Digital Escrow

Remmertz / Kast

2022
ISBN 978-3-406-77386-0
C.H.BECK

keit für Systeme gefordert, ist das **Service Level Agreement** (SLA) mit den Vertragspartnern gefährdet, wenn ein längerer Stromausfall ansteht und keine Netzersatzgeräte vorhanden sind.

II. Hinterlegungsgegenstand

Zentrale Komponente ist die hinterlegte Software, die im Herausgabefall die Fortführung der Geschäftsprozesse sicherstellen soll. Risiken, die eine Inbetriebnahme, einen sicheren Betrieb und eine lizenzkonforme Nutzung gefährden, müssen im Vorfeld identifiziert werden. Neben der Beschaffenheit des Hinterlegungsgegenstands selbst mit all seinen Anteilen sind auch die Risiken zu berücksichtigen, die aufgrund der Einbeziehung externer Dienste entstehen. 43

1. Proprietärer Source Code

Eine funktionierende Übersetzung und Verlinkung (→ Rn. 16–19) funktioniert nur durch eine vollständige und integre **Entwicklungsbasis.** Folgende Risikofaktoren gefährden hierbei den Wiederherstellungserfolg. 44

a) Fehlende Anteile. Nicht jede Komponente, die Teil der Entwicklung ist, unterliegt ständigen Anpassungen. Viele Bibliotheken, welche Teil der Basisfunktionalität sind, ändern sich in der Regel nicht mehr und werden daher nur noch verlinkt. Der Source Code dieser Bibliotheken ist daher auch nicht Teil der Entwicklungsumgebung und wird an anderer Stelle archiviert. Somit besteht die Gefahr, dass diese Anteile bei der Hinterlegung nicht berücksichtigt werden und bei notwendigen Anpassungen nicht zur Verfügung stehen. Ein weiterer Grund für **unvollständigen Source Code** ist die Erzeugung aus anderen Quellen. Vor allem auf Architektur und Designebene werden abstrakte Modellierungswerkzeuge zur Anwendungsentwicklung eingesetzt, die den eigentlichen Source Code generieren. Eine manuelle Pflege ist also nur auf Basis dieser **abstrakten Codebestandteile** möglich. Beispiel hierzu sind WSDL[12]-Dateien, die den Informationsaustausch zweier Systeme beschreiben und komplett generiert werden. 45

b) Falsche Version oder Ausführung. Operative Systeme unterliegen einem ständigen Anpassungsprozess, dh die implementierten Funktionen müssen stetig Schritt halten mit den Geschäftsprozessen oder Produktänderungen. Wird versäumt, diese **Aktualisierungen** zeitgleich auch am Hinterlegungsgegenstand durchzuführen, laufen die Versionen und Software-Stände auseinander und eine Wiederherstellung im Herausgabefall liefert nicht das erwartete Ergebnis. 46

c) Software-Schwachstellen. Schwachstellen innerhalb der **Implementierung** machen Software-Systeme für Angreifer verwundbar. Solche Schwachstellen entstehen durch Programmierfehler oder durch die Übernahme ungeprüfter Programmroutinen. Ungeprüfte Speicherbelegungen beispielsweise bergen die Gefahr eines provozierten Speicherüberlaufs, in dessen Folge ein Angreifer beliebigen Code auf dem Zielsystem ausführen kann. Häufig werden auch Eingaben ins System nur schlecht oder gar nicht validiert. Folge kann der unautorisierte Zugriff auf Subsysteme oder ein privilegierter Zugang zum System mit erweiterten Rechten sein (→ Rn. 79). 47

2. FOSS

Der Einsatz von **Free and Open Source Software** bietet viele Vorteile für die Entwicklung wie die Einführung einheitlicher offener Standards, Zeit- und Kostenersparnis und 48

[12] Web Services Description Language.

Transparenz hinsichtlich der Implementierung. Die speziellen Lizenzbedingungen in der FOSS bergen aber auch besondere Risiken für die Kommerzialisierung.

49 **a) Lizenzrisiken.** Mit der Nutzung und Weitergabe von FOSS verpflichtet sich der Verwender zur Einhaltung der jeweiligen Lizenzbedingungen. Diese Auflagen betreffen mindestens die Erhaltung von Copyrightvermerken und Lizenztexten bei permissiven Lizenzauflagen. Software unter komplexeren und strengeren Lizenzen wie zB der GNU General Public License (GPL) in den verschiedenen Versionen bergen das Risiko eines Copylefts. Der **Copyleft**-Effekt tritt in Kraft, wenn eine entsprechend lizenzierte Software dynamisch oder statisch mit der eigenen (proprietären) Software verlinkt wird. Hierbei geht die Lizenzgültigkeit der FOSS mit allen Auflagen auf den eigenen Quellcode über. Der Eigentümer des proprietären Codes verliert in dieser Kombination die exklusiven Rechte an seinem eigenen Werk und es gelten die gleichen Auflagen hinsichtlich der Veröffentlichung des Quellcodes, des Rechts auf Veränderung und des zur Weitergabe an Dritte. Insbesondere bei kommerzieller Nutzung entstehen hierbei für den Verwender aber massive Nachteile, da er die Verwertungsrechte an seinem eigenen Source Code verliert und dem Empfänger des Binärcodes auch den Quellcode überlassen muss. Auch dieser hat wiederum die gleichen Rechte und Pflichten, die sich aus der Lizenz ergeben. Verstößt der Verwender jedoch gegen die Lizenzbedingungen, erlischt die Lizenz und somit auch das Recht die Software zu verwenden, zu ändern und weiterzugeben. Es liegt dann eine **Urheberrechtsverletzung** vor.

50 **b) Risiken in der Informationssicherheit.** FOSS wird oft unverändert in den Projekten eingesetzt. Daher verzichten viele Entwickler auf eine Übersetzung des Quellcodes im Rahmen der Software-Erstellung. Vielmehr werden die eingesetzten Bibliotheken als Binärdatei direkt mit den eigenen Anteilen verlinkt. Die verwendeten Binärdateien werden entweder zuvor heruntergeladen oder mittels Scripts direkt auf öffentlichen Internetservern für den Build (→ Rn. 14–19) referenziert. Nachteil ist, dass keine verlässliche Prüfung mehr möglich ist, ob die Binärdatei tatsächlich der beabsichtigten Version oder Ausführung entspricht. Es geht letztlich der Vorteil der **Codetransparenz** verloren.

51 Ein weiteres IT-Sicherheitsrisiko bei Einsatz von FOSS entsteht durch die Verwendung schwachstellenbehafteter Softwareversionen, deren Sicherheitslücken erst im Laufe der Zeit bekannt werden. Der Vorteil **öffentlich bekannter Schwachstellen** innerhalb von FOSS kehrt sich zum Nachteil für die eigene Entwicklung, wenn diese Anteile ungepatcht im Hinterlegungsgegenstand verbleiben.

52 Das Angebot an FOSS ist extrem groß. Aber nicht jedes Open Source-Projekt „überlebt". In Folge entstehen **verwaiste Softwareprojekte,** für die es keine Community und keine Unterstützung mehr gibt. Werden Anpassungen notwendig, weil Schwachstellen oder Bugs bekannt werden oder weil sich Standards bzw. Schnittstellen ändern, sind diese Änderungen durch den Verwender selbst zu leisten. Hinzu kommt, dass gerade bei kleineren Projekten oder Einzelentwicklungen keine oder nur sehr spärliche **Entwicklungsdokumentation** verfügbar ist.

3. Kommerzielle Software

53 In der Entwicklung werden nicht alle Funktionen neu geschrieben. Auch dem Einsatz von FOSS sind Grenzen gesetzt, zB wenn es um die Einbindung spezieller herstellerspezifischer Schnittstellen oder Funktionen geht. Auch kann es Vorgaben durch Kunden oder Empfänger von Produkten geben, die eine Einbeziehung kommerzieller Softwarebestandteile in die eigene Entwicklung erforderlich machen. Hierbei liegen die Risiken einerseits in den **spezifischen Lizenzbedingungen** und vertraglichen Regelungen mit dem Urheber der hinterlegten Software (Ist es möglich, die Lizenz im Herausgabefall auf einen neuen Vertragspartner zu übertragen?). Zum anderen besteht mit der Verwendung kommerzieller Software ein erhöhtes **Obsoleszenz-Risiko,** da in der Regel kein Source Code

verfügbar ist und keine Erlaubnis zur Änderung seitens des Urhebers vorliegt. Es gibt keine Software ohne Sicherheitslücken. Diese bedrohen daher auch kommerzielle Bibliotheken. Das wird insbesondere dann zu einem Problem, wenn der Hersteller nicht mehr existiert, oder die betroffene Software keinen Support mehr erhält. Aufgedeckte Sicherheitslücken bleiben somit in der eigenen Software ungepatcht oder werden erst gar nicht entdeckt.

4. Dokumentation

Eine ausreichende und aktuelle Dokumentation, begleitend zur Entwicklung, ist ein wesentlicher Baustein zum Aufbau, Wiederherstellung und Fortführung eines Entwicklungsprozesses. Größtes Risiko ist hierbei eine **unvollständige** oder **veraltete Dokumentation.** Häufig werden solche Dokumente nur bei der Ersteinrichtung verfasst und anschließend nicht mehr fortgeschrieben. Spätere Änderungen an der Software, der Konfiguration oder den Entwicklungswerkzeugen und Plattformen sind somit nicht dokumentiert. Das verzögert die Wiederherstellung der Entwicklungsumgebung und erschwert die künftige Pflege der hinterlegten Software. Sind Zugänge oder Passwörter nicht dokumentiert oder aktuell, kann dies die Wiederherstellung- und Pflege sogar verhindern. Eine Sichtung des Hinterlegungsgegenstands zur Bewertung möglicher Fragen hinsichtlich des Datenschutzes, der Informationssicherheit oder anderer Qualitätsmerkmale ist bei unzureichender Dokumentation ebenfalls erschwert. 54

5. Externe Services (Cloud Computing)

Nutzt die hinterlegte Software externe Servicemodelle des **Cloud Computing**, sind spezifische Risiken zu berücksichtigen, die Einfluss auf die Verfügbarkeit und Integrität des Gesamtsystems haben. Grundsätzlich gilt, dass jeder Service, der integrativer Bestandteil der Gesamtlösung ist, an den Empfänger der Hinterlegung möglichst ohne **Anpassungsaufwand** und in bestehender Form übertragen und weiter betrieben werden kann und somit das Gesamtsystem in seiner Funktionalität nicht eingeschränkt wird. Aus dem Blickwinkel der Sicherheit gilt grundsätzlich, dass ein mehr oder weniger großer Anteil des operativen Systems außerhalb des **Verantwortungsbereichs** des Nutzers liegt. Abhängig vom gewählten Service-Modell betrifft das sowohl die IT-Infrastruktur wie auch Teile der genutzten Software. Daher muss hier in der Frage des Datenschutzes und der IT-Sicherheit dem Cloud-Betreiber vertraut werden. 55

D. Maßnahmen des Informations- und Datenschutzes und der Validierung

I. Vorgehen der methodischen Informationssicherheit

Grundsätzlich wird mit der **methodischen Informationssicherheit** eine formale Vorgehensweise zur Etablierung eines angemessenen Schutzniveaus mindestens in den Schutzzielen **Vertraulichkeit, Integrität** und **Verfügbarkeit** für informationsverarbeitende Systeme angestrebt. Dieser Prozess berührt hierbei alle Phasen des Lebenszyklus eines betrachteten Systems, also beginnend mit der Planung, während der Entwicklung und dem anschließenden operativen Betrieb bis hin zur Stilllegung. Methodische Informationssicherheit, speziell im Escrow fokussiert aber insbesondere auf die Entwicklung und Erstellung der Software. Zur Validierung der Informationssicherheit braucht es spezifische Bewertungskriterien. 56

Speziell für die **Bewertung der Sicherheit von IT-Produkten** wurde bereits 1994 auf internationaler Ebene das **Common Criteria**-Vorhabensmodell vereinbart[13]. Die CC sind im internationalem ISO Standard 15408-1 bis 3 festgeschrieben. Als ein relevanter 57

[13] www.commoncriteriaportal.org/iccc/ICCC_arc/history.htm (1. 1. 2022).

Standard zur Bewertung der Softwareentwicklung als Prozess eignet sich die Common Criteria jedoch nicht, da sie zwar Kriterien zur Bewertung informationstechnischer Produkte liefert, nicht aber den Entwicklungsprozess selbst fokussiert. Der **Standard IEC 62443** i[14] kann hierzu eine Basis liefern. Teil 4.1 des Standards beschreibt die Anforderungen an eine sichere Produktentwicklung. Der Standard basiert auf der Vorgehensweise des Security Development Lifecycle (→ Rn. 77) ergänzt um die Aspekte eines Informationssicherheitsmanagementsystems (→ Rn. 58 ff). Allerdings fokussiert er primär auf ICS-Systeme, bezieht sich also auf Industriesteuerungen. In den folgenden Unterkapiteln wird eine Vorgehensweise mit Fokussierung auf die Entwicklung und Pflege zu hinterlegender Software beschrieben.

1. Informationssicherheitsmanagementsystem (ISMS)

58 Der englische Begriff Information Security Management System oder kurz ISMS wird in der **ISO 27000:2015** wie folgt definiert: *„Ein Managementsystem zur Informationssicherheit umfasst Politik, Verfahren, Richtlinien und damit verbundene Ressourcen und Tätigkeiten, die von einer Organisation gesteuert werden, um ihre Informationswerte zu schützen."*

Auch der IT-Grundschutz und weitere Standards (→ Rn. 72–75) beschreiben ein ISMS als ein Managementsystem, dass alle Regelungen zur Steuerung und Lenkung der Informationssicherheit umfasst. Die Einführung eines ISMS beeinflusst nahezu alle Bereiche eines Unternehmens, da neben der Informationstechnik auch die Organisation, Infrastruktur, Prozesse und Personal einen entscheidenden Einfluss auf den sicheren Umgang mit Informationen haben.

[14] IEC 62443-4-1:2018 Security for industrial automation and control systems – Part 4-1: Secure product development lifecycle requirements. Zugehörige DIN Norm: DIN EN IEC 62443-4-1*VDE 0802-4-1.

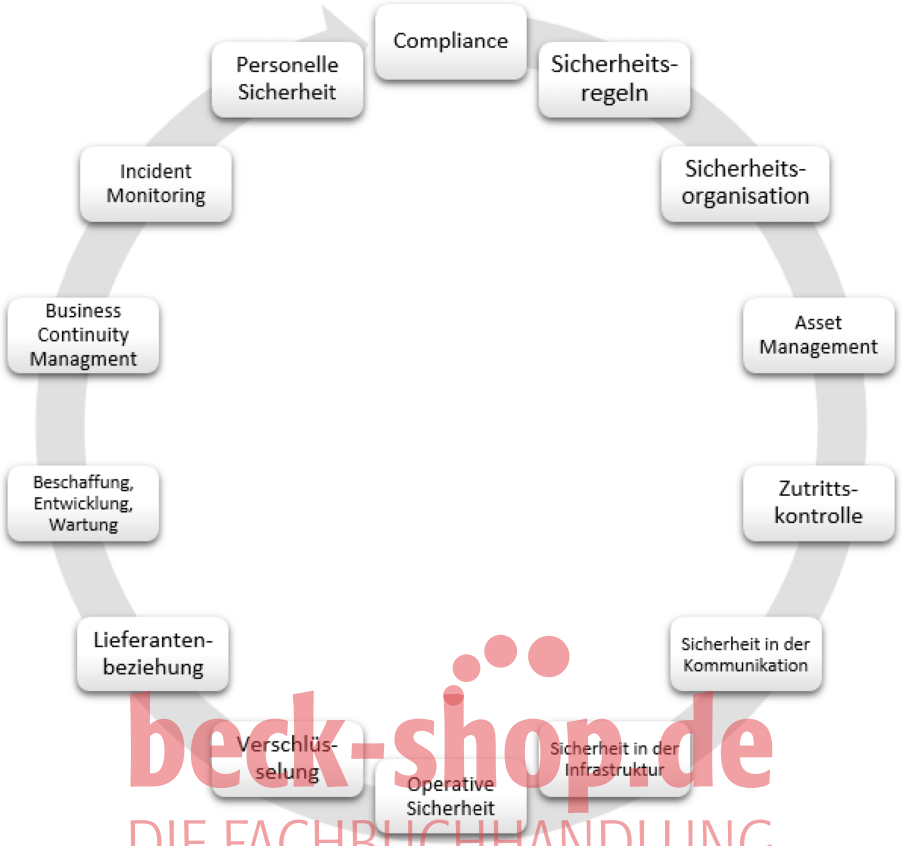

Abb. 3: Informationssicherheitsmanagementsystem

Ein vollständiges ISMS berücksichtigt nach Möglichkeit alle **Prozesse** und **Ressourcen** 59
eines Unternehmens:

- Zuständigkeiten und Aufgabenverteilung (zB Rollen/Rechte-Zuordnung)
- Umgang mit schützenswerten Informationen und IT-Komponenten
- Training und Sensibilisierung zur Informationssicherheit
- Personal
- Kommunikation
- Datenschutz
- Verschlüsselung von Daten
- Hard- und Softwareentwicklung
- Erfassung und das Änderungsmanagement von Hard- und Software
- Datensicherung auf allen Ebenen des Geschäftsbetriebs
- Schutz vor Schadsoftware und Viren
- Verhalten bei Sicherheitsvorfällen
- Notfallvorsorge und Aufrechterhaltung der Geschäftstätigkeit im Notfall[15]
- Zutritts- und Zugriffskontrolle
- sichere Infrastruktur
- Beschaffung und Lieferantenbeziehung
- Outsourcing und Cloud-Nutzung

[15] Engl. Business Continuity Management (BCM).

60 Die Entwicklungsabteilung muss für sichere Infrastruktur und überprüftes bzw vertrauenswürdiges Personal sorgen. Entwickler müssen hinsichtlich IT-Sicherheit und sichere Programmierung regelmäßig geschult werden. Bestehende IT-Systeme sind aktuell zu halten und mit entsprechendem Schutz gegen Schadsoftware zu versehen. Wenn sich Entwicklungssysteme das Netzwerk mit anderen Funktionsbereichen teilen, muss über eine Segmentierung oder Trennung der Netze nachgedacht werden. Informationen sind gegebenenfalls für den Transport und die Ablage zu verschlüsseln. Die zu erstellende Software muss kontinuierlich auf Schwachstellen, unsichere Programmierung, mögliche Lizenzverstöße, Vollständigkeit, Übersetzbarkeit und Konfiguration hin überprüft werden. Hierzu liefern → Rn. 43 ff. detaillierte Informationen.

2. Sicherheitsziele

61 Informationen werden durch verschiedene Angriffsvektoren bedroht. Um einen wirksamen Schutz zu etablieren, ist es zunächst mal wichtig, diese Angriffspunkte zu identifizieren. Mit der **Vertraulichkeit** (engl. confidentiality) wird sichergestellt, dass zu schützende Informationen nur durch autorisierte Empfänger zur Kenntnis genommen werden können. Mit der **Integrität** (engl. integrity) wird die Unversehrtheit von Informationen garantiert. Ein berechtigter Empfänger dieser Information kann sich also darauf verlassen, dass keine unautorisierte Änderung an dieser Information erfolgt ist. Das dritte Sicherheitsziel, was durch ein ISMS sichergestellt wird, ist die **Verfügbarkeit** einer zu schützenden Information oder eines Services. Aufgabe ist die Verhinderung eines Ausfalls oder des Zugriffs auf diese Information oder Service. Ein berechtigter Nutzer muss sich also darauf verlassen können, eine zu schützende Information zu vorgesehenen Zeiten zugreifen oder einen zu schützenden Service erreichen und nutzen zu können.

62 Darüber hinaus kennt insbesondere der **Datenschutz** weitere spezifische Sicherheitsziele, die vor allen Dingen dem Schutz der Daten Dritter dienen. So verlangt die DS-GVO in Art. 5,[16] dass die Speicherung von Daten Dritter „…rechtmäßig, zweckbindend und richtig … sein muss." Ferner sollen Daten nur in einem Umfang und solange gespeichert werden, wie es der Zweck erfordert. Sichergestellt wird dies vor allem durch die primären Sicherheitsziele des Datenschutzes:

- **Transparenz:** Die Verarbeitung personenbezogener Daten muss durch einen Betroffenen nachvollziehbar, überprüfbar und bewertbar sein.
- **Nichtverkettbarkeit:** Eine Verarbeitung personenbezogener Daten darf nur für den vorgesehenen Zweck erfolgen. Eine Verknüpfung oder Aggregierung mit weiteren Daten über dieses Ziel hinaus muss unterbunden werden.
- **Intervenierbarkeit:** Es müssen Verfahren etabliert werden, dass Betroffene ihre Rechte an über sie gespeicherte Daten ausüben können. Dazu gehören die Auskunft, die Benachrichtigung, die Berichtigung, die Sperrung oder die Löschung von Daten.
- **Datenportabilität:** Ein Betroffener muss die Möglichkeit erhalten, seine Daten zu einem anderen Provider oder Dienstleister zu übertragen.
- **Recht auf Vergessenwerden:** Gespeicherte Daten sind auf Veranlassung des Betroffenen zu löschen, sofern keine anderweitigen gesetzlichen Regelungen entgegenstehen.

Nach Bedarf werden weitere Sicherheitsziele berücksichtigt, die zum Beispiel die **Verbindlichkeit** oder **Zurechenbarkeit** einer Information zusichern.

3. Sicherheitsprozess

63 Der Sicherheitsprozess führt ein angemessenes Schutzniveau in einen bestehenden IT-Verbund und Organisation ein. Bewährt hat sich hierbei die Vorgehensweise nach dem

[16] dejure.org/gesetze/DSGVO/5.html (1.1.2022).

PDCA[17]-Zyklus. Es werden hierbei zunächst Sicherheitsziele und Geltungsbereich (PLAN) definiert und anschließend die notwendigen Maßnahmen implementiert (DO). In der Phase der Verifikation (CHECK) findet eine kontinuierliche Prüfung auf Wirksamkeit und Aktualität dieser Maßnahmen statt. Bei Abweichungen werden Planung und Implementierung entsprechend angepasst (ACT). Somit unterliegt der Sicherheitsprozess einer kontinuierlichen Anpassung an Veränderungen.

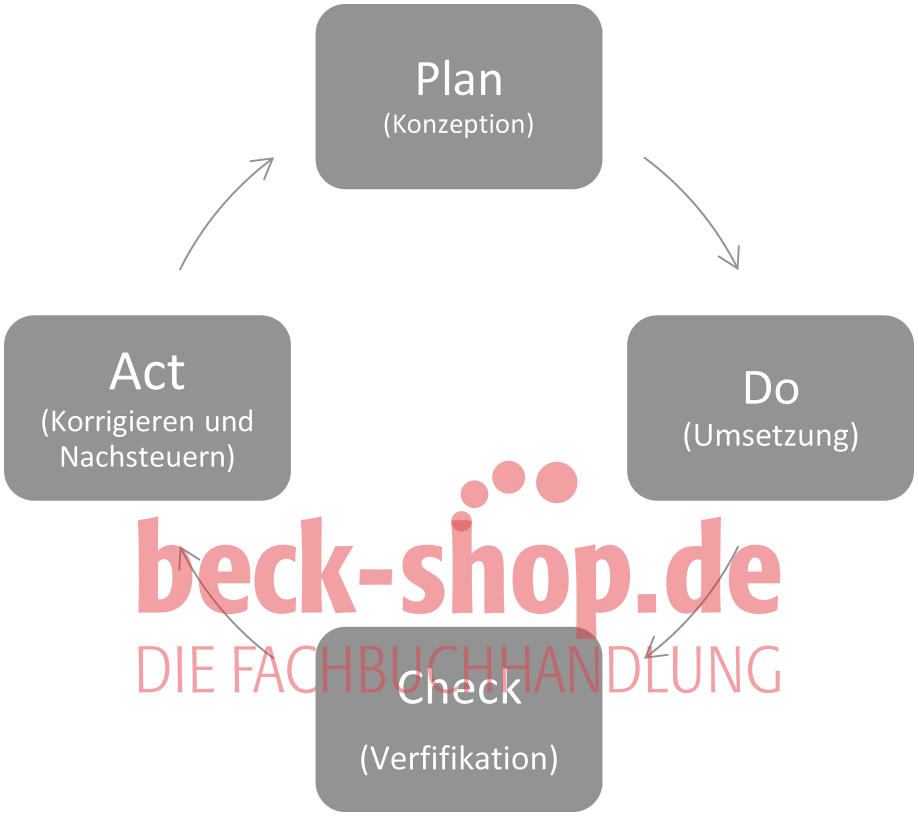

Abb. 4: PDCA-Wirkungskreis

64 **a) Planung und Konzeption (PLAN).** Der Planungsabschnitt legt fest, welche **Sicherheitsziele** (zB Schutz der Softwareentwicklung) verfolgt werden sollen und wo der **Geltungsbereich** liegt (zB Produktentwicklung). Mit der Planung wird ferner die **IT-Sicherheitsorganisation** etabliert und deren Verantwortlichen und Rollen benannt. Zur Umsetzung der Aufgaben wird diese Organisation mit den notwendigen **Ressourcen** und Mitteln ausgestattet. Wichtiger Teil des Prozesses ist die **Informationssicherheitskonzeptionierung.** Dieser Abschnitt dient der Erreichung eines angemessenen Schutzniveaus innerhalb des ISMS. Das Informationssicherheitskonzept untergliedert sich hierbei in mehrere Abschnitte.

65 **b) Erfassung des Informationsverbunds.** Für ein notwendiges aber auch wirtschaftlich vertretbares ISMS ist der erste Schritt, kritische **Geschäftsprozesse** und damit verbunde-

[17] PDCA: Plan-Do-Check-Act, auch Demingkreis genannt. Ein vierphasiger Prozess, welcher der kontinuierlichen Verbesserung dient. Beschreibung erstmals 1939 durch *William Edwards Deming*.

ne Informationen zu identifizieren. Zur Erfassung des **Informationsverbunds** ist eine Aufstellung aller Prozesse, Dienste, Systeme und Infrastruktur notwendig. Dies betrifft ua Anwendungen, IT-Systeme, Netzwerke und Gebäude als konkrete Systembestandteile. Zu den übergreifenden Prozessbausteinen gehören ua das Personal, Schulungskonzepte sowie das Notfall- und Sicherheitsmanagement. Diese Auflistung muss belastbar und vollständig gemäß dem zuvor festgelegten Geltungsbereich erfolgen. Hierbei ist es notwendig, auch alle Abhängigkeiten zwischen den jeweiligen Zielobjekten zu dokumentieren. Diese Abhängigkeiten spielen später eine wichtige Rolle bei der Schutzbedarfsvererbung. Es werden dann die erfassten Zielobjekte des Informationsverbunds, welche der Bereitstellung und Verarbeitung der zu schützenden Geschäftsprozesse und Informationen dienen, zugeordnet. Somit kann die Schutzbedürftigkeit von Informationen transparent auf die zugrunde liegende IT, die Prozesse und Dienste übertragen werden. Um den Verbund übersichtlich zu halten und Redundanzen bei den umzusetzenden Maßnahmen und zu betrachtenden Gefährdungen zu vermeiden, werden einzelne Zielobjekte zu Gruppen zusammengefasst. Eine Zusammenfassung von Zielobjekten ist möglich, wenn sie ua vom gleichen Typ, von gleicher Schutzbedürftigkeit und Netzzuordnung sind. Eine Zusammenfassung muss immer individuell geprüft werden und alle Bedingungen müssen gleichermaßen zutreffen. Die erfassten und bereinigten Zielobjekte werden in einem vereinheitlichten Netzplan zusammengefasst, welcher Netzsegmente, IT-Systeme sowie Räume und Gebäude in eine visuelle Darstellung überführt.

66 **c) Schutzbedarf feststellen.** Ziel ist es, den Schutzbedarf hinsichtlich **Vertraulichkeit, Integrität** und **Verfügbarkeit** für alle im Informationsverbund enthaltenen Zielobjekte und Gruppen zu bewerten. Die Bewertung erfolgt zunächst für die Unternehmensziele, also all die Informationen, Dienste und Zielobjekte, die als schutzwürdig identifiziert wurden. Hilfreich ist hierbei die Schutzbedarfsvererbung. Existieren Abhängigkeiten zwischen Unternehmenszielen und verarbeitenden Systemen, so vererbt sich die Schutzbedürftigkeit des Unternehmensziels auf diese Zielobjekte. Diese Einstufung ist nochmals individuell für jedes Element zu überprüfen, da es immer besondere Umstände geben kann, die eine abweichende Bewertung erfordern.

67 **d) Risikoanalyse und -behandlung durchführen.** Die **Risikoanalyse** ist als iterativer Prozess zu verstehen. Eine Analyse erfolgt auf Basis bestehender Schwachstellen und Bedrohungen und resultiert in einem sich daraus ergebendem Risiko. Es gibt unterschiedliche Verfahren wie zB **COBIT**, ISO 27005 oder **BSI 200-3** zur Risikobewertung, die zur Anwendung kommen können. Eine Bewertung der Risiken erfolgt hinsichtlich möglicher Konsequenzen des Risikoeintritts,[18] der Schadensshöhe, der Eintrittswahrscheinlichkeit und der Risikostufe für die Sicherheitsziele. Die anschließende Risikobehandlung resultiert in vier Optionen:

- Risikovermeidung (zB durch Anpassen von Prozessen in organisatorischen Abläufen, Weglassen von Funktionen in der SW-Entwicklung etc.)
- Risikoreduzierung (zB durch Festlegung von Sicherheitsmaßnahmen)
- Risikoübertragung (zB durch Abschließen von Versicherungen)
- Risikoakzeptanz (zB bei geringem Risiko)

68 **e) Maßnahmen festlegen.** Innerhalb des Risikomanagements identifizierte Risiken lassen sich häufig durch **Sicherheitsmaßnahmen** reduzieren. Diese Maßnahmen beinhalten konkrete Vorgaben, um zB bestehende Assets oder Komponenten sicherer einzusetzen, Informationen besser zu schützen oder Prozesse sicherer zu gestalten. Mögliche Maßnahmen können auch das Personal oder die bestehende Infrastruktur betreffen.

69 **f) Implementierung und Umsetzung (DO).** Die auf Basis der Risikobehandlung ermittelten Maßnahmen und die im Rahmen der Planung vereinbarten Regelungen müssen

[18] Engl. *impact*.