsanforderungen. Neben dem Stakeholder, der die Bestätigung gegeben hat und deren Gültigkeitsdauer, wird zur besseren Nachverfolgbarkeit zusätzlich eine Begründung festgehalten.
- Die Klasse *Revocation* repräsentiert die Aufhebung einer einzelnen Bestätigung. Auch hier wird der verantwortliche Stakeholder sowie eine Begründung gespeichert.
- Die beiden Klassen *FulfillmentModel* und *RevalidationModel* dienen der Definition der automatische State Propagation. Das *FulfillmentModel* bestimmt die Bedingungen, zu denen ein automatischer Statuswechsel erfolgen soll. Die Bedingungen für diesen Statuswechsel können somit individuell für jede Sicherheitsanforderung mit Hilfe der formalen Sprache OCL bestimmt werden (z.B. alle untergeordneten Anforderungen müssen erfüllt sein, zumindest eine muss erfüllt sein, nicht alle untergeordneten Anforderungen mit hoher Priorität sind erfüllt). *RevalidationModels* bestimmen hingegen die Gültigkeitsdauer von Bestätigungen und wie auf den bevorstehenden Ablauf von Bestätigungen reagiert werden soll.

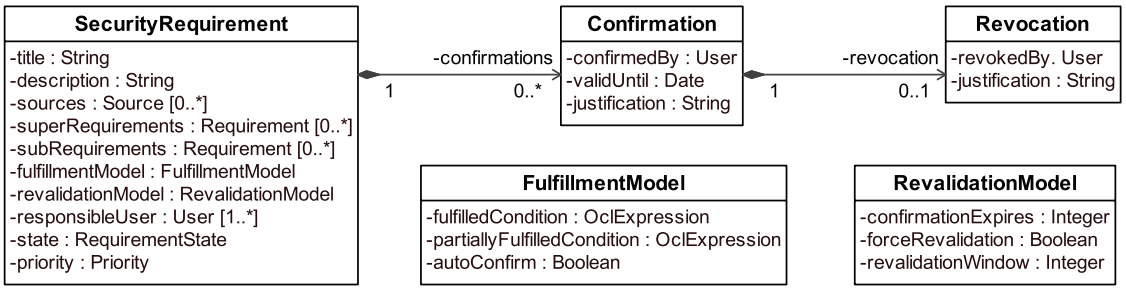

Abb. 2: UML Klassendiagramm für Sicherheitsanforderungen und deren Bestätigungen

Der Sicherheitsstatus des gesamten Systems wird durch den Status der einzelnen Sicherheitsanforderungen bestimmt. Dabei unterscheidet das ADAMANT Framework drei unterschiedliche Stati:

FULFILLED: Die Erfüllung der Sicherheitsanforderung wurde entweder durch den verantwortlichen Stakeholder, durch die automatische State Propagation oder durch eine formale Regel bestätigt. Die Bestätigung ist zum aktuellen Zeitpunkt noch gültig.

PARTIALLY_FULFILLED: Entsprechend der für diese Sicherheitsanforderung per *FulfillmentModel* definierten Bedingung sind nicht alle (relevanten) untergeordneten Anforderungen erfüllt. Dieser Status kann nur durch die automatische State Propagation erreicht werden.

NOT_FULFILLED: Für diese Sicherheitsanforderung liegt keine Bestätigung ihrer Erfüllung vor, die Bestätigung wurde explizit durch einen Stakeholder, die automatische State Propagation oder eine automatische Regel zurück genommen oder die zuletzt erteilte Bestätigung ist abgelaufen.

Die möglichen Übergänge zwischen diesen drei Stati und die jeweiligen Bedingungen sind in Abbildung 3 als UML State Machine Diagramm dargestellt. Die beiden oberen Übergänge *confirmation* und *revocation* beziehen sich sowohl auf die manuelle Bestätigung von Sicherheitsanforderungen durch Stakeholder als auch auf automatische Operationen, ausgelöst durch Modellveränderungen (siehe Abschnitt 2.4) oder bei Überschreiten der Gültigkeitsdauer von Bestätigungen. Auf die Angabe der vollständigen Bedingungen bei Statusänderungen auf Basis der State Propagation wurde aus Platzgründen in einigen Fällen verzichtet, diese können aber einfach aus den beiden unten angeführten Bedingungen abgeleitet werden.

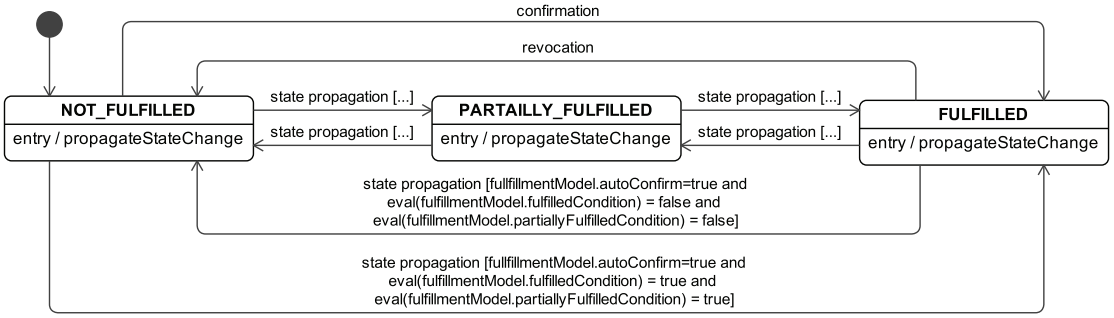

Abb. 3: UML State Machine Diagramm für Sicherheitsanforderugnen

2.3 Unterschiedliche Arten von Sicherheitsanforderungen

Unser Ansatz unterscheidet vier Arten von Sicherheitsanforderungen. Die *Basic Security Requirements* bilden die Grundbausteine für die meisten Modellinstanzen. Sie werden eingesetzt, um die grundlegende Struktur und Hierarchie der verwalteten Sicherheitsanforderungen abzubilden. Abgesehen von der vorgestellten State Propagation stellen diese Sicherheitsanforderungen keine weiteren Automatismen bereit und werden durch die beteiligten Stakeholder manuell verwaltet.

Message-based Security Requirements dienen der Integration von externen Stakeholdern über entsprechend strukturierte Kommunikationswege (z.B. per E-Mail oder einen gesonderten Web-Zugriff). Damit benötigen externe Stakeholder keinen direkten Zugriff auf das System und können trotzdem in den Prozess eingebunden werden. Message-based Security Requirements erlauben externen Stakeholdern beispielsweise, die Erfüllung der Ihnen zugewiesenen Sicherheitsanforderungen per E-Mail zu bestätigen. Das System stellt sicher, dass Bestätigungen rechtzeitig eingefordert werden und stößt ggf. einen Eskalationsmechanismus an.

Auto-check Security Requirements erlauben die einfache Einbindung weiterer Tools zur periodischen Überprüfung des Erfüllungsgrades einzelner Sicherheitsanforderungen. Der Grundgedanke besteht darin, die für Basic Security Requirements manuell veranlassten Statusänderungen basierend auf Aufrufen externer Tools zu automatisieren. Dies erlaubt die effiziente Einbindung bereits implementierter tool-getriebener Kontrollmechanismen. Einsatzmöglichkeiten sind beispielsweise die technische Überprüfung von Access Control Policies oder die Überwachung der Verfügbarkeit bestimmter Systemkomponenten.

Rule-based Security Requirements erlauben die automatische Reaktion auf Modellveränderungen durch Definition formaler Regeln. Diese Art von Sicherheitsanforderungen wird im nächsten Abschnitt vorgestellt.

2.4 Automatische Reaktion auf Modellveränderungen

Neben der automatischen State Propagation sieht das ADAMANT Framework ebenso die automatische Reaktion auf Änderungen angebundener Unternehmensmodelle mittels *Rule-based Security Requirements* vor. Dabei werden Änderungen an den importierten *ProtectionTargets* (siehe Abbildung 1) erkannt und durch ein Event-Condition-Action (ECA) System verarbeitet. Abbildung 4 zeigt ein UML Klassendiagramm mit vorgesehenen Klassen zur Definition dieser Regeln. Die wesentlichen Klassen und deren Assoziationen werden nachfolgend erläutert.

- Die Klasse *Rule* repräsentiert in unserem Modell eine automatische Regel, die bei Erfüllung einer Bedingung für eine definierbare Anzahl an Elementen eine festgelegte Folge von einzelnen Aktionen ausführt. Eine Regel ist mit einer Sicherheitsanforderung verknüpft und wird durch einen Trigger angestoßen.
- Die Klasse *Trigger* definiert die Bedingungen für die Auswertung von Regeln. Ein Trigger wird mit einem als Trigger-Target bezeichneten Element aus der Unternehmenslandschaft verknüpft und veranlasst die Ausführung von zugewiesenen Regeln. Es wird zwischen mehreren möglichen Trigger Events (z.B. neue Abhängigkeit, veränderte Attribute) unterschieden und eine zusätzliche Bedingung kann die Auslösung eines Triggers weiter einschränken.
- Das Interface *Action* dient der Definition unterschiedlicher Aktionen, die von Regeln verarbeitet werden. Die unterschiedlichen Aktionen erlauben die Bestätigung von Sicherheitsanforderungen sowie deren Zurücknahme, das Erstellen neuer Sicherheitsanforderungen oder das Löschen bestehender und die Benachrichtigung unterschiedlicher Stakeholder durch den Versand von Nachrichten oder die Zuweisung neuer Aufgaben.

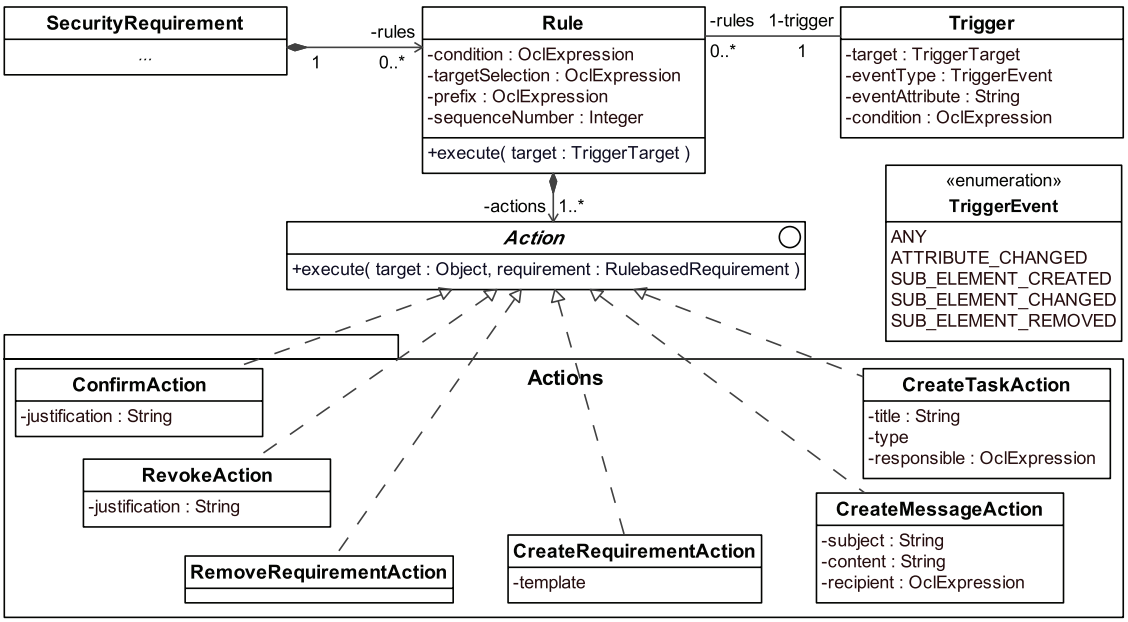

Abb. 4: UML Klassendiagramm für Rule-based Security Requirements

Durch den Einsatz von OCL zur Formulierung von Bedingungen und bei der Selektion der jeweiligen Ziel-Objekte für Aktionen können komplexe Abläufe automatisiert werden.

ADAMANT ist beispielsweise in der Lage, automatisch neue Sicherheitsanforderungen für neue Geschäftsprozesse zu erstellen, verantwortliche Stakeholder zuzuweisen und damit zeitnah die Einhaltung von Policies für eingesetzte IT Services zu garantieren. Eine weitere Möglichkeit für den Einsatz von Regeln ist die automatische Überprüfung von Constraints auf Unternehmensmodelle um beispielsweise zu überprüfen, dass nur entsprechend abgesicherte IT Services auf sensible Daten zugreifen.

2.5 Vorgehensmodell

Das allgemeine ADAMANT Vorgehensmodell unterscheidet zwischen einer dezidierten Einführungsphase und dem operativen Management der Sicherheitsanforderungen.

Die Einführungsphase erfolgt in zwei Schritten. Zunächst wird mit entsprechender Beteiligung des Top-Managements und in Abstimmung mit dem Risikomanagement der Umfang der zu betrachtenden Sicherheitsanforderungen bestimmt. Das Ergebnis ist eine vollständige Aufschlüsselung aller relevanten Top-Level Sicherheitsanforderungen, Standards, gesetzlicher Rahmenbedingungen und Risiken, die im Rahmen von ADAMANT betrachtet werden sollen. Diese bilden in Folge die oberste Ebene von Sicherheitsanforderungen, die im Tool erfasst werden. Ausgehend von diesen Zielen werden die benötigten Artefakte aus dem Unternehmenskontext erhoben und mögliche Informationsquellen (z.B.: EAM Tools, CMDBs) zur Einbindung in das Tool bestimmt. Abschließend werden die Schnittstellen zu diesen Dokumentationstools konfiguriert und die Definition der Top-Level Sicherheitsanforderungen im ADAMANT Tool abgeschlossen. Am Ende der Einführungsphase sind die obersten Ebenen des Sicherheitsmodells definiert und mit den jeweils betrachteten Artefakten aus der Unternehmenslandschaft verknüpft.

Das operative Management vom Sicherheitsanforderungen folgt prinzipiell einem kontinuierlichem Verbesserungsprozess wie in ISO 27001 [ISO13] und anderen ISM Standards empfohlen. Durch Analyse der verknüpften Artefakte aus der Unternehmenslandschaft und unter Zuhilfenahme der von ADAMANT bereit gestellten Berichte kann die Notwendigkeit für neue Sicherheitsanforderungen erkannt werden (z.B. bei neuen Geschäftsprozessen, IT Systemen, Infrastruktur-Komponenten oder veränderten Abhängigkeiten zwischen einzelnen Artefakten der Unternehmenslandschaft). Stakeholder können zudem die ihnen zugewiesenen Anforderungen durch Definition neuer untergeordneter Sicherheitsanforderungen weiter detaillieren und entlang der jeweiligen Hierarchie im Unternehmen die Verantwortlichkeiten für diese Unteranforderungen festlegen. Damit wird das Sicherheitsmodell kooperativ mit Beteiligung der jeweils domänenverantwortlichen Stakeholder erweitert und stetig konkretisiert. Dabei erfolgt die eigentliche Bewertung von Sicherheitsanforderungen unmittelbar und nichterfüllte Anforderungen werden durch die State Propagation, automatische Workflows und die Reporting-Möglichkeiten an die richtigen Stakeholder kommuniziert. Diese können damit zeitnah auf Missstände reagieren und die Einhaltung der vorgegeben Sicherheitsziele sicher stellen.

3 Der adamant Prototyp

Zur Umsetzung unseres Frameworks wurde ein innovativer Prototyp zur kooperativen Verwaltung von kontextuellen Sicherheitsanforderungen in Unternehmen entwickelt. Der adamant Prototyp implementiert die in Abschnitt 2 vorgestellten Modelle und Konzepte. Aufgrund der angestrebten Beteiligung von unterschiedlichen Stakeholder-Gruppen aus unterschiedlichen Domänen wurde großes Augenmerk auf klare Benutzerführung und leicht verständliche Visua-

lisierungen für Sicherheitsanforderungen und deren Kontextinformationen gelegt. Abbildung 5 zeigt die hierarchische Darstellung von Sicherheitsanforderungen in unserem Tool, Abbildung 6 zeigt die kontextuelle Darstellung einer Sicherheitsanforderung gemeinsam mit den Abhängigkeiten zu und zwischen Artefakten der Unternehmenslandschaft.

Abb. 5: Screenshot des adamant Prototypen

Das adamant Tool stellt die nachfolgend aufgelisteten Hauptfunktionen zur Verfügung.

- *Systematische Dokumentation von Security Requirements während ihres gesamten Lebenszyklus*: Die Einhaltung von Sicherheitsanforderungen sowie Gründe für deren Nichterfüllung werden nachvollziehbar und können transparent kommuniziert werden. Die ganzheitliche Betrachtung im Rahmen des jeweiligen Unternehmenskontexts erhöht die Qualität der Sicherheitsbetrachtung und ermöglicht effizientere Audits.
- *Stakeholderzentrierte Berichte und Views*: In Abstimmung mit Sicherheitsmanagern und Auditoren wurden bedarfsgerechte Abfragen, Visualisierungen und Berichte umgesetzt. Diese unterstützen eine zielgerichtete Handlungsweise und erleichtern Audits und Zertifizierungsprozesse.
- *Regelbasierte Workflow-Unterstützung*: Automatische Regeln und die State Propagation sowie ein darauf abgestimmtes Workflow-System erlauben die effiziente Gestaltung von standardisierten Arbeitsabläufen im Bereich des IT Sicherheitsmanagements. Damit können Risiken minimiert werden und Unternehmen werden in die Lage versetzt, unmittelbar auf Missstände und nicht erfüllte Sicherheitsanforderungen zu reagieren.
- *Template-System*: Zur Unterstützung bei der Umsetzung von Standards und anderer wiederkehrender Sicherheitsanforderungen wurde ein umfassendes Template-System entwickelt.
- *Automatische Konflikt-Erkennung*: adamant ist in der Lage, Konflikte zwischen bestehenden Sicherheitsanforderungen zu identifizieren und verfügt über einen dezidierten Workflow zu deren Auflösung. Die Qualität und Korrektheit von Sicherheitsanforderungen kann dadurch nachhaltig verbessert werden.
- *Import von Modellen aus beliebigen IT Dokumentations-Tools*: Offene Schnittstellen erlauben die Einbindung beliebiger Dokumentationstools als Datenquelle für die Unternehmenslandschaft. Es werden keine zusätzlichen Abhängigkeiten oder Anforderungen an die Geschäftsprozess- oder IT- Dokumentation gestellt.

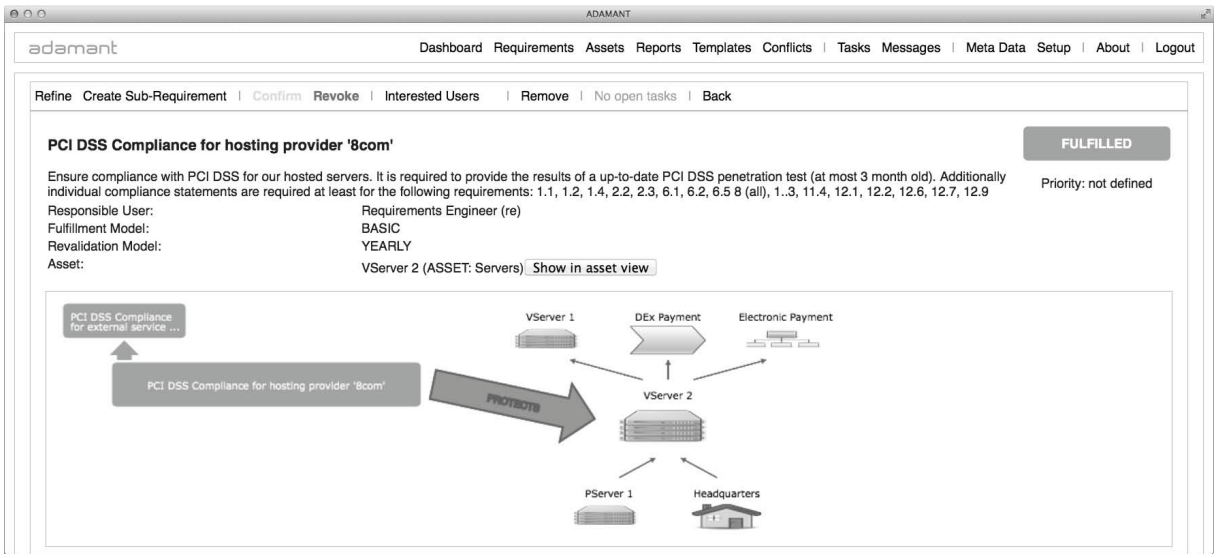

Abb. 6: Kontextuelle Darstellung einer Sicherheitsanforderung in adamant

Der adamant Prototyp wurde als Open Source Java Web-Anwendung realisiert und unter der Eclipse Public License (EPL) veröffentlicht. Ein Plugin-System ermöglicht die einfache Erweiterung von Kernfunktionen, um beispielsweise weitere Schnittstellen zu Dokumentationstools anzubinden, Workflows zu integrieren oder die State Propagation auf die jeweiligen Bedürfnisse in Unternehmen anzupassen. Weitere Informationen sowie der Quellcode und eine Live-Demo stehen auf der offiziellen adamant Homepage unter http://adamant.q-e.at zur Verfügung.

4 Vergleich mit anderen Ansätzen

Ähnliche Ansätze für die kooperative Verwaltung von kontext-bezogenen Sicherheitsanforderungen finden sich in den Bereichen des IT Risikomanagements, des Security Engineering sowie im Umfeld von Governance, Risk Management und Compliance (GRC). In diesen Bereichen wird die Integration mit Modellierungsinitiaitven forciert, um die Qualität der IT Sicherheitsmanagement Prozesse zu verbessern. Kooperative Vorgehensmodelle wie in [AgAl12, NtPP11, BrFG10, PNGD+13] sind für bestimmte Domänen oder Systembereiche optimiert und eignen sich nur bedingt zur unternehmensweiten Betrachtung von Sicherheitsanforderungen über Domänengrenzen hinaus. Die verfügbaren Tools im Bereich IT Sicherheitsmanagement und GRC sind in vielen Fällen auf die Unterstützung eines spezifischen Standards oder Vorgehensmodells beschränkt oder erlauben die modulare Einbindung einzelner Standards. Die systematische Integration mit Modellen aus den Bereichen der Unternehmensarchitektur und die automatische Reaktion auf Veränderungen in diesen Modellen wird nicht fokussiert und ist im Gegensatz zu ADAMANT auf bestimmte Modellierungswerkzeuge beschränkt.

Das ADAMANT Framework ist die konsequente Umsetzung des Living Models Paradigmas [BAFF+11], zugeschnitten auf den Bereich des IT Sicherheitsmanagements im komplexen unternehmerischen Umfeld. Ein ähnlicher Ansatz wurden in [InHB11] für Service-orientierte Systeme und Architekturen beschrieben. In [InBr06] wird der Einsatz von Modellen aus der Unternehmensarchitektur zur Verbesserung des IT Risiko Managements präsentiert. Das in diesem Beitrag vorgestellte Framework kondensiert die genannten Ansätze durch Modelle für die Unterstützung etablierter Standards, sowohl des IT Sicherheitsmanagements als auch der Un-

ternehmensarchitektur, und adressiert konkrete Herausforderungen im Zusammenhang mit der kooperativen Verwaltung von Sicherheitsanforderungen, deren systematischen Dokumentation und laufenden Auswertung.

5 Zusammenfassung und Ausblick

In diesem Beitrag haben wir ein innovatives Framework zur modellunterstützten, kooperativen Verwaltung von Sicherheitsanforderungen vorgestellt. ADAMANT folgt einem Top-Down Ansatz für die Ableitung immer konkreterer Sicherheitsanforderungen, die entlang vorhandener Unternehmensmodelle kontextuell definiert werden. Aus abstrakten Sicherheitsanforderungen für Geschäftsprozesse werden durch Einbindung der jeweils domänenspezifischen Stakeholder konkrete Anforderungen für die technische Umsetzung der benötigten IT Services und Systemkomponenten ermittelt. Das vorgestellte ADAMANT Framework ermöglicht dabei die systematische Unterstützung bei der fortlaufenden Sicherstellung der betrieblichen Sicherheitsziele, selbst bei sich stetig verändernden Rahmenbedingungen und Systemlandschaften.

Der präsentierte Prototyp erlaubt die Evaluierung unseres Ansatzes im Rahmen von Fallstudien und wurde im Rahmen des EU FP7-Projektes PoSecCo bei einem Service-Provider und einem Auditor eingesetzt. Die nächsten Schritten bestehen in der Durchführung weiterer Fallstudien und die Erweiterung des Frameworks für eine bessere Integration mit Werkzeugen aus dem Bereich des IT Risikomanagements. Die Zusammenführung mit Anforderungen aus anderen Bereichen in Richtung GRC sowie die Integration von Standards aus angrenzenden Domänen (z.B. für medizinische Systeme, Car-to-X-Systeme, Cyber-Physical Systems) wird aktiv verfolgt.

Danksagung

Diese Arbeit wurde durch die Tiroler Wirtschaftsförderung über die “Stiftungsassistenz QE-Lab” und zum Teil durch die Europäische Union über das FP7 Projekt “PoSecCo” (IST 257129) gefördert.

Literatur

[AgAl12] I. Aguirre, S. Alonso: Improving the Automation of Security Information Management: A Collaborative Approach. In: *IEEE Security & Privacy*, 10, 1 (2012), 55–59.

[BAFF+11] R. Breu, B. Agreiter, M. Farwick, M. Felderer, M. Hafner, F. Innerhofer-Oberperfler: Living Models - Ten Principles for Change-Driven Software Engineering. In: *Int. J. Software and Informatics*, 5, 1-2 (2011), 267–290.

[BaFo10] Y. Barlette, V. V. Fomin: The Adoption of Information Security Management Standards. In: *Information Resources Management: Concepts, Methodologies, Tools, and Applications* (2010), 69.

[BrFG10] C. Broser, C. Fritsch, O. Gmelch: Towards Information Security Management in Collaborative Networks. *In: 2010 21st International Conference on Database and Expert Systems Applications (DEXA)*, IEEE (2010), 359–363.

[BSI05] BSI (Bundesamt für Sicherheit in der Informationstechnik): IT Baseline Protection Catalogues (2005).

[HMPR04] A. R. Hevner, S. T. March, J. Park, S. Ram: Design science in Information Systems research. In: *MIS Quarterly*, 28, 1 (2004), 75–105.

[InBr06] F. Innerhofer-Oberperfler, R. Breu: Using an Enterprise Architecture for IT Risk Management. *In: ISSA'06: Proc. Information Security South Africa Conference, South Africa* (2006).

[InHB11] F. Innerhofer-Oberperfler, M. Hafner, R. Breu: Living security collaborative security management in a changing world. In: *IASTED International Conference on Software Engineering* (2011).

[ISO13] ISO (Int.l Organization for Standardization): ISO/IEC 27001: Information technology – Security techniques – Information security management system – Requirements. ISO/IEC (2013).

[IT 07] IT Governance Institute: COBIT v4.1. IT Governance Institute (2007).

[Long12] J. O. Long: ITIL® 2011 at a Glance. In: (2012).

[NtPP11] T. Ntouskas, G. Pentafronimos, S. Papastergiou: STORM: collaborative security management environment. *In: WISTP'11: Proceedings of the 5th IFIP WG 11.2 international conference on Information security theory and practice: security and privacy of mobile devices in wireless communication*, Springer-Verlag (2011).

[OMG12] OMG (Object Management Group): Object Constraint Language (OCL), V2.3.1. Object Management Group (2012).

[PCI10] PCI Security Standards Council LLC: Payment Card Industry Data Security Standard Version 2.0. PCI Security Standards Council LLC (2010).

[PNGD+13] D. Polemi, T. Ntouskas, E. Georgakakis, C. Douligeris, M. Theoharidou, D. Gritzalis: S-Port: Collaborative security management of Port Information systems. *In: Information, Intelligence, Systems and Applications (IISA), 2013 Fourth International Conference on*, IEEE (2013), 1–6.

[Sipo06] M. Siponen: Information security standards focus on the existence of process, not its content. In: *Communications of the ACM*, 49, 8 (2006), 97–100.

[SiWi09] M. Siponen, R. Willison: Information security management standards: Problems and solutions. In: *Information & Management*, 46, 5 (2009), 267–270.

[SpBa10] J. L. Spears, H. Barki: User Participation in Information Systems Security Risk Management. In: *MIS Quarterly* (2010).

[SPOA+12] M. Salehie, L. Pasquale, I. Omoronyia, R. Ali, B. Nuseibeh: Requirements-driven adaptive security: Protecting variable assets at runtime. In: *2012 IEEE 20th International Requirements Engineering Conference (RE)* (2012), 111–120.

[TBDM12] S. Thalmann, D. Bachlechner, L. Demetz, R. Maier: Challenges in Cross-Organizational Security Management. *In: System Science (HICSS), 2012 45th Hawaii International Conference on*, IEEE (2012), 5480–5489.