ProSA-Vorgehensmodell zur prozessorientierten IT-Sicherheitsanalyse

Daniela Simić-Draws

Universität Koblenz-Landau
Fachbereich Informatik
dsimic@uni-koblenz.de

Zusammenfassung

IT-Sicherheitsanalysen werden im Regelfall so durchgeführt, dass sich diese auf ausgewählte technische Komponenten beziehen. Handelt es sich bei der Einsatzumgebung um kleine Unternehmen mit einer überschaubaren IT-Landschaft, dann ist die Durchführung einer IT-Sicherheitsanalyse mit einem vertretbaren Aufwand möglich. Zur Herausforderung gerät eine IT-Sicherheitsanalyse jedoch dann, wenn die Größe der zugrunde liegenden Organisation und somit die Komplexität der dort eingesetzten IT-Infrastruktur zunehmen. Hierbei besteht die Gefahr, dass nicht offensichtlich sicherheitskritische Systembestandteile nicht erkannt und von der Analyse ausgenommen werden. Darüber hinaus haben die technisch fokussierten Analysemethoden den Nachteil, dass der Faktor Mensch in dessen unterschiedlichen Ausprägungen bei der Untersuchung nicht oder nur unzureichend berücksichtigt wird. Schließlich bleibt hierbei auf die Dynamik sicherheitskritischer IT-Systeme und ihrer Sicherheitsvorfälle verborgen, solange die Analyse lediglich Systemzustände und nicht die zugrunde liegenden Prozesse zum Ausdruck bringt. Die in diesem Beitrag vorgestellte Vorgehensweise soll einen Vorschlag liefern, wie den dargelegten Aspekten begegnet werden und somit eine möglich ganzheitliche IT-Sicherheitsanalyse durchgeführt werden kann.

1 Einleitung

IT-Sicherheit ist grundlegend als eine Eigenschaft zu verstehen, die sich auf die Zustände und Zustandsübergangsregeln informationstechnischer Systeme bezieht [GSB+14]. Eine ausschließliche Fokussierung auf die Technik ist hierbei jedoch nicht ausreichend. Vielmehr muss hierbei die Anwendungsumgebung mit berücksichtigt werden, da die Anforderung nach IT-Sicherheit vom menschlichen Anwender ausgeht und darüber hinaus von diesem interpretiert wird. In welchem Umfang die Anwendungsumgebung bei einer Sicherheitsbetrachtung berücksichtigt wird, ist von dem konkreten gewählten Analyseverfahren abhängig: Eine organisationsumfassende Untersuchung bietet beispielsweise das Vorgehen nach IT-Grundschutz [BSI11]. Hierbei wird gewissermaßen das „Big Picture“ der Sicherheitslage einer kompletten Organisation geliefert; dies kommt auch der Intention des IT-Grundschutzes entgegen, der eine Art Basis-Sicherheit anstrebt. Die formale Analyse eines spezifischen IT-Produkts gemäß Common Criteria [Comm12] zeichnet hingegen ein völlig anderes Bild von IT-Sicherheit. Hier ist nicht eine möglichst umfassende Betrachtung das Ziel, sondern die detaillierte Überprüfung technischer Systeme wird hier angestrebt. Andere Vorgehensweisen wie z.B. KORA [HPR93] oder Security Awareness Kampagnen [Heli09] bilden ebenfalls nur Teilbereiche des Komplexes IT-Sicherheit ab. Es fehlt eine Perspektive der IT-Sicherheit, welche von den genannten

P. Schartner (Hrsg.) · D•A•CH Security 2016 · syssec (2016) 13-23.

Teilbereichen Organisation, Technik, Recht und Anwender eine systematische und nachvollziehbare Abgrenzung vorzunehmen vermag. Darüber hinaus fehlen den genannten Verfahren Aspekte wie z.B. die Berücksichtigung der Dynamik und der differenzierte Einbezug des menschlichen Akteurs. Als ein Vorschlag dient hier die prozessorientierte Sichtweise auf die IT-Sicherheit.

2 Verwandte Arbeiten

Sicherheitsanforderungen können mit Hilfe einer Vielzahl unterschiedlicher Methoden spezifiziert werden: Eine international anerkannte und etablierte Methode zur Spezifikation von Sicherheitsanforderungen an IT-Produkte und die Evaluierung dieser IT-Produkte in Hinblick auf die spezifizierten Sicherheitsanforderungen sind die Common Criteria[1] [Comm12]. Diese bieten einen umfassenden Katalog an standardisierten funktionalen Sicherheitsanforderungen, welche von einem spezifischen IT-Produkt oder einer generischen Produktgruppe gefordert werden. Die Common Criteria können daher sowohl im Produktentstehungs- als auch im eigentlichen Produktlebenszyklus angewendet werden. Ein Vorgehen, welches speziell für Softwareentwicklungs-Projekte konzeptioniert wurde, ist SQuaRE (Security Quality Requirements Engineering) nach [MHS05]. Diese Methode zielt darauf ab, in neun Schritten kategorisierte und priorisierte Sicherheitsanforderungen zu spezifizieren. Ebenfalls eine Anwendung im frühen Stadium der Software-Entwicklung sieht CLASP (Comprehensive Lightweight Application Security Process) nach [Vieg05] vor. CLASP liefert eine Systematik in Form von Prozessbausteinen, welche in den regulären Software-Entwicklungs- bzw. Verbesserungsprozess integriert werden können. Dabei werden basierend auf Rollen, Ressourcen und erwarteten Interaktionen korrespondierende Sicherheitsanforderungen abgeleitet. Eine Vorgehensweise, mit der rechtliche Anforderungen in technische Implementierungsvorschläge konkretisiert werden, liefert [HPR93] mit KORA (Konkretisierung rechtlicher Anforderungen). Technische Sicherheitsanforderungen können aufgrund der Zusammenführung juristischen und informatischen Fachwissens systematisch und für beide Seiten nachvollziehbar von den abstrakten rechtlichen Vorgaben abgeleitet werden. IT-Sicherheitsanforderungen werden bei KORA jedoch nur dann gewürdigt, wenn sie rechtlich erforderlich sind. Eine ganzheitliche Sicht auf die IT-Sicherheit nimmt [SNK+13] vor, indem sowohl KORA als auch die Common Criteria und die BSI-Grundschutzkataloge [BSI11] ineinander integriert werden.

Eine Integration der Prozess- mit der Sicherheitsperspektive wird bereits in einigen Arbeiten thematisiert: Das SLP-Konzept nach [MBH14] basiert auf etablierten Standards und beschreibt deren Zusammenspiel, um sichere Logistikprozesse entwerfen zu können. Die Erweiterung bestehender Notationen zur Geschäftsprozessmodellierung wird von einigen Arbeiten mit jeweils unterschiedlichen Schwerpunkten thematisiert: eine Menge festgelegter Sicherheitsanforderungen wird in Form grafischer Elemente repräsentiert, mit denen bestehende Prozessmodelle annotiert werden wie z.B. in [DzKa06], [CZM+ 11] und [PGP+12]. Eine syntaktische Erweiterung der BPMN, indem neue Notationselemente spezifiziert werden, nehmen z.B. [RFP07a], [RFP07b] und [LMV+05] vor. Das Versehen bestehender Notationselemente mit einer neuen Bedeutung im Sinne einer semantischen Erweiterung nehmen z.B. [AMA12] vor.

[1] Die drei Teile der Common Criteria wurden als DIN ISO/IEC 15408-1...3 veröffentlicht.

3 Das ProSA-Vorgehensmodell

Indem auf Geschäftsprozesse als Untersuchungsgegenstand zurückgegriffen wird, ergeben sich die Chancen in Richtung einer ganzheitlichen IT-Sicherheitsanalyse (z.B. nach [SNK+13]). Die zu untersuchenden IT-Systeme werden hierbei nicht mehr willkürlich aus einem organisatorischen Verbund herausgegriffen, sondern ergeben sich aus den für die Analyse ausgewählten primären – d.h. wertschöpfenden – Geschäftsprozessen. Da die somit betrachteten Geschäftsprozesse als existenzsichernd einzuordnen sind, ist auch die unterstützende IT einer bevorzugten Betrachtung zu unterziehen. Auf diese Weise ist grundlegend eine sinnvolle und nachvollziehbare Abgrenzung der zu betrachtenden IT-Systeme gegeben. Schwachstellen sind jedoch nicht nur in den verwendeten IT-Systemen zu finden, sondern lassen sich darüber hinaus beispielsweise auch in der Gestalt des zugrunde liegenden Geschäftsprozesses oder insbesondere auch auf Anwenderseite finden. Hinsichtlich der Prozessgestaltung kann eine Optimierung zunächst einmal in Richtung Effizienzsteigerung vorgenommen werden. Diese ist insbesondere durch das Eliminieren oder der parallelen Durchführung von Aufgaben möglich. Dadurch wird die Prozesslaufzeit verkürzt sowie Ressourcen gespart. Darüber hinaus sind noch weitere Gestaltungsmaßnahmen möglich, welche sich effizienzverbessernd auf Prozesse auswirken können. Solche Eingriffe in einen Geschäftsprozess bringen jedoch die Gefahr mit sich, dass sicherheitskritische Schwachstellen entstehen. Dies ist beispielsweise dann der Fall, wenn aufwendige Prüfschritte zu Gunsten einer Prozesslaufzeitverkürzung eliminiert werden. Ob und welche Gestaltungsmaßnahmen sich letztlich in einem sichereren oder geschwächten Prozess niederschlagen, kann pauschal nicht gesagt werden, sondern ist abhängig von dem spezifischen Anwendungsfall.

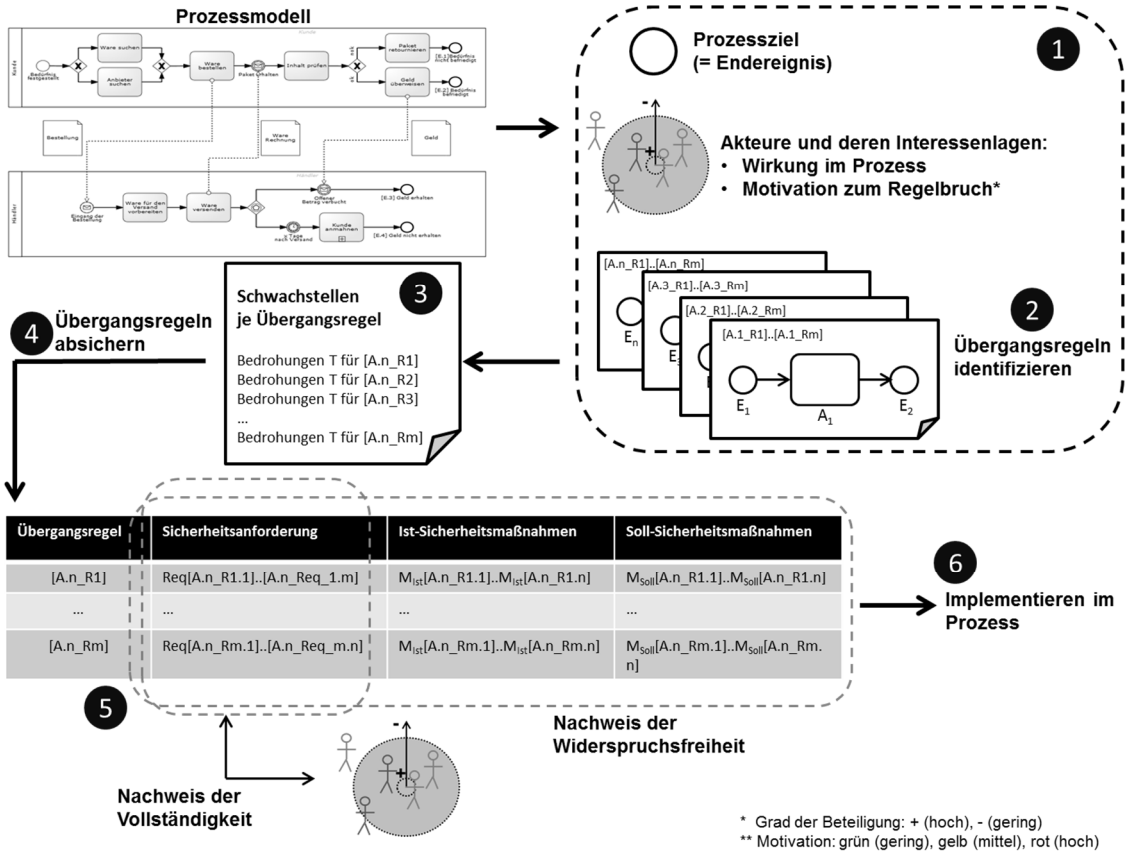

Abb. 1: Die sechs Schritte des ProSA-Vorgehensmodells

Die Vorgehensweise zur Durchführung einer prozessorientierten IT-Sicherheitsanalyse nach [Simi16] die nachfolgend vorgestellt wird, stellt eine neue mögliche Lösung für die zuvor aufgeführten Probleme dar. Ausgehend von Geschäftsprozessen erlaubt sie die Durchführung einer IT-Sicherheitsanalyse und damit die Herleitung von IT-Sicherheitsanforderungen auf Basis von Geschäftsprozessmodellen. Die Besonderheit besteht vor allem darin, dass die IT-Sicherheitsanalyse unter expliziter Berücksichtigung der Interessenslagen der Anwender erfolgt. Dadurch kann ermittelt werden, ob und in wie weit der korrekte – d.h. regelkonforme – Prozessverlauf durch möglicherweise entgegen gesetzter Interessen bedroht sein kann. Hierzu werden die Rollen Sicherheitsrisiko, Sicherheitsträger und Wissensträger definiert, die der am Prozess beteiligte Anwender einnehmen kann: Als Sicherheitsrisiko nutzt er ihm bekannte Schwachstellen aus, in der Rolle des Sicherheitsträgers hingegen agiert er als Sicherheitsmaßnahme (z.B. bei der Anwendung des Mehr-Augen-Prinzips). Als Wissensträger wird der Anwender ebenfalls in die Untersuchung involviert, indem dessen implizites Anwendungswissen ebenfalls der IT-Sicherheitsanalyse zugeführt wird. Die ProSA-Vorgehensweise (siehe Abbildung 1) gliedert sich in die folgenden sechs Schritte:

3.1 Prozessziel und Interessenlage identifizieren

Für den ersten Schritt sollte das zu untersuchende Geschäftsprozessmodell vorliegen, da dieses die Ausgangsbasis für die IT-Sicherheitsanalyse liefert. Das durch den Prozess angestrebte Ziel wird als ein schützenswertes Gut nach [GSB+14] aufgefasst. Dieses wird erreicht, wenn die an einem Prozess beteiligten Akteure miteinander kooperieren. Hierfür führen sie eine Menge von Aktivitäten aus, welche sich mit Hilfe möglichst atomarer so genannter Übergangsregeln detailliert beschreiben lassen. Solange die innerhalb der Aktivitäten spezifizierten Übergangsregeln von den Akteuren eingehalten werden, ist ein regelkonformer und zielgerichteter Verlauf des Prozesses gegeben. Wird dagegen eine Übergangsregel durch einen Akteur unterlaufen – z.B. zur Durchsetzung seiner eigenen Interessen – kann der Prozess als gefährdet bezeichnet werden. Nicht regelkonformes Verhalten eines Akteurs kann also dazu führen, dass das Prozessziel verfehlt wird. Das Ziel der Analyse besteht in den weiteren Schritten der hier beschriebenen Vorgehensweise also darin, solche Schwachstellen aufzudecken, welche eine Regelverletzung erlauben.

Hierzu wird zunächst anhand des zu untersuchenden Prozessmodells das formal spezifizierte Prozessziel, die am Prozess beteiligten Anwender sowie deren Interessen ermittelt. Letzteres ist von Bedeutung, da sich unterschiedliche Interessen in Kombination mit Schwachstellen zu einer Bedrohung für das Prozessziel und somit – bei wertschöpfenden Prozessen – auch für die übergeordnete Organisation auswirken können.

Um zu einer Einschätzung darüber zu gelangen, von welchen Akteuren eine hohe Gefährdung ausgeht, wird ein einfaches Maß eingeführt (siehe Abbildung 2): Für jeden am Prozess beteiligten Akteur wird dessen Beteiligung am Prozess (1. Faktor) sowie dessen Motivation zum Regelbruch (2. Faktor) ermittelt. Das hierbei entstehende Produkt ist ähnlich zu dem, wie es auch im Rahmen des Risikobegriffs verstanden wird (hier: das Risiko bildet sich aus dem Produkt von Eintrittswahrscheinlichkeit und Schadenshöhe). Das vorgeschlagene Maß sorgt hierbei für eine Profilierung der Bedrohungslage und dient als Grundlage für die Entscheidungen bezüglich der notwendigen IT-Sicherheitsmaßnahmen. In diesem Schritt erfolgt also eine explizite Berücksichtigung der Akteure als Sicherheitsrisiko.

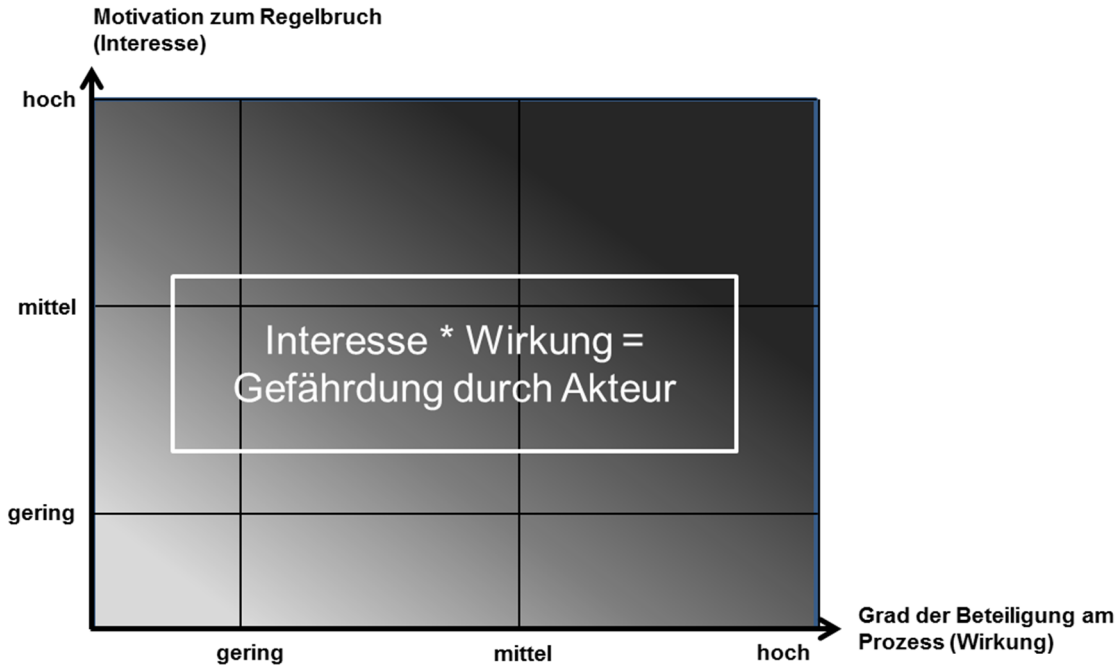

Abb. 2: Die Gefährdung durch den Akteur ist abhängig von dessen Interesse und Beteiligung

Die erlangten Ergebnisse nach Durchführung von Schritt 1 lauten:

- Prozessziel
- Am Prozess beteiligte Akteure
- Einschätzung der Gefährdung durch den jeweiligen Akteur

3.2 Übergangsregeln je Aktivität beschreiben

Zwischen dem Prozess initiierenden Ereignis und dem Endereignis befinden sich inhaltlich abgeschlossene Aktivitäten. Für jede dieser Aktivitäten können implizite Handlungsanweisungen – die so genannten Übergangsregeln – ermittelt werden. Die Übergangsregeln und somit die Aktivität werden von einem oder mehreren Akteuren – gegebenenfalls auch IT-gestützt – ausgeführt. Das Gerüst eines Prozesses wird jedoch nicht durch die Aktivitäten gebildet, sondern von den Ereignissen. Diese steuern die Prozessausführung, indem sie nicht nur den gesamten Prozess, sondern auch jede einzelne Aktivität starten bzw. als deren Ergebnis generiert werden. Diese Zwischenereignisse sind aufgrund von Komplexitätsgründen oftmals nicht explizit im Prozessmodell dargestellt. Sie sind jedoch durchaus als wichtige Meilensteine zu verstehen, die bei der formalisierten Beschreibung der Aktivitäten bzw. der darin enthaltenen Übergangsregeln zu berücksichtigen sind.

Im zweiten Schritt werden also die dem untersuchten Prozess zugrunde liegenden Übergangsregeln identifiziert und wie in Abbildung 3 formalisiert dargestellt. Ein Geschäftsprozess besteht aus Ereignissen und Aktivitäten [SiHi14]. Übergangsregeln beschreiben, wir ausgehend von einem Ereignis E(n) mittels einer Aktivität A(n) ein Ereignis E(n+1) erreicht wird. Übergangsregeln sind einer Aktivität untergeordnet und umfassen eine Menge von Beschreibungen, wie Anwender und IT-Systeme miteinander interagieren müssen, um zu dem nachgelagerten Ereignis E(n+1) zu gelangen. Ereignisse werden hierbei also als Meilensteine angesehen, wel-

che zum Prozessziel führen sollen. Wie bereits dargelegt, werden Aktivitäten durch ein vorangehendes Ereignis angestoßen und führen zu geplanten – im Sinne von in den nicht explizit vorhandenen Übergangsregeln beschriebenen – Wechselwirkungen zwischen Anwendern und IT-Systemen bzw. sonstigen Ressourcen. Die Umsetzung der Übergangsregeln kann in unterschiedlich hohem Grad mittels informationstechnischer Systeme umgesetzt werden; eine Teilmenge dieser Regeln kann aber beispielsweise auch in organisatorischen Richtlinien formuliert sein. Auch hier ist es wieder von dem Anwendungsfall abhängig, ob Übergangsregeln explizit beschrieben bzw. darüber hinaus technisch unterstützt sind. Die Übergangsregeln werden unter Einbezug der am Prozess beteiligten Akteure – hier in ihrer Rolle als Wissensträger – sowie unter Zuhilfenahme bestehender Dokumentationen ermittelt.

```
1  ActivityID
2  ON Event(TimeEvent | MessageEvent | …)
3  IF Condition (Prämisse / Voraussetzung wird mittels Übergangsregeln
4  überprüft)

5       THEN DO Action (Konklusion durch Umsetzung einer Menge von
6       Übergangsregeln)
7       FOR Event (TimeEvent | MessageEvent | … das von der Aktivität
8       getriggert wird)

9       ELSE DO Alternative Action (Konklusion durch Umsetzung einer
10      alternativen Menge von Übergangsregeln)
11      FOR Alternative Event (Abbruchevent)
```

Abb. 3: Formalisierter Aufbau einer Aktivität als ECA^mA^n-Regel

Die erlangten Ergebnisse nach Durchführung von Schritt 2 lauten:

- Das die Aktivität initiierende Ereignis
- Von der Aktivität ausgelöste Ereignisse (Standard, Abbruchereignis)
- Übergangsregeln je Aktivität

3.3 Schwachstellen in den Übergangsregeln aufdecken

Die Übergangsregeln, die in dem vorherigen Schritt für jede Aktivität identifiziert worden sind, liefern einen detaillierten Einblick in den zu untersuchenden Prozess. Hierbei wird nicht nur expliziert, durch welches Ereignis eine jede Aktivität ausgelöst wird. Sondern es wird zudem die Wechselwirkung zwischen Akteuren und IT-Systemen sowie die erzeugten, verbrauchten oder veränderten Ressourcen sichtbar. Die Übergangsregeln beschreiben den planvollen Ablauf dieser Interaktion, weshalb diese in dem vorliegenden Schritt auf Schwachstellen untersucht werden sollen. Eine mögliche Kategorisierung der Schwachstellen lautet wie folgt:

- **Technische Schwachstellen:** fehlerhafte Implementierung, veraltete Algorithmen, etc.
- **Organisatorische Schwachstellen:** Nichteinhalten von Prüfvorgängen, Konsolidieren von Wissen, etc.
- **Rechtliche Schwachstellen:** „wo kein Kläger, da kein Richter“ – es kann angenommen werden, dass die Risikobereitschaft für nicht regelkonformes Verhalten steigt, wenn keine rechtlichen Konsequenzen zu erwarten sind
- **Anwenderbezogene Schwachstellen**: unbewusster Umgang mit IT-Systemen, mangelnde Awareness, etc.

- **Prozessbezogene Schwachstellen:** die Reihenfolge und der Aufbau der zu durchlaufenden Aktivitäten bzw. der zugrunde liegenden Übergangsregeln können sowohl sicherheitsbefördernde als auch sicherheitsreduzierende Auswirkungen haben

Der zentrale Schritt der Vorgehensweise besteht in dem vorliegenden dritten Schritt: Die Übergangsregeln, welche den korrekten Prozessverlauf und somit das Prozessziel sicherstellen sollen, sind Bedrohungen ausgesetzt. Die im Prozess eingesetzten IT-Systeme weisen Schwachstellen auf, die von den am Prozess beteiligten Anwendern zur Durchsetzung der eigenen Interessen ausgenutzt werden können. Eine explizite Bedrohung für eine Übergangsregel ist aber erst dann gegeben, wenn beides zusammenfällt: Ein Interessenskonflikt, der auf einem robusten IT-System ausgetragen wird, kann als nicht bedrohlich aufgefasst werden. Ebenso ist eine Schwachstelle, an deren Ausnutzung kein Interesse besteht, auch nicht wirklich gefährlich. Eine Bedrohung ist demnach also erst dann gegeben, wenn ein Interesse unberechtigt durchgesetzt werden kann, indem eine vorhandene technische Schwachstelle ausgenutzt wird. Die vorgeschlagene Vorgehensweise soll also bei der Beantwortung der Frage helfen, auf welche Art und Weise die einzelnen Übergangsregeln konkret verletzt werden können, da sich dort die einzelnen Schwachstellen auffinden lassen. Sobald dies ermittelt worden ist, können IT-Sicherheitsanforderungen zu deren Absicherung formuliert werden. Wichtig in diesem Zusammenhang ist, dass lediglich die Schwachstellen behoben werden können. Die Interessenkonflikte müssen hingegen als gegeben hingenommen werden.

Die erlangten Ergebnisse nach Durchführung von Schritt 3 lauten:

- Schwachstellen je Aktivität
- Bedrohungen je Aktivität als Ergebnis der Zusammenführung von Schwachstellen mit Interessenkonflikten

3.4 Übergangsregeln absichern

Das Zerlegen der einzelnen Aktivitäten eines Prozesses in möglichst atomare Übergangsregeln bietet den Vorteil, dass detaillierte Einblicke in potenzielle Sicherheitsprobleme ermöglicht werden. Indem Schwachstellen in den Übergangsregeln mit den Interessen der beteiligten Akteure zusammen geführt werden, lassen sich konkrete Bedrohungen identifizieren. Die ermittelten Bedrohungen beziehen sich nicht nur auf technische Systeme, sondern auch auf darüber hinausgehende Aspekte. Beispielsweise kann ein Akteur eine technische Schwachstelle – hier: unverschlüsselte Kommunikation sensibler Daten – ausnutzen, um die zwischen IT-Systemen übertragenen Nachrichten mitzulesen. Eine organisatorische Schwachstelle wird dagegen ausgenutzt, wenn ein Akteur bewusst in weiteren Aktivitäten partizipiert, durch die er in der Lage ist, Wissen unberechtigt zu konsolidieren. Auch kann der Akteur selbst als eine Schwachstelle angesehen werden; beispielsweise wenn er aufgrund mangelnden Sicherheitsbewusstseins vertrauliche Informationen an unberechtigte Dritte weitergibt. Um zu vermeiden, dass aus den möglichen Bedrohungen tatsächliche Angriffe wie z.B. Phishing, MITM-Attacken, etc. entstehen, müssen abstrakte IT-Sicherheitsanforderungen spezifiziert werden. Diese bezeichnen wünschenswerte Eigenschaften an ein IT-System und werden konkreten schützenswerten Gütern zugeordnet (z.B. „Integrität von gespeicherten Daten“).

Steht also fest, ob und in wie weit die einzelnen Übergangsregeln des untersuchten Prozesses Bedrohungen ausgesetzt sind, kann der vierte Schritt durchlaufen werden. Hier wird für jede untersuchte Übergangsregel eine Menge von relevanten IT-Sicherheitsanforderungen abgeleitet, wobei jede identifizierte Bedrohung durch mindestens eine Anforderung begegnet werden

soll. Die Implementierung der abstrakten IT-Sicherheitsanforderungen erfolgt über Maßnahmen, die nicht nur technischer (z.B. Verschlüsselung), sondern auch organisatorischer (z.B. Mehr-Augen-Prinzip), rechtlicher (z.B. Androhung von Strafe), anwenderbezogener (z.B. Schulungsmaßnahmen) oder auch prozessgestalterischer Natur – wie zuvor beschrieben – sein können.

Die erlangten Ergebnisse nach Durchführung von Schritt 4 lauten:

- Abstrakte IT-Sicherheitsanforderungen je Aktivität
- Bereits implementierte IT-Sicherheitsmaßnahmen je Aktivität
- Soll-Maßnahmen zur Abwehr der identifizierten Bedrohungen je Aktivität

3.5 Vollständigkeit und Widerspruchsfreiheit prüfen

Neben Sicherheitsproblemen lassen sich mittels einer Analyse der einzelnen Übergangsregeln auch bereits eingesetzte Sicherheitsmaßnahmen erkennen. Aus diesem Grund können die hergeleiteten Sicherheitsanforderungen dahingehend bewertet werden, ob sie durch Ist-Maßnahmen bereits erfüllt oder (noch) nicht erfüllt bzw. nicht erfüllbar sind. Als nicht erfüllbar werden solche Anforderungen bezeichnet, für die beispielsweise keine adäquate Sicherheitsmaßnahme existiert oder wenn die Aufwand-Nutzen-Relation ungünstig erscheint. Eine Sicherheitsanforderung ist aber auch dann als nicht erfüllbar einzustufen, wenn nicht auflösbare Inkonsistenzen zwischen einzelnen Sicherheitsanforderungen aufgedeckt werden. Dies ist beispielsweise dann der Fall, wenn eine gleichzeitig anonyme und nachvollziehbare Kommunikation gewünscht ist.

Sobald die Menge der IT-Sicherheitsanforderungen und möglicher Maßnahmen zu deren Umsetzung feststeht, müssen sie nicht nur hinsichtlich ihrer Widerspruchsfreiheit, sondern auch bezüglich ihrer Vollständigkeit überprüft werden. Die Vollständigkeit wird in mehreren Stufen nachgewiesen: Jede ermittelte Bedrohung muss durch mindestens eine abstrakte Sicherheitsanforderung adressiert werden. Für jede Sicherheitsanforderung muss mindestens eine Sicherheitsmaßnahme vorgeschlagen werden. Dieses Vorgehen existiert beispielsweise im Rahmen der Common Criteria und kann als durchaus etabliert angesehen werden. Für die Behandlung von Widersprüchen wurde im Rahmen der ProSA-Vorgehensweise ein neues Verfahren entwickelt, das hier eingeführt werden soll:

- Die Anforderungen jeder Aktivität werden untereinander auf Widersprüche untersucht; ohne Berücksichtigung des Bezugsobjekts (z.B. die Konstellation „Anonymität und Authentizität“ in einer Aktivität).
- Widersprüchliche Anforderungspaare innerhalb einer Aktivität werden dann unter Einbezug des Objekts untersucht. Beziehen sich die Anforderungen auf unterschiedliche Objekte, so kann ein scheinbarer Widerspruch durchaus aufgelöst werden. Beziehen sich inkonsistente Anforderungen hingegen auf das gleiche Objekt, muss der entstehende Widerspruch durch eine Anpassung der Anforderungen oder des Objektes aufgelöst werden.
- Sind die Anforderungen jeder Aktivität widerspruchsfrei, dann werden die einem Akteur zugewiesenen Aktivitäten auf die gleiche Art untersucht. Ein Auflösen von Widersprüchen zwischen Anforderungen zweier Aktivitäten einer Rolle ist z.B. durch Änderungen in der Ablaufmodellierung möglich.
- Zuletzt werden die Anforderungen von Aktivitäten in einen Zusammenhang gebracht, die bei der Kommunikation zwischen verschiedenen Akteuren miteinander interagieren.

- Anforderungsmengen von Aktivitäten verschiedener Akteure, zwischen denen kein Kommunikationsfluss stattfindet, werden nicht abgeglichen.
- Auf die gleiche Weise erfolgt die Überprüfung der Widerspruchsfreiheit der Anforderungen gegenüber den bereits vorhandenen Maßnahmen.

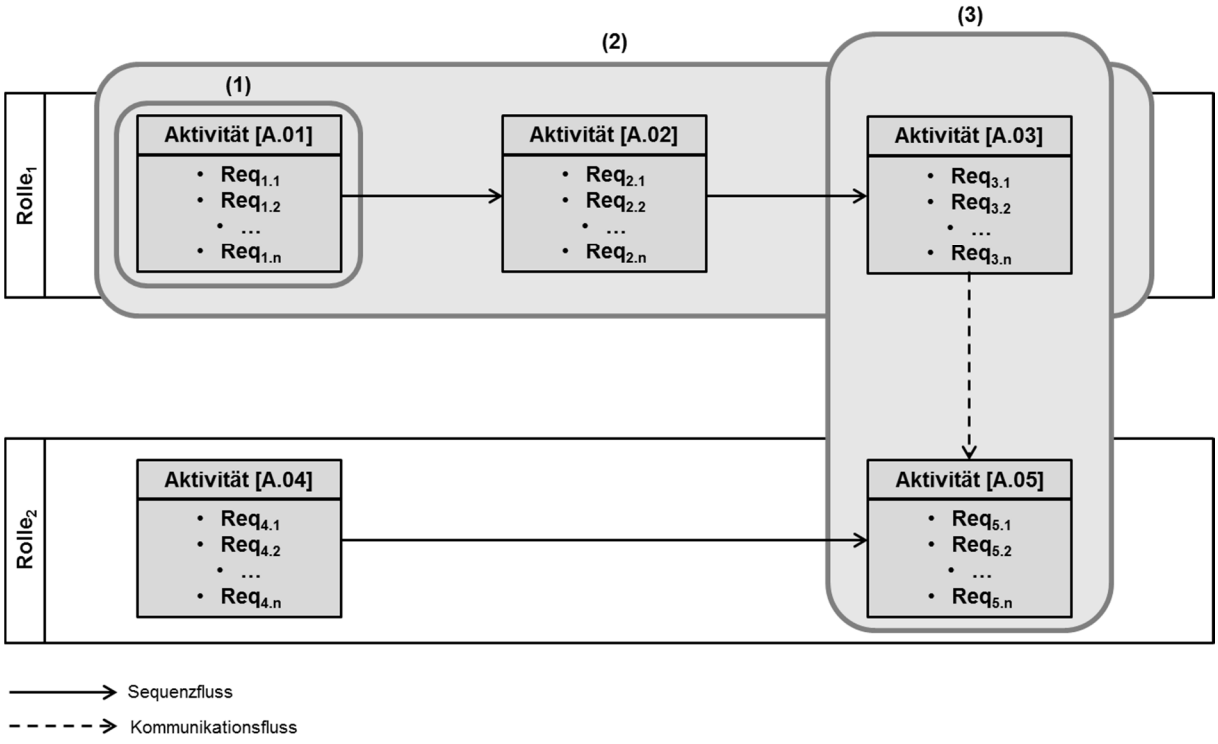

Abb. 4: Überprüfung der Widerspruchsfreiheit

Das beschriebene Vorgehen hat nicht nur den Vorteil, dass der entstehende Aufwand $< (n \cdot (n+1))/2$ ist. Sondern es werden zusätzlich sowohl die Reihenfolge als auch die Interaktion der einzelnen Aktivitäten des untersuchten Prozesses berücksichtigt. Beides – Reihenfolge und Interaktion – können sich sowohl sicherheitsbefördernd als auch sicherheitsmindernd auswirken.

Die erlangten Ergebnisse nach Durchführung von Schritt 5 lauten:

- Nachweis der Vollständigkeit der IT-Sicherheitsanforderungen gegenüber der auf den Interessenlagen basierenden Bedrohungen
- Nachweis der Widerspruchsfreiheit der IT-Sicherheitsanforderungen
  - untereinander
  - gegenüber bereits implementierter IT-Sicherheitsmaßnahmen

3.6 Umsetzung der ausgewählten Maßnahmen im Prozess

Sobald die Vollständigkeit und die Widerspruchsfreiheit insbesondere der abstrakten IT-Sicherheitsanforderungen festgestellt worden ist, können anhand derer die zu berücksichtigenden IT-Sicherheitsmaßnahmen konkretisiert werden. Diese sollen die Übergangsregeln so schützen, dass die Motivation zum Regelbruch bei den am Prozess beteiligten Akteuren reduziert wird. Die Identifikation von Maßnahmen, die für den spezifischen untersuchten Prozess als angemessen erscheinen, erfolgt beispielsweise unter Berücksichtigung des Return of

Security Investment (RoSI), deren Wartbarkeit, Usability, etc. Die ausgewählten Maßnahmen werden im Prozess implementiert und bewirken dort konkrete Sicherheitsoperationen (z.B. GPG kann als konkrete Sicherheitsoperation der Sicherheitsmaßnahme „Verschlüsselung“ angesehen werden).

Es ist hervorzuheben, dass die zuvor identifizierten Interessenkonflikte durch den Einsatz neuer IT-Sicherheitsmaßnahmen nicht behoben werden, sondern im Gegenteil bestehen bleiben. Die durch die Maßnahmen umgesetzten Sicherheitsanforderungen stellen jedoch die Einhaltung der Übergangsregeln und somit die Erreichung des Prozessziels sicher. Der mittels ProSA optimierte Prozess sollte ständig überwacht werden, damit zeitnah auf bekannt werdende neue Sicherheitsprobleme reagiert werden kann. Aus diesem Grund wird empfohlen, den optimierten Prozess gemäß des Prinzips der kontinuierlichen Verbesserung erneut der IT-Sicherheitsanalyse gemäß der ProSA-Vorgehensweise zuzuführen. Abschließend sei angemerkt, dass die einzelnen Schritte der hier vorgeschlagenen Sicherheitsanalyse nicht für sich abgeschlossen sind. Vielmehr sind Rückkopplungen erwünscht, die z.B. zu einer qualitativen Verbesserung der Ergebnisse führen.

Die erlangten Ergebnisse nach Durchführung von Schritt 6 lauten:

- Vorschläge an einzusetzenden IT-Sicherheitsmaßnahmen
- Sicherheitsoptimierter Prozess
- Durch Einbezug der Akteure in die Analyse hat im günstigen Falle deren Sensibilisierung stattgefunden

4 Fazit

Es existieren unterschiedliche Vorgehensweisen zur Durchführung einer IT-Sicherheitsanalyse, die jeweils unterschiedliche Sichtweisen fokussieren. Lücken bestehen bei den etablierten Verfahren insbesondere hinsichtlich der Berücksichtigung dynamischer Vorgänge, der differenzierte Einbezug des Menschen sowie eine nachvollziehbare und systematische Abgrenzung des Kontextes. Aus den dargelegten Gründen wird in diesem Beitrag eine Vorgehensweise – genannt ProSA – vorgestellt, die die Durchführung einer IT-Sicherheitsanalyse anhand von Prozessen erlaubt. Die Anwendbarkeit und der praktische Nutzen von ProSA konnte beispielsweise im Rahmen einer Kooperation mit der Stadt Koblenz nachgewiesen werden.

Literatur

[AMA12] O. Altuhova, R. Matulevičius, N. Ahmed: Towards Definition of Secure Business Processes. *LNBIR: Workshop on Information Systems Security Engineering,* 1-15, (2012).

[BSI11] Bundesamt für Sicherheit in der Informationstechnik: IT-Grundschutzkataloge. Bundesanzeiger-Verlag, Köln, 12. Ergänzungslieferung, (2011).

[Comm12] Common Criteria for Information Technology Security Evaluation: Version 3.1 (2012).

[CZM+11] I. Ciuciu, G. Zhao, J. Mülle, S. von Stackelberg, C. Vasquez, T. Haberecht, R. Meersman, K. Böhm: Semantic Support for Security-Annotated Business Process models. *LNBIP*, Vol. 81, 284-298, (2011).

[DzKa06] G. Dzhendova, D. Kalmring: Modellierung von IT-Sicherheit. Analyse und Synthese. *Proc. DACH Security*, (2006).

[GSB+14] R. Grimm, D. Simić-Draws, K. Bräunlich, A. Kasten, A. Meletiadou: Referenzmodell für ein Vorgehen bei der IT-Sicherheitsanalyse. Informatik Spektrum (2014).

[Heli09] M. Helisch: Security Awareness. Neue Wege zur erfolgreichen Mitarbeiter-Sensibilisierung. Vieweg+Teubner, Wiesbaden, (2009).

[HPR93] V. Hammer, U. Pordesch, A. Roßnagel: KORA. Eine Methode zur Konkretisierung rechtlicher Anforderungen zu technischen Gestaltungsvorschlägen für Informations- und Kommunikationssysteme. *Infotech 1/1993*, S. 21ff, (1993).

[LMV+05] J. Lopez, J.A. Montenegro, J.L. Vivas, E. Okamoto, E. Dawson: Specification and Design of Advanced Authentication and Authorization Services. *Computer Standards & Interfaces 27(5),* 467-478, (2005).

[MBJ14] M. Middelhoff, C. Böhle, B. Hellingrath: Modeling and Analyzing Information Security in Secure Logistics Business Processes. *Proc. MKWI 2014*, 1925-1926, (2014).

[MHS05] N.R. Mead, E.D. Hough, T.R. Stehney II: Security Quality Requirement Engineering (SQUARE) Methodology. Technical Report CMU/SEI-2005-TR-009. Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA, (2005).

[PGP+12] E. Paja, P. Giorgini, S. Paul, P.H. Meland: Security Requirements Engineering Support for Security-Annotated Business Processes. *LNBIP*, 77-89, (2012).

[RFP07a] A. Rodríguez, E. Fernández-Medina, M. Piattini: A BPMN Extension for the Modeling of Security Requirements in Business Processes. *Proc. IEICE Trans. Inf. & Syst., Vol. E90-D, No. 4,* (2007).

[RFP07b] A. Rodríguez, E. Fernández-Medina, M. Piattini: M-BPSec: A Method for Security Requirement Elicitation from a UML 2.0 Business Process Specification. *LNI Vol. 4802*, 106-115, (2007).

[Simi16] D. Simić-Draws: Prozessorientierte IT-Sicherheitsanalyse unter Berücksichtigung divergierender Interessenlagen der Prozessbeteiligten. Dissertation. Universität Koblenz – Landau. Fachbereich 4: Informatik (2016).

[SiHi14] C. Simon, B. Hientzsch: Prozesseigner. Wissen & Methoden für Manager für Unternehmensprozesse. Wiesbaden: Springer, (2014).

[SNK+13] D. Simić-Draws, S. Neumann, A. Kahlert, P. Richter, R. Grimm, M. Volkamer, A. Roßnagel: Holistic and Law Compatible IT Security Evaluation: Integration of Common Criteria, ISO 27001/IT-Grundschutz and KORA. In: *IJISP, 7(3),* 16-35, (2013).

[Vieg05] J. Viega: Building security requirements with CLASP. *Proc. Workshop on software engineering for secure systems (SESS)*, 1-7, (2005).