TRENNUNG OPERATIVER UND KONTROLLIERENDER FUNKTIONEN

Das „Risiko Mensch“

Kreditinstitute und Finanzdienstleister unterliegen einer strengen Aufsicht mit vorgegebenen Maßnahmen zur Reduktion von Risiken. Die Banken laufen dabei allerdings Gefahr, diese Vorgaben auf das ganze Institut zu übertragen. Risiken werden dadurch jedoch nicht automatisch minimiert. Wie steht es dabei um das „Risiko Mensch“? Welche Maßnahmen können Kreditinstitute treffen, um Mitarbeiter vor bewussten oder unbewussten Fehlern zu bewahren?

Die Mindestanforderungen an das Risikomanagement (kurz MaRisk) schreiben unter anderem eine Funktionstrennung zwischen Markt und Marktfolge innerhalb des Kreditgeschäfts vor. Außerdem wird im Bereich Handel die Trennung der Funktionen Risikocontrolling, Abwicklung und Kontrolle aufbauorganisatorisch und prozessual bis zur Geschäftsleitungsebene gefordert. Die bankaufsichtlichen Anforderungen an die IT (BAIT) präzisieren Teile der MaRisk und Anforderungen zum Risikomanagement aus dem Kreditwesengesetz (KWG). Darin werden auch konkrete Funktionen benannt, z. B. die Informationssicherheitsbeauftragten, die bei entsprechender Aufbau- und Ablauforganisation einem Interessenskonflikt ausgesetzt sein könnten. Diese Art von Funktion muss deshalb innerhalb der Organisationsstruktur und auch von Prozessen unabhängig sein.

Das Bundesamt für Sicherheit in der Informationstechnik (BSI) erläutert in seinen Vorgaben zum IT-Grundschutz unter „Umsetzungshinweise zum Baustein ORP.1 Organisation“[1] die Funktionstrennung im Sinn der Trennung von operativen und kontrollierenden Aufgaben. Folglich macht eine Funktionstrennung durch ein Vier-Augen-Prinzip Sinn, damit eine einzelne Person keine Änderung vornehmen kann. Abnahmetests dürfen zum Beispiel durch Softwareentwickler/innen nicht durchgeführt werden, wenn diese an der Software, die getestet werden soll, selbst Änderungen vorgenommen haben. Da die Aufbau- und Ablauforganisation von jedem Kreditinstitut selbst vorgegeben werden kann, müssen die Kreditinstitute auch die Maßnahmen für eine wirkungsvolle Funktionstrennung auf den beschriebenen Ebenen selbst definieren. Für die beiden Bereiche Handel und Kreditgeschäft wird vorgegeben, dass auf allen Ebenen der Aufbauorganisation und innerhalb der Prozesse die Funktionen zu trennen sind. So lässt sich verhindern, dass eine höhere Ebene (also Vorgesetzte) einen Interessenskonflikt verursachen könnten.

In allen anderen Bereichen hingegen spielen Interessenkonflikte eine geringere Rolle, und sachkundige Mitarbeiter müssen bei der Ausübung einer kontrollierenden Aufgabe selbst eventuelle Unregelmäßigkeiten registrieren. Bei dieser Art von Kontrollfunktion stellt sich am Ende die Frage: „Wird etwas zum Nachteil des Arbeitgebers oder dessen Kunden ausgeführt?“ Die Antwortmöglichkeit lautet lediglich „Ja“ oder „Nein“, und damit ergibt es keinen Sinn für eine höhere Ebene, eine solche Entscheidung zu übersteuern. Insofern betrifft diese Art von Funktionstrennung neben den Kreditinstituten auch IT-Dienstleister beziehungsweise Rechenzentren.

In diesen Fällen kann neben dem Vier-Augen-Prinzip auch die Vergabe von unterschiedlichen Berechtigungen bzw. die Definition sogenannter „toxischer Berechtigungen“ sinnvoll sein. Es ist also zu beachten, dass eine Person nicht die Berechtigungen konkurrierender Funktionen erhält. Welche Funktionen konkurrierend sind, ist abhängig vom einzelnen Prozess, sodass das Kreditinstitut diese Funktionen selbst identifizieren muss.

Trennung innerhalb der Bereiche

Eine Trennung von Anwendungsentwicklung („Change The Bank“) und IT-Betrieb („Run The Bank“) wird allerdings nicht gefordert und ist im Allgemeinen auch nicht notwendig. So erfordert z. B. der Datensicherheitsstandard der Payment Card Industry (PCI) ebenfalls die Trennung von Umgebungen. Allerdings wird explizit erwähnt, dass in der Anwendungsentwicklung ein Administratorkonto in einer Entwicklungsumgebung und gleichzeitig ein Benutzerkonto in einer Produktionsumgebung benutzt werden darf. So kann die Anwendungsentwicklung bei Störungen den IT-Betrieb durch Analysen viel einfacher unterstützen. Wie beim IT-Betrieb muss auch bei der Anwendungsentwicklung die Informationssicherheit gewahrt werden.

In der Regel soll der IT-Betrieb dafür Sorge tragen, dass die Software reibungslos funktioniert. Im Fehlerfall greift er ein und steht als Ansprechpartner für die Anwender zur Verfügung. Der IT-Betrieb muss außerdem die notwendige Hardware bereitstellen, die Software installieren und die technische Konfiguration vornehmen. Um einen sicheren Betrieb der Software zu gewährleisten,

sollte er natürlich auch die Prozesse dahinter verstehen; es muss beispielsweise klar sein, dass bei einem System, das aus mehreren Anwendungen besteht, durch den Ausfall einer Anwendung vielleicht das gesamte System nicht zur Verfügung steht.

Die Anwendungsentwicklung überträgt dieses Wissen an den IT-Betrieb, da sie die Funktionen der Software und das Zusammenspiel verschiedener Komponenten kennt. Im besten Fall besitzt der IT-Betrieb das gleiche Know-how über die Software wie die Anwendungsentwicklung. Dies führt allerdings zu den gleichen Interessenkonflikten, als wenn die Anwendungsentwicklung selbst die Software im Sinn des IT-Betriebs betreut. Denn Änderungen von Konfigurationen müssen auch innerhalb des IT-Betriebs im Vier-Augen-Prinzip geändert werden – jede Änderung muss also durch eine andere Person überprüft und freigegeben werden.

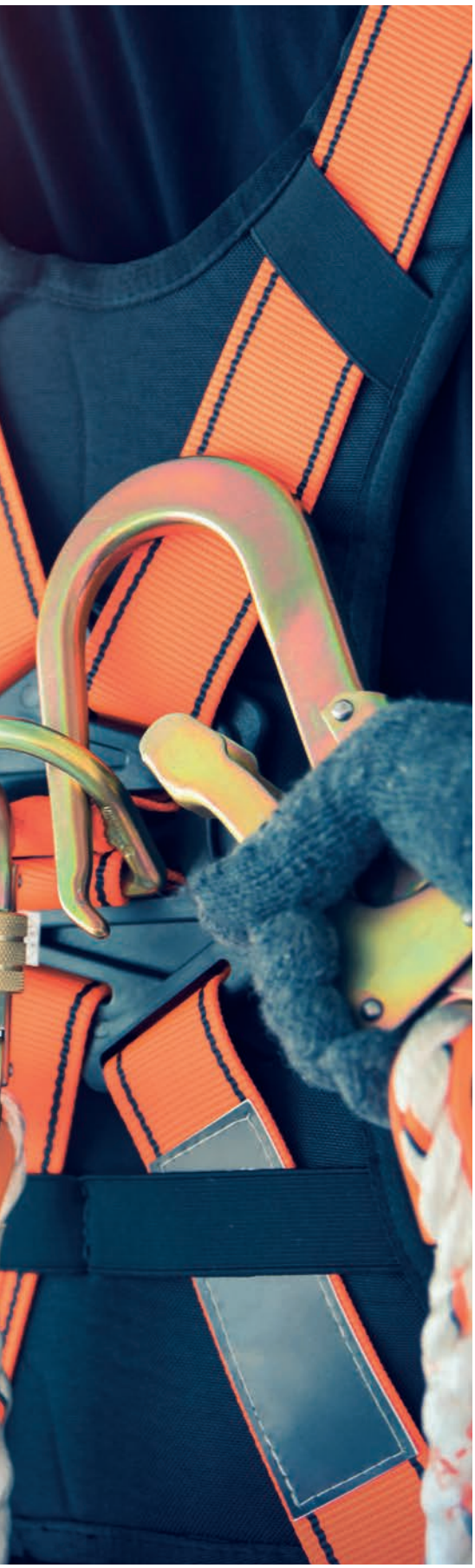
Durch die Trennung von IT-Betrieb und Anwendungsentwicklung wird also nicht automatisch die Funktionstrennung im Sinn des BSI eingehalten, denn es werden operative nicht von kontrollierenden Aufgaben getrennt.

Trennung operativer und kontrollierender Aufgaben

Eins der größten Risiken für ein Kreditinstitut ist das Ausnutzen von Know-how durch einzelne Mitarbeiter. Für die reine Betreuung von Anwendungen (Serververwaltung, Start und Stopp einer Anwendung, Installation, First-Level-Support) wäre es für den IT-Betrieb nicht notwendig, ähnliches Know-how wie die Anwendungsentwicklung aufzubauen. Hier helfen Handbücher, um zu entscheiden, was in einem Fehlerfall zu tun ist.

Da der IT-Betrieb aber für Kontrollaufgaben über mindestens ähnliches Wissen über die Anwendungen verfügen sollte wie die Anwendungsentwicklung, könnte er auch die die gleichen Mechanismen ausnutzen – oder umgekehrt die Anwendungsentwicklung die Unwissenheit des IT-Betriebs. Entsprechend ist es wichtig, dass zu kontrollierende Aufgaben durch Mitarbeiter wahrgenommen werden, die über das notwendige Know-how verfügen, um die Auswirkungen von Änderungen beurteilen zu können.

Die Anwendungsentwicklung muss also in einem Änderungsprozess eine Kontrollfunktion wahrnehmen. Es wäre sonst denkbar, dass eine Abteilung im Rahmen einer Softwareaktualisierung unkontrolliert Dateien über den IT-Betrieb in ein Verzeichnis schieben lässt, die dann – beispielsweise im Zahlungsverkehr – großen Schaden anrichtet. Das kann nur verhindert werden, wenn jemand mit dem ent-



sprechenden Know-how die Daten und Verzeichnisse identifiziert und die richtigen Schlüsse daraus zieht. Und wer könnte das besser als die Anwendungsentwicklung?

Zwar sieht es in der Realität meist so aus, dass nicht alle über das gleiche Know-how verfügen. Dieses lässt sich aber durch Reviews von Code und Dokumenten wie DV-Konzepte ausweiten. Die Reviews sollten auch tatsächlich als kontrollierende Aufgabe in Bezug auf mögliche Risiken wahrgenommen werden.Das bedeutet natürlich auch, dass jemand, der eine Änderung vorgenommen hat, keinen Review für diese durchführen darf. Da aber in der Regel der IT-Betrieb keine Code-Reviews durchführen kann, macht es wenig Sinn, ihm die Kontroll-Funktion zu übertragen. Vielmehr müssen sich die Mitarbeiterinnen und Mitarbeiter der Anwendungsentwicklung gegenseitig kontrollieren. Das führt im Nebeneffekt oft dazu, dass sich die gelieferte Softwarequalität erhöht.

Die Reviews dürfen aber nicht bei den Code-Reviews enden. Sie müssen konsequent von der Änderung der Software über die Tests bis hin zur Installation in der produktiven Umgebung unter Berücksichtigung der Trennung zwischen operativen und kontrollierenden Aufgaben durchgeführt werden.

Wer bei jeglicher Änderung der produktiven Umgebung den gleichen Prozess ablaufen lässt, landet schnell bei der Automatisierung einzelner Prozessschritte. Dabei sind auch Änderungen im Rahmen von Störungen beziehungsweise Fehlern eingeschlossen, bei denen z. B. SQL- oder Shell-Skripte ausgeführt werden müssen. Die Automatisierung sollte auch in diesen Fällen das Ausrollen von Änderungen innerhalb weniger Minuten inklusive aller Kontrollmechanismen ermöglichen.

Trennung – aber wie?

Um eine Trennung zwischen operativen und kontrollierenden Aufgaben zu ermöglichen, müssen entsprechende Kontrollpunkte im Prozess eingerichtet werden. Für Code-Reviews könnte man etwa sogenannte Pull-Requests einführen, die ein Review ermöglichen, bevor Änderungen in die Codebasis überführt werden.

Durch die Automatisierung und einzelne Freigabepunkte kann die Software über mehrere Umgebungen (Test, Integration, Abnahme, Produktion und mehr) ausgerollt werden. Die operativen und kontrollierenden Funktionen müssen durch Werkzeuge gestützt getrennt werden, wobei diese Tools in verschiedenen Fällen entscheiden können sollten, wer im jeweiligen Fall eine operative Funktion innehat und somit eine kontrollierende Funktion wahrnehmen darf.

FAZIT

Noch viel mehr als in anderen Unternehmen gilt es in Kreditinstituten, erdenkliche Risiken soweit wie möglich mit geeigneten Maßnahmen zu minimieren. Eine schlichte Trennung zwischen der Anwendungsentwicklung und dem IT-Betrieb ist sicherlich keine ausreichende Maßnahme, wenn unklar bleibt, bei welchen Prozessen welche Personen welche Funktionen operativ oder kontrollierend bekleiden sollen. Wichtig ist, dass die einzelnen Schritte – vom Schreiben von Code bis hin zur Installation der Software in der Produktionsumgebung – getrennt betrachtet werden. Nur dann kann ermittelt werden, wer in einer Kombination aus Know-how und Gelegenheit eventuell zum Nachteil des Kreditinstituts oder dessen Kunden handeln könnte.

Autor

Jens Kötterheinrich ist Informatiker und unterstützt seit 2007 die Unternehmensberatung Beckmann & Partner CONSULT (Bielefeld) in Softwareentwicklungs-Projekten. Sein besonderer Schwerpunkt liegt auf der Softwareentwicklung mit Java.

1 Abzurufen unter www.bsi.bund.de.