Musterprozess
für einen sicheren Produkt-Entwicklungslebenszyklus nach IEC 62443-4-1

Quelle: blackboard/stock.adobe.com

Gefördert durch:

Bundesministerium für Wirtschaft und Klimaschutz

aufgrund eines Beschlusses des Deutschen Bundestages

Inhaltsverzeichnis

Autoren

Prof. Dr.-Ing. Karl-Heinz Niemann
ist Professor im Fachbereich Prozessinformatik und Automatisierungstechnik (PIA) an der Hochschule Hannover.

Jan-Niklas Puls
arbeitet als wissenschaftlicher Mitarbeiter an der Hochschule Hannover und ist Experte für IT-Sicherheit im Mittelstand-Digital Zentrum Hannover.

Haftungsausschluss

Die diesem Dokument zu Grunde liegenden Informationen wurden mit größtmöglicher Sorgfalt recherchiert und zusammengestellt. Dennoch wird es ohne eine Gewährleistung zur Verfügung gestellt. Die Autoren lehnen ausdrücklich jede Art von vertraglicher oder gesetzlicher Haftung für dieses Dokument ab.

In keinem Fall sind die Autoren für Schäden verantwortlich, die durch Fehler oder fehlende Informationen in diesem Dokument entstehen könnten. Logos und Markennamen werden ohne Hinweis auf ggf. bestehende Schutzrechte verwendet.

Abkürzungsverzeichnis

Abkürzung	Deutscher Begriff	Englischer Begriff
SM	IT-Sicherheitsmanagement	Security Management
SR	IT-Sicherheitsanforderungen	Security Requirements
SD	Gesicherter Entwurf	Security Design
SI	Gesicherte Implementierung	Secure Implementation
SVV	Verifikations- und Validierungs-prüfung der IT-Sicherheit	Security Verification and Validation
DM	Mängelbehandlung	Defect Management
SUM	Verwaltung von IT-Sicherheitsupdates	Security Update Management
SG	IT-Sicherheitsrichtlinien	Security Guidelines
SDL	Gesicherter Entwicklungslebenszyklus	Security Development Life-cycle
SRS	Spezifikation der Sicherheits-anforderungen	Security-Requirement-Spezifikation
KMU	Kleine und mittlere Unternehmen	Small and medium-sized enterprises
CVSS	Allgemeines Bewertungssystem für Schwachstellen	Common Vulnerability Scoring System
CERT	Computersicherheits-Ereignis- und Reaktionsteam	Computer Emergency Response Team

DOI
10.25968/opus-2935

Version 1.0
Dieses Dokument beschreibt einen Musterprozess, der an die [IEC_62443-4-1] angelehnt ist.
Bei diesem Dokument handelt es sich um die Version 1.0.

1 Einleitung

Kleine und mittlere Unternehmen (KMU), die dem Bereich der Automatisierungstechnik zuliefern, stehen vor der wachsenden Herausforderung, dass Kundinnen und Kunden vermehrt Produkte fordern, die im Sinne der IT-Sicherheit „sicher" entwickelt werden. Die Norm [IEC_62443-4-1] beschreibt einen solchen sichereren Produkt-Entwicklungslebenszyklus. Derartige Standards stellen hohe Anforderungen an die Organisation der Prozesse. Um die Umsetzung dieses Prozesses auch KMU zu ermöglichen, werden im folgenden Dokument Musterprozesse beschrieben, die Unternehmen befähigen die Anforderungen zu verstehen und im eigenen Unternehmen ein-, bzw. fortzuführen.

Entstanden ist dieses Dokument in Kooperation mit der SSV Software Systems GmbH. Wir bedanken uns für die vertrauensvolle und professionelle Zusammenarbeit.

2 Einführung in die IEC 62443 Normenreihe

Die Normreihe IEC 62443 liefert Standards, Spezifikationen und technische Berichte, um das Thema Informationssicherheit in Automatisierungssystemen zu adressieren [GMO2018]. Die verschiedenen Teile der Normenreihe richten sich an unterschiedliche Unternehmen der Wertschöpfungskette, unterteilt in vier Hauptkategorien. Abbildung 1 stellt die vier Hauptkategorien sowie die dazugehörigen Teile und Zugehörigkeiten dar. Zurzeit sind noch nicht alle Teile der Normreihe veröffentlicht worden. Die bereits veröffentlichten Normen sind rot hinterlegt, wohingegen die noch nicht veröffentlichten Normen grau dargestellt sind.

In den ersten Teilen der Normreihe [IEC_62443-1-1], [IEC_62443-1-2], [IEC_62443-1-3], [IEC_62443-1-4] werden allgemeine Grundlagen thematisiert. Die nächsten Teile [IEC_62443-2-1], [IEC_62443-2-2], [IEC_62443-2-3], [IEC_62443-2-4], [IEC_62443-2-5] adressieren Anforderungen und Leitfäden an Betreiber und Dienstleister von Automatisierungssystemen. Die dritte Hauptkategorie in den Teilen [IEC_62443-3-1], [IEC_62443-3-2] und [IEC_62443-3-3] stellt detaillierte Anforderungen an Automatisierungssysteme dar. Die Teile [IEC_62443-4-1] und [IEC_62443-4-2] beschreiben Anforderungen an den sicheren Entwicklungslebenszyklus sowie einzelne Automatisierungskomponenten. Somit richten sich die zuletzt genannten Teile an Hersteller bzw. Lieferanten einzelner Komponenten eines Automatisierungssystems [NIE2021].

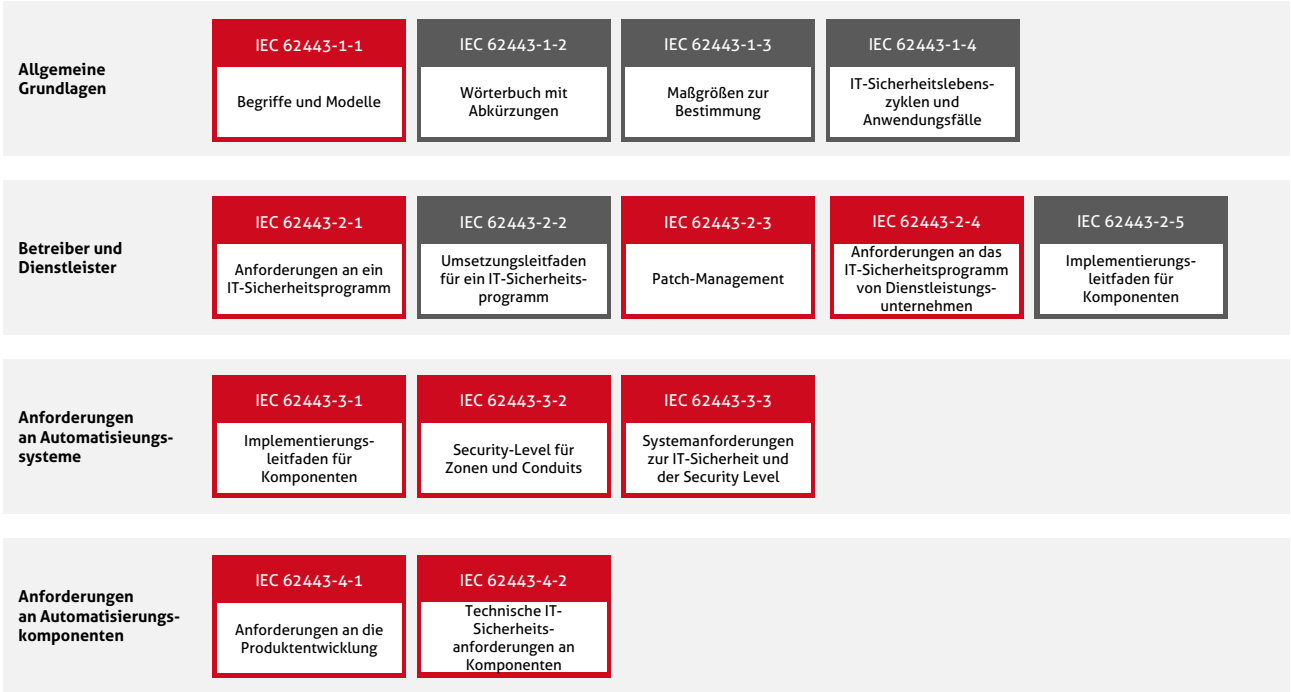

Abbildung 1: Normreihe IEC 62443, angelehnt an [DKE2022]

3 Fokus auf die IEC 62443-4-1

Während in der [IEC_62443-4-2] konkrete technische Anforderungen an Komponenten von Automatisierungssystemen definiert werden, betrachtet die [IEC_62443-4-1] den gesicherten Entwicklungslebenszyklus (SDL) von Komponenten von Automatisierungssystemen. Dabei wird der gesamten Lebenszyklus einer Komponente von der Produktidee über die Produktentwicklung, die Betriebsphase bis hin zur Außerbetriebnahme des Produktes dargestellt. Besonderer Fokus liegt hierbei bei den Themen Entwurf (engl. Security by Design) und der Umsetzung eines Defense in Depth Konzepts (siehe 5.2.4. Ableitung von Security-Anforderungen). Mit Einhaltung der in der Norm aufgeführten Anforderungen wird sichergestellt, dass Komponenten der IT-Sicherheit, wie sie in der IEC 62443-4-2 beschrieben sind, ordnungsgemäß umgesetzt werden können. Hieraus ergibt sich, dass die Einführung der IEC 62443-4-1 eine zwingende Voraussetzung für die anschließende Umsetzung der IEC 62443-4-2 im Unternehmen darstellt.

Abbildung 2 beschreibt den in IEC 62443-4-1 definierten Produktlebenszyklus. Weiterhin sind verschiedene Kurzbezeichner in den grauen Boxen dargestellt, die die verschiedenen Anforderungsklassen darstellen. Jede Anforderung der IEC 62443-4-1 ist einer Anforderungsklasse zugeordnet.

Bei einer Zertifizierung nach IEC 62443-4-1 können Reifegrade (Maturity Level) von 1 – 4 erreicht werden, wobei 1 den niedrigsten und 4 den höchsten Reifegrad darstellt. Der Reifegrad beschreibt, in welchem Umfang der sichere Entwicklungsprozess im Unternehmen eingeführt und aktuell umgesetzt wird.

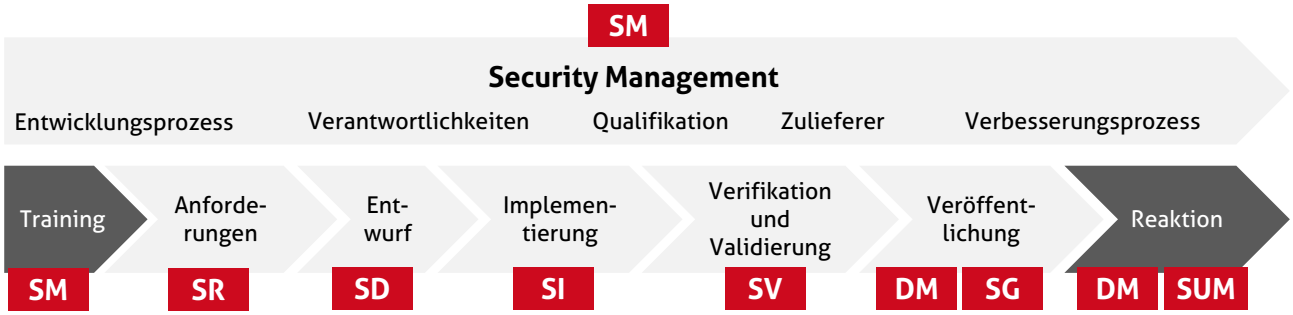

Abbildung 2: Sicherer Entwicklungslebenszyklus, angelehnt an [WAL2020]

4 Inhalt und Struktur der IEC 62443-4-1

Die folgenden Unterkapitel beschreiben den chronologischen Aufbau sowie den jeweiligen Inhalt einzelner Kapitel der [IEC_62443-4-1], orientiert sich dabei an Abbildung 2.

4.1 Security Management (SM)

Das Kapitel „Security Management" definiert den Entwicklungsprozess, die Verantwortlichkeiten und die Qualifikation der Mitarbeitenden. Außerdem werden in diesem Abschnitt Anforderungen an Zulieferer, an einen kontinuierlichen Erfahrungsaustausch und den kontinuierlichen Verbesserungsprozess spezifiziert.

4.2 Security Requirements (SR)

Das zweite Kapitel der Norm beschreibt Sicherheitsanforderungen (engl. Security Requirements). Konkret wird ein Prozess definiert, in dem Anforderungen an das Umfeld und das Produkt während des Betriebs, der Instandhaltung und der Außerbetriebnahme festgelegt und dokumentiert werden. Diese Anforderungen können sich mit der Zeit, die das Produkt im Markt ist, ändern. Dies ist ebenfalls festzustellen und zu dokumentieren.

Weiterhin beschreibt das Kapitel Anforderungen an das Einsatzumfeld hinsichtlich des zu entwickelnden Produktes. In diesem Zusammenhang ist die Durchführung einer umfangreichen Bedrohungsanalyse zu bedenken. Abschließend definiert die Norm die Prozesseinführung bezüglich der Prüfung der geplanten und anschließend umgesetzte Sicherheitsanforderungen, einschließlich der Mängelüberarbeitung.

4.3 Security Design (SD)

Das Kapitel IT-Sicherheit durch den Entwurf (engl. Security Design) beschreibt die konkrete Umsetzung des gesicherten Produktentwurfs. Dies beinhaltet beispielsweise logische und physische Schnittstellen. Weiterhin thematisiert das Kapitel die notwendige Umsetzung eines Defense in Depth Konzepts sowie die Überprüfung und eventuelle anschließende Überarbeitung des Produktentwurfs. Allgemein sind für den gesicherter Produktentwurf bereits bewährte Verfahren und Entwurfsmuster zu verwenden.

4.4 Security Implementation (SI)

In diesem Kapitel wird die gesicherte Implementierung (engl. Security Implementation) beschrieben. Besonderen Fokus legt das Kapitel hierbei darauf, dass die gesicherte Implementierung überprüft und gesicherte Codierungsnormen eingehalten werden.

4.5 Security Verification and Validation (SV)

Das Kapitel Verifikations- und Validierungsprüfungen der IT-Sicherheit (engl. Security Verification and Validation) beschreibt die Einführung von Prozessen, um die eingeführten Sicherheitsanforderungen, Maßnahmen zur Risikominimierung sowie das Prüfen von Sicherheitslücken zu gewährleisten. Um Sicherheitslücken aufzudecken werden beispielsweise Eindringprüfungen durchgeführt. Um den Erfolg dieser Prüfung zu gewährleisten, darf der Entwickelnde nicht gleichzeitig der Prüfende seiner eigenen Entwicklung sein.

4.6 Defect Management (DM)

Dieses Kapitel beschäftigt sich mit der Behandlung von sicherheitsbezogenen Problemen (engl. Defect Management). Hierbei muss ein Prozess erarbeitet werden, in dem Meldungen über ein sicherheitsbezogenes Problem entgegengenommen, anschließend geprüft, behandelt und abschließend über Kanäle, wie beispielsweise eine Website, offengelegt werden. Außerdem beschreibt das Kapitel eine kontinuierliche Überprüfung des Prozesses.

4.7 Security Update Management (SUM)

Die Verwaltung von IT-Sicherheitsupdates (engl. Security Update Management) gewährleistet die Schließung der Sicherheitslücke unter Beachtung der Frage, ob es möglicherweise zu Seiteneffekten auf andere Komponenten gekommen ist. Nach erfolgreichen Tests gilt es das Update zeitnah und über sichere Wege auszuliefern und dies zu dokumentieren.

4.8 Security Guidelines (SG)

Im letzten Kapitel, Kapitel IT-Sicherheitsrichtlinien (engl. Security Guidelines) wird beschrieben, dass spezielle Sicherheitsrichtlinien auszuarbeiten und für Kundinnen und Kunden zu dokumentieren sind. Neben der Defense in Depth-Strategie sind ebenfalls Richtlinien zur Härtung des Systems, zum gesicherten Betrieb, zur Nutzerkontenverwaltung sowie zur gesicherten Entsorgung zu verfassen und Kundinnen und Kunden bereitzustellen.

5 Musterprozess zur IEC 62443-4-1

In den folgenden Unterkapiteln wird ein Musterprozess zur Erstellung eines sicheren Produktentwicklungsprozesses vorgestellt. Orientiert wird sich am chronologischen Ablauf des Gesamtprozesses. Ausgehend von der Produktidee werden die Themen Produktentwicklung, Produktverwendung bis hin zur Phase der Produktaußerdienststellung dargestellt. Der Musterprozess wird anhand eines Embedded System veranschaulicht.

Embedded Systems (dt.: eingebettetes System) sind Computer, die für Anwendenden weitgehend unsichtbar sind. Beispiele aus dem Alltag, in denen Embedded Systems vorhanden sind, stellen Kaffeemaschinen, Fernseher oder Smartphones dar. Aber auch im industriellen Umfeld sind Embedded Systems weit verbreitet, beispielsweise in Maschinen, als speicherprogrammierbare Steuerung oder in Motorsteuerungen sowie im Mobilfunk bei der Übertragung von Daten. Ein Embedded System weist verschiedene Merkmale auf [GES2020]:

1. Embedded Systems führen Funktionen wiederholend aus.
2. Es gibt strikte Vorgaben in Bezug auf Stückkosten, Energieverbrauch und der physischen Abmessungen.
3. Es wird in Echtzeit auf äußere Einflüsse, beispielsweise von Sensoren, reagiert.

5.1 Grundlegende Prozesse

Dieses Unterkapitel beschreibt die ersten Prozesse, die im Zuge der Produktidee und der ersten Betrachtung umzusetzen sind.

Zu Beginn eines jeden Produktes steht die Produktidee. Bereits hier ist zu definieren, welche Fachkenntnisse für die Entwicklung des Produktes erforderlich sein werden. Bei der Entwicklung von Embedded Systems sind beispielsweise Kenntnisse im Bereich der physischen Konstruktion des Systems, beziehungsweise dem Platinenlayout, sowie der Entwicklung von sicherer Software notwendig. Für die Entwicklung von sicherer Software ist zu gewährleisten, dass Eingangsdaten der Software auf Gültigkeit überprüft und Tests Schwachstellen der entwickelten Software aufspüren.

Für den sicheren Produktenwicklungsprozesses eines Embedded Systems werden Fachkenntnisse in den Bereichen Security-Requirements (SR), der Entwicklung eines Defense in Depth Konzeptes, der Durchführung von Risikoanalysen oder der Entwicklung von sicherer Software benötigt. Fachkenntnisse können durch vorhandene Erfahrung, Teilnahme an Konferenzen oder Zertifizierungen nachgewiesen werden. Durchgeführte Schulungen sind im Rahmen der Norm zu dokumentieren, um die vorhandenen Fachkenntnisse gegenüber Zertifizierern nachzuweisen.

Zertifizierer bestätigen als unabhängige Prüfende die Umsetzung von Normen oder branchenspezifischen Standards in Unternehmen. Hierbei werden konkret die Umsetzung von SR auf Konformität überprüft. In Deutschland übernimmt die Aufgabe der IEC 62443-4-1-Zertifizierung unter anderem der TÜV.

Aufgrund der Dokumentation ist das im Unternehmen vorhandene spezielle Knowhow bekannt. Folgende Tabelle 1 zeigt eine mögliche Dokumentationsstruktur für die Planung und Durchführung von Schulungen sowie für notwendige Auffrischungsschulungen.

Name des Mitarbeitenden	Schulungsthema	Datum der Durchführung	Datum der nächsten Auffrischungsschulung
Max Mustermann	Sicher Programmieren – Grundlagen zum Erstellen sicherer Software	01.03.2023	28.02.2025
	Softwarearchitektur für Embedded Systems	16.08.2021	Nicht notwendig
Erika Musterfrau	Durchführen von manuellen und automatischen Penetrationstests	15.02.2020	31.02.2023

Tabelle 1: Planung und Dokumentation von Schulungen

Weiterhin sind bereits in diesem Schritt absehbare verantwortliche Personen über die angehende Produktentwicklung zu informieren, sofern diese bisher noch nicht in den Prozess involviert sind. Hierbei handelt es sich um organisatorische Rollen, die für jeden einzelnen Teilprozess festgelegt und dokumentiert werden müssen. Konkret muss dargelegt werden, wer die Verantwortung eines jeden Prozesses trägt. Dies kann in einem Tabellendokument (siehe Tabelle 2) festgehalten werden. Die Bearbeitung des Dokumentes darf ausschließlich durch autorisiertes Personal erfolgen. Die Verantwortlichen werden in der eigentlichen Produktentwicklung hinzukommen und in die Dokumentation eingepflegt.

Prozess	Prozess-beschreibung	Notwendige Qualifikations-maßnahmen	Verant-wortliche*r	Qualifika-tion vorhanden?	Falls ja, welche?
Sicherer Software-Entwurf	Der Software-entwurf ist nach aktuellen Standards durchzuführen und zu dokumentieren, beispielsweise mithilfe von Unified Modeling Language (UML) Diagrammen.	• Allgemeine IT-Sicherheits-schulung • Schulung zum sicheren Soft-ware-Entwurf • Erfahrung durch bereits durchge-führte Software-Projekte	Max Muster-mann	ja	• Teilnahme an der allgemeinen IT-Sicherheits-schulung • Teilnahme an Schulungen zum sicheren Soft-ware-Entwurf

Tabelle 2: Prozesse und Verantwortlichkeiten

5.2 Produktentwicklung

Nach Erarbeitung einer konkreten Produktidee ist der Produktentwicklungsprozess unter Beachtung folgenden Ablaufs durchzuführen:

1. Beschreibung des Security Kontextes
2. Beschreibung des Defense in Depth Konzeptes
3. Durchführen einer Bedrohungs- und Risikoanalyse
4. Ableitung von Security-Anforderungen
5. Umsetzung der Anforderungen
6. Prüfung der einzelnen Anforderungen, beispielsweise mithilfe von Code Reviews
7. Prüfung von Anforderungen mithilfe von Komponenten- oder Systemtests
8. Erstellen eines Security-Handbuchs, bzw. entsprechender Kapitel in der Produktdokumentation

Die aufgelisteten acht Punkte werden in den folgenden Unterkapiteln dieses Dokumentes weiter detailliert beschrieben.

Generell ist für den sicheren Produktentwicklungslebenszyklus ein Qualitätsmanagementsystem nach [ISO_9001] hilfreich, da hier bereits anerkannte Produktentwicklungsprozesse definiert sind, die ebenfalls im Rahmen der IEC 62443-4-1 gefordert werden. Die ISO 9001 legt Anforderungen an Qualitätsmanagementsysteme fest. Hierzu gehören beispielsweise die Punkte Konfigurationsverwaltung, Erarbeitung und Festlegung von Anforderungen sowie die Unterstützung beim Entwurf, der Implementierung und Prüfung. Eine Umsetzung kann mithilfe von unterstützender Software sowie beschreibenden Dokumenten erfolgen. Diese gilt es anschließend mithilfe der Anforderungen der IEC 62443-4-1 anzupassen und zu erweitern, sodass der gesamte sichere Produktlebenszyklus vom Entwurf im Bereich Software und Hardware bis hin zur Außerbetriebnahme abgedeckt ist.

5.2.1 Beschreibung des Security Kontextes

Die Beschreibung des Security Kontextes legt die Verwendung eines Gerätes durch Kundeninnen und Kunden sowohl in Bezug auf den physischen Verwendungsort als auch hinsichtlich der Softwareschnittstellen fest.

Zunächst muss erarbeitet werden, in welchem Kontext das zu entwickelnde Produkt seitens Kundinnen und Kunden eingesetzt werden soll. Hierfür ist zu spezifizieren, wie die Einsatzumgebung auszusehen hat. So kann bereits hier definiert werden, dass das System, beispielsweise ein Embedded System, während des Betriebs ausschließlich in einem verschlossenen Schaltschrank zu verwenden ist.

Weiterhin wird definiert, wie das Produkt durch Kundinnen und Kunden zu nutzen ist. Konkret wird hier schriftlich festgehalten, welche Daten von Komponenten an das Embedded System gesendet werden und in welcher Art und Weise das Embedded System selber Daten zur Verfügung stellt. Dabei werden beispielsweise Portnummern und Protokolle festgelegt sowie die dafür zu nutzenden physischen Schnittstellen. Außerdem wird definiert, ob das System durch zusätzliche Maßnahmen geschützt werden muss, zum Beispiel durch eine vorgeschaltete Firewall, die ausschließlich für definierte Ports und Protokolle geöffnet ist, die durch das Embedded System benötigt werden und alle weiteren Anfragen automatisch blockiert. Hierdurch wird ein möglicher Angriffsvektor auf ein Minimum reduziert.

Diese Informationen gilt es detailliert bei der Produktausarbeitung zu dokumentieren, da sie als Grundlage für die weiteren Prozessschritte dienen. Hierfür eignet sich die folgende Tabelle 3, die ein Muster für den Security-Kontext eines Embedded Systems darstellt.

Muster: Security-Kontext für ein Embedded System
Für den Betrieb der Embedded System-Produktfamilie XYZ werden folgende Sicherheitsanforderungen für einen ordnungsgemäßen Betrieb benötigt. Diese sind durch den Betreiber kontinuierlich und ohne Pause zu gewährleisten. Andernfalls kann seitens des Herstellers keine Garantie gewährleistet werden.

Physische Voraussetzungen:
1. Das Embedded System darf unter keinen Umständen physisch geöffnet werden.
2. Das Embedded System ist in einem verschlossenen Schaltschrank zu betreiben. Ein Zugang zum Schaltschrank ist ausschließlich autorisiertem Personal zu gestatten.
3. Eine Diagnose-Verbindung ist ausschließlich über den lokalen Diagnoseport durch autorisiertes und geschultes Personal durchzuführen.

Logische Voraussetzungen:
1. Das Embedded System darf ausschließlich über das OPC UA-Protokoll mit dem Kommunikationspartner kommunizieren. Damit keine Alternativkommunikation ermöglicht wird, ist das Embedded System in einem eigenen Netzwerksegment zu betreiben.
2. Für Wartungszwecke darf über den Diagnoseport ausschließlich das Kommunikationsprotokoll SSH verwendet werden.
3. Nach der erfolgreichen Inbetriebnahme muss der Standard-Login durch einen neuen Nutzernamen sowie ein starkes Passwort ausgetauscht werden. Hierzu muss das Passwort aus mindestens 12 Zeichen, Klein- und Großbuchstaben, Ziffern und Sonderzeichen bestehen.
4. Das Embedded System wird durch den Hersteller automatisch über die integrierte Internetverbindung mit aktuellen Sicherheitsupdates versorgt.

Tabelle 3: Muster Security-Kontext Embedded System

5.2.2 Beschreibung des Defense in Depth Konzeptes

Neben der Beschreibung des Security-Kontextes des Embedded Systems gilt es im Kontext der IEC 62443-4-1 ebenfalls das Konzept Defense in Depth zu beschreiben. Es sieht die Kombination von verschiedenen, aufeinander aufbauender Schutzmaßnahmen vor, sodass bei einer Überwindung einer Schutzmaßnahme, weitere andersartige Schutzmaßnahmen greifen. Das Konzept Defense in Depth lässt sich mit einer Burg vergleichen, die mehrere hintereinanderliegende Schutzmaßnahmen aufweist, um ein Erstürmen zu verhindern. Während bei einer Burg nur der physikalische Weg betrachtet wird, können in Bezug auf die IT-Sicherheit sowohl auf physikalischer, als auch technischer und organisatorischer Ebene Schutzmaßnahmen errichtet werden.

5.2.3 Bedrohungs- und Risikoanalyse

Die Bedrohungs- und Risikoanalyse sind zwei Prozessschritte, die nacheinander durchgeführt werden. Zunächst werden mithilfe der Bedrohungsanalyse mögliche Bedrohungen für das System ermittelt, um im Anschluss eine Risikoanalyse durchzuführen.

Bedrohungen sind Möglichkeiten, die das zu entwickelnde System beeinträchtigen, im schlimmsten Fall in Folge eines Cyberangriffs dieses übernehmen. Um Bedrohungen zu identifizieren ist ein Bedrohungsmodell zu entwickeln und kontinuierlich während des Entwicklungsprozesses sowie des Betriebs zu überprüfen. Mithilfe eines Bedrohungsmodells wird das zu entwickelnde System, in diesem Beispiel ein Embedded System, allumfassend betrachtet. Hierzu bietet sich die Verwendung des STRIDE-Modells an. STRIDE steht für:

- Spoofing (Vortäuschung)
- Tampering (Manipulation)
- Repudiation (Abstreitbarkeit)
- Information disclosure (Offenlegung von Informationen)
- Denial of service (Dienstleistungsverhinderung)
- Elevation of privilege (Erhöhung von Rechten)

Ziel des STRIDE-Modells ist es, Bedrohungen herauszufinden. Hierbei werden konkret interne Prozesse des Embedded Systems, Vertrauensgrenzen zwischen dem Embedded System und Komponenten dritter, Datenspeicher, Datenflüsse, interne und externe Kommunikationsprotokolle sowie die physischen Anschlüsse betrachtet [SWI2004]. Im Folgenden ist ein Datenflussdiagramm inklusive Datenspeicher, Prozessen und einer Vertrauensgrenze (Abbildung 3) dargestellt, dass exemplarisch den Datenfluss eines Embedded Systems zeigt. Die Erläuterung der Symbole des Datenflussdiagramms folgen in Tabelle 4 auf der folgenden Seite.

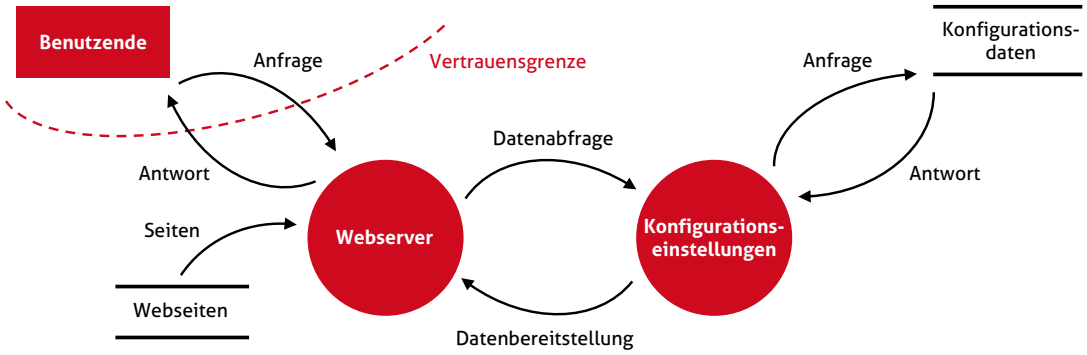

Abbildung 3: Datenflussdiagramm, angelehnt an [CDS2023]

Symbol	Beschreibung
	**Externe Entität:** Externe Instanz, z.B. ein Benutzender oder eine intelligente Funktionseinheit (kommunikationsfähige Steuerung bzw. Sensorik oder ein anderes Subsystem mit geeigneten Schnittstellen).
	**Prozess(e):** Das Prozesssymbol verdeutlicht die Ausführung einer einzelnen Aktion oder einer bestimmten Sequenz von Aktionen zur Datenverarbeitung. Prozesse verändern Daten, die aber grundsätzlich außerhalb des Prozesses gespeichert werden.
	**Datenspeicher:** Ein Symbol für den Datenspeicher repräsentiert den Ort, an dem Daten gespeichert werden. Dabei ist zu beachten, dass hier lediglich Daten gespeichert, aber nicht verändert werden. Für die Veränderung der gespeicherten Daten sind Prozesse verantwortlich.
	**Daten- bzw. Informationsfluss:** Der gesamte Datenfluss wird durch richtungsweisende Pfeile gekennzeichnet. Sie symbolisieren durch die jeweilige Ausrichtung eingehende bzw. ausgehende Datenflüsse. Unabhängig von der Richtung repräsentiert ein Datenflusssymbol auch immer eine Interaktion zwischen zwei Objekten.
	**Vertrauensgrenze:** Das Symbol für die Vertrauensgrenzen separiert in einem Datenflussdiagramm die Zonen mit einem unterschiedlichen Vertrauensgrad. Grundsätzlich symbolisiert eine solche Grenze auch immer Änderungen des Vertrauensgrads und somit der jeweiligen Rechte für eine externe Instanz.

Tabelle 4: Erklärung der Symbole eines Datenflussdiagramms, angelehnt an [CDS2023]

Um mögliche Bedrohungen mithilfe des STRIDE-Modells ermitteln zu können, ist die Dokumentation der dargelegten Prozesse, Datenflüsse, Verwendungsorte des Gerätes, Zugriffsmöglichkeiten und weitere Details von entscheidender Bedeutung. Nur so sind die unterschiedlichsten Bedrohungen zu ermitteln. Mögliche Bedrohungen für ein Embedded System können sein:

1. Feuer
2. Verschmutzung, Staub und Korrosion
3. Ausfall der Stromversorgung
4. Manipulation von Hard- und Software
5. Unbefugtes Eindringen in IT-Systeme
6. Softwareschwachstellen
7. Schadsoftware, z. B. Verschlüsselungstrojaner

Nach der Ermittlung der möglichen Bedrohungen gilt es diese zu bewerten. Hierbei ist es essentiell, dass diese Bewertung über alle Bedrohungen hinweg vergleichbar und nachvollziehbar durchgeführt wird. Um dies zu ermöglichen bietet es sich an, den Ansatz zur Risikoanalyse aus der [IEC_62443-3-2] oder der [VDI/VDE 2182] zu verwenden. Beide Standards beschreiben explizit die Risikoanalyse. Hierbei wird die mögliche Eintrittswahrscheinlichkeit einer Bedrohung mit dem Schadenausmaß, der im Falle des Eintritts entsteht, multipliziert. Als Ergebnis der Multiplikation ergibt sich ein Risikowert. Dieser kann einen Wert zwischen null und neun ergeben. Je höher der Risikowert ist, desto dringender sind Maßnahmen zur Risikominimierung umzusetzen. Das folgende Muster (Tabelle 5) zeigt eine beispielhafte Risikobewertung:

Asset	Bedrohung	Verursacher	Schwachstelle	Schadens-ausmaß [0-3]	Eintrittswahr-scheinlichkeit [0-3]	Risiko	Akzeptanz Risiko [0...9]	Reduzierung erforderlich?
Konfigurations-einstellungen des Embedded System	Unautorisier-ter Zugriff auf Gerät über SSH, Zugriff über vordefiniertes Standard-Pass-wort	Angreifer vor Ort	Standard-Pass-wort ohne Zwang zur Änderung	3	1	3	2	erforderlich

Tabelle 5: Muster Risikobewertung

5.2.4 Ableitung von Security-Anforderungen

Die Ableitung der Security-Anforderungen basiert auf dem Security-Kontext, dem Defense in Depth Konzept sowie der Bedrohungs- und Risikoanalyse und beschreibt allgemein die Security-Anforderungen für das zu entwickelnde Produkt. Konkret wird versucht, mithilfe von Schutzmaßnahmen, den Einfluss von Bedrohungen auf das zu entwickelnde System auf ein akzeptables Niveau zu reduzieren.

Nach der Umsetzung von Schutzmaßnahmen sollte eine erneute Durchführung der Schwachstellenanalyse als Ergebnis einen verminderten Risikowert erreichen. Im Folgenden wird beispielhaft das Ableiten einer Security-Anforderung für ein Embedded System beschrieben.

Im Zuge der Bedrohungsanalyse und der anschließenden Bewertung durch die Risikoanalyse wurde ermittelt, dass das zu entwickelnde Embedded System über eine schwerwiegende Schwachstelle verfügt. In diesem Beispiel ist es möglich, im Falle einer aktiven Verbindung zwischen dem Embedded System und einem Kommunikationspartner, die Herstellersoftware auf dem Embedded System zu manipulieren. Durch eine Manipulation der Software könnte das kompromittierte System zu übermittelnde Werte manipulieren und so Falschinformationen versenden. Um dies zu verhindern ist es wichtig, dass bereits während des Startvorgangs des Systems eine Überprüfung der Herstellersoftware stattfindet, um eine mögliche Kompromittierung der Systemintegrität zu ermitteln. Dies kann mithilfe von Secure Boot erfolgen.

Beim Secure Boot wird Software nur gestartet, wenn diese zuvor nicht kompromittiert wurde. Hierzu wird überprüft ob die Software durch einen kryptografischen Schlüssel signiert wurde. Dieser verwendete kryptografische Schlüssel muss hierfür in einer Firmware-Datenbank des Embedded Systems hinterlegt sein. Diese Datenbank ist explizit auf die Speicherung von kryptografischen Schlüsseln ausgelegt und kann nur durch den Hersteller des Embedded Systems unter Anwendung eines privaten kryptografischen Schlüssels verändert werden. Erst nach einer erfolgreichen Überprüfung der Signatur wird der Startvorgang initiiert. Wurde die Herstellersoftware hingegen kompromittiert ändert sich automatisch die Signatur, sodass eine Überprüfung der Herstellersoftware während des Startvorgangs fehlschlägt und der Start der Software unterbunden wird.

Im Folgenden sind weitere beispielhafte Security-Anforderungen, die keinen Anspruch auf Vollständigkeit erheben, beschrieben, die im Zuge der IEC 62443-4-1 in einem Embedded System umzusetzen sind:

1. Aufbringen eines Siegels auf dem Gehäuse, um zu überprüfen, ob das Embedded System physisch geöffnet wurde. Wurde das Siegel gebrochen, muss davon ausgegangen werden, dass eine Manipulation des Embedded Systems stattgefunden hat.
2. Ein Verbindungsaufbau zur Analyse des Embedded Systems ist ausschließlich mit sicheren Netzwerkprotokollen wie beispielsweise Secure Shell (SSH) zu ermöglichen. Unsichere Netzwerkprotokolle, die keine Verschlüsselung ermöglichen, werden nicht unterstützt.
3. Der Austausch von Daten zwischen Embedded System und Kundinnen und Kunden findet ausschließlich über kryptografisch gesicherte Verbindungen statt.
4. Kundinnen und Kunden sind beim erstmaligen Start des Produkts zum Austausch des Standardpassworts aufzufordern. Für das neue Passwort gilt es Vorgaben einzuhalten, wie beispielsweise Klein- und Großbuchstaben, Zahlen und Sonderzeichen.
5. Das Embedded System ist in einem gehärteten Zustand auszuliefern. Das heißt, dass alle nicht zwingend erforderlichen Dienste zunächst deaktiviert sind. Aktivierung erfolgt lediglich bei Bedarf vom Endanwender.
6. Für den Aufbau einer kryptografisch gesicherten Kommunikation von Embedded System zu Kundinnen und Kunden werden digitale Zertifikate verwendet.

Die ausgearbeiteten Security-Anforderungen sind in einer Security-Requirements-Spezifikation (SRS) zu dokumentieren. Hierbei ist auf Gültigkeit, Nachvollziehbarkeit sowie die Übereinstimmung zum STRIDE-Bedrohungsmodell zu achten. Danach ist die SRS einem Review zu unterziehen. Hierfür werden die folgenden Fachrichtungen benötigt:

- Entwickelnde
- Prüfende
- Kundinnen- und Kundenvertreter
- Sicherheitsberatende

Wenn Komponenten von Dritten im Embedded System verwendet werden, die sich entweder auf die IT-Sicherheit des Produktes auswirken können oder die ausschließlich durch einen Lieferanten für dieses Produkt hergestellt worden sind, gelten für diese ebenfalls die zuvor genannten Kriterien der [IEC_62443-4-1].

Die in diesem Kapitel dargestellten und ermittelten Anforderungen sind im Zuge einer SRS zusammenzufassen. Eine SRS beschreibt beispielsweise Anforderungen an die Betriebsumgebung, die Bedrohungsanalyse und Risikoanalyse des Produktes oder Anforderungen an verschiedene Betriebsmodi und ist somit auch im weiteren SDL immer wieder zu Rate zu ziehen. Entsprechende Verweise sind in diesem Dokument vorhanden. Sofern bereits beschreibende Dokumente existieren, beispielsweise für die Bedrohungsanalyse oder die Risikoanalyse kann im SRS auf diese verwiesen werden. Im Folgenden finden Sie eine beispielhafte Gliederung für eine SRS:

Muster: Gliederung einer Security-Requirement-Specification (SRS)

1. Zweck und grundlegende Beschreibung der SRS
2. Betrachtung des Embedded Systems
3. Sicherheitskontext
   a. Definition des Sicherheitskontextes in Anlehnung an die IEC 62443-4-1
   b. Sicherheitskontext des Embedded Systems
4. Produktsicherheitsanforderungen und -inhalte
   a. Definition der Produktsicherheitsanforderungen und -inhalte in Anlehnung an die IEC 62443-4-1
   b. Produktsicherheitsanforderungen und -inhalte des Embedded Systems
5. Bedrohungs- und Risikoanalyse des Embedded Systems
6. Überprüfung der Sicherheitsanforderungen
   a. Definition der Überprüfung der Sicherheitsanforderungen in Anlehnung an die IEC 62443-4-1
   b. Dokumentation der ersten Überprüfung der Sicherheitsanforderungen des Embedded Systems
7. Zusätzliche Informationen
   a. Auflistung zugehöriger Dokumente
   b. ...
8. Anhang
9. Überarbeitungen des Dokuments

Tabelle 6: Muster Security-Requirement-Specification

5.2.5 Umsetzung der Security-Anforderungen

Nach der Ermittlung von Security-Anforderungen gilt es im Zuge des SDL diese umzusetzen, um bereits während der Entwicklung auf sichere Entwicklungsverfahren zurückgreifen zu können und so zu verhindern, dass sich in diesem Schritt sicherheitskritische Fehler einschleichen, die erst während der Betriebsphase auffallen und zu hohen Kosten in der Fehlerbehebung oder gar zu einem Produktrückruf führen. Als Basis dient hier die zuvor ausgearbeitete SRS (s. Tabelle 6).

Um dies zu ermöglichen ist es wichtig, dass die Entwicklungsumgebung beispielsweise Computer von Entwickelnden oder Prüfenden sowie Server und wichtige Daten und Dateien abgesichert sind. Daten des sich in Entwicklung befindenden Embedded Systems sollten zentral auf einem Server gespeichert werden. Die Daten sollten ausschließlich für autorisiertes Personal erreichbar sein. Dies setzt die Einführung und Pflege eines Berechtigungskonzepts voraus.

Weiterhin sollten regelmäßig verschlüsselte Datensicherungen durchgeführt werden. Diese und weitere Maßnahmen sollen verhindern, das Produktdetails offengelegt werden oder eine Manipulation der zu entwickelnden Software erfolgt. Weitere Maßnahmen werden beispielsweise in der [DIN_EN_ISO_27001] und [DIN_EN_IEC_27002] beschrieben. Eingeführte Maßnahmen müssen sie dokumentieren.

Nach der Einführung eines IT-Sicherheitskonzeptes kann anschließend mit der Entwicklung des Embedded Systems begonnen werden. Hier ist auf Basis der in Kapitel 5.2.4 Ableitung von Security-Anforderungen ermittelten Maßnahmen die Produktentwicklung durchzuführen sowie zu dokumentieren, welche Sicherheitsmaßnahmen im SDL wann umzusetzen sind. Die für die jeweilige Sicherheitsmaßnahme verantwortliche Person ist ebenfalls zu dokumentieren (vgl. Tabelle 2). Um dies umzusetzen bietet es sich an den Ansatz des gesicherten Entwurfs (engl. Security Design) zu verwenden. Mit dessen Hilfe wird gewährleistet, dass das Konzept Defense in Depth, Einzug in das Produkt während des SDL erhält.

Im Folgenden werden Security-Anforderungen aufgelistet die im Zuge des gesamten SDL auf Basis der [IEC_62443-4-1] in einem Embedded System sowie in der Entwicklungsumgebung zu schützen sind:

- Datenbanken und Datenbanktabellen
- Konfigurationsdateien
- Speicher für kryptographische Schlüssel
- Zugriffskontrolllisten (hier wird beschrieben, welche Personen sich in das Embedded System einloggen dürfen)
- Registrierungsschlüssel
- Webseiten
- Auditprotokolle
- Netzwerkanschlüsse
- Die Kommunikation zwischen verschiedenen Prozessen
- Dateien und Verzeichnisse
- Weitere Speicherressourcen

5.2.6 Prüfung der einzelnen Security-Anforderungen

Im Anschluss an die Entwicklung des Embedded Systems sowie der Integration der Security-Anforderungen beginnt die Testphase. Neben Funktionstests gilt es ebenfalls die Security-Anforderungen auf Wirksamkeit und Effektivität zu prüfen. Hierbei ist entscheidend, dass eine Person keine Überprüfung der eigenen Entwicklungen durchführen darf. Eine Prüfung muss immer durch unabhängige Prüfende durchgeführt werden. Optimal wäre es, wenn es hierfür eine eigene Testabteilung geben würde. Dies ist gerade in KMU nicht umsetzbar, sodass hier Kolleginnen und Kollegen die Prüfung durchführen, die bei der Entwicklung des zu testenden Prozesses nicht involviert waren. Als Basis für dieses Kapitel dient die SRS (siehe Tabelle 6).

Prüfung der IT-Security-Anforderungen
Jede einzelne umgesetzte Security-Maßnahme wird beispielsweise auf die Umsetzung der Einführung von Secure Boot, um die Integrität der Herstellersoftware des Embedded Systems gegenüber Manipulation zu schützen, getestet.

Prüfung der Bedrohungsabschwächung
Auf Basis der Risikoanalyse wurden Maßnahmen zur Minderung von Bedrohungen ermittelt. Diese Tests sollen feststellen, ob die jeweils eingeführten Maßnahmen im gewollten Umfang zu einer Abschwächung des Risikos führen.

Prüfung von Sicherheitslücken
Embedded Systems weisen neben der Herstellersoftware ebenfalls ein zusätzliches Betriebssystem auf und ggf. weitere proprietäre Software und diese können ebenfalls Schwachstellen beinhalten. Deswegen gilt es diese, wie auch die Herstellersoftware auf mögliche Sicherheitslücken zu überprüfen. Sicherheitslücken können durch verschiedene Angriffsvektoren ausgenutzt werden. Folgende Prüfungen gilt es auf Basis der [IEC_62443-4-1] durchzuführen:

- Eingaben von unvorhersehbaren oder verfälschten Eingaben beispielsweise auf Webseiten, über Schnittstellen oder die Abfrage von Daten einer Datenbank.
- Die Überprüfung aller physischen und logischen Schnittstellen sowie eine Überprüfung auf mögliche Schwachstellen. Gibt es beispielsweise ungewünschte logische Schnittstellen? Ist ein nicht verwendeter Port offen?
- Durchführen eines Black-Box-Scans. Bei einem Black-Box-Scan werden bekannte Sicherheitslücken in Bezug auf Hardware und Software getestet. Dies kann beispielsweise mithilfe eines Netzwerkscans erfolgen.
- Software muss auf ihre Zusammensetzung hin analysiert werden. Hierbei können verwendete Bibliotheken Schwachstellen enthalten. Als Beispiel sei hier die in vielen Softwareprodukten verwendete Java-Bibliothek log4j genannt, die über eine kritische Schwachstelle verfügt, die mithilfe eines Updates geschlossen werden konnte. Weiterhin ist zu überprüfen, ob die entwickelte Software allumfassend nach vorhandenen Sicherheitsvorschriften entwickelt wurde sowie keine Compiler-Einstellungen verwendet wurden, die zu Sicherheitslücken führen können.

Durchführen von Eindringprüfungen
Bei diesen Tests wird versucht das fertig entwickelte Embedded System mithilfe von verschiedenen Angriffsvektoren zu kompromittieren. Hierzu werden Schwachstellen gesucht und aktiv ausgenutzt. Ziel ist es möglichst tief in das System einzudringen, also mehrere Schutzmaßnahmen des Defense in Depth Konzeptes zu überwinden und so die Schutzziele Vertraulichkeit, Integrität und Verfügbarkeit zu untergraben. Um dies zu erreichen werden Penetrationstest durchgeführt, sowohl manuell als auch automatisiert.

5.2.7 Erstellen eines Security-Handbuchs

Im Rahmen der Umsetzung der IEC 62443-4-1 sind ebenfalls Richtlinien zu verfassen, um Kundinnen und Kunden über das Sicherheitskonzept, über mögliche Konfigurationsmöglichkeiten sowie umzusetzende Schutzmaßnahmen durch Kundinnen und Kunden, zu informieren. Es bietet sich an die umzusetzenden Richtlinien in Form eines Security-Handbuchs oder in entsprechenden Kapiteln der Bedienungsanleitung zusammenzufassen. Im Folgenden werden die umzusetzenden Richtlinien in Form eines Handbuchs thematisiert.

Zunächst wird im Rahmen des Security-Handbuchs die Defense in Depth Strategie des Produktes erläutert, um Kundinnen und Kunden darzulegen, wie das Embedded System durch eine Vielzahl an Maßnahmen geschützt wird. Dieses Konzept, dass sich zunächst nur auf das Produkt bezieht, wird anschließend mit Maßnahmen verknüpft, die in der Umgebung des Embedded Systems umzusetzen sind. Hierbei müssen Kundinnen und Kunden aktiv werden. Erst mit der Implementierung dieser Maßnahmen greift das Gesamtkonzept Defense in Depth. Schutzmaßnahmen für das Produkt werden in Kapitel 5.2.4 Ableitung von Security-Anforderungen und für das Umfeld in 5.2.1 Beschreibung des Security Kontextes genannt.

Weiterhin sind im Handbuch Erläuterungen zu geben, wie das Embedded System während der Installation und des Betriebs gehärtet werden kann. Das bedeutet, dass die jeweilige Funktion der einzelnen zur Verfügung stehenden Sicherheitsfunktionen beschrieben wird, sowie welche Parameter für eine Aktivierung oder Modifikation einzutragen sind.

Eine Begründung im Kontext des Nutzens im Defense in Depth Konzept ist ebenfalls darzulegen. Hier werden die durchzuführenden Maßnahmen aufgelistet, die durch die Umgebung des Embedded Systems umzusetzen sind, damit zum einem das Embedded System durch die Umgebung zusätzlich geschützt wird und zum anderen weitere Komponenten, die für das Embedded System notwendigen Daten nicht blockieren und somit die Funktionsfähigkeit des Systems gefährden. Um Kundinnen und Kunden hierbei als Hersteller zu unterstützen muss eine Empfehlung der vorzunehmenden Konfigurationseinstellungen im Security-Handbuch erfolgen.

Zusätzlich werden Anweisungen erstellt, wie im Falle eines Sicherheitsvorfalls vorzugehen ist, beispielsweise wie der Hersteller Kundinnen und Kunden in einem solchen Fall unterstützen kann.

Wie das Produkt am Ende des SDL außer Betrieb genommen werden soll, wird ebenfalls im Handbuch beschrieben. Folgende Punkte müssen hierbei im Handbuch dargestellt und durch Kundinnen und Kunden beachtet werden:

- In welchen Schritten ist das Embedded System physisch aus der Aktivumgebung zu entfernen.
- Wie können Referenz- und Konfigurationsdaten nachhaltig aus dem Embedded System entfernt werden.
- Wie können sensible, unternehmensspezifische Daten unwiderruflich aus dem Embedded System entfernt werden.
- Wie kann das Embedded System physisch entsorgt werden, sodass die Wiederherstellung von Daten durch unbefugte Dritte unmöglich ist.

Seitens Kundinnen und Kunden muss außerdem eine Verantwortliche Person für die Umsetzung der Sicherheitsmaßnahmen zuständig sein. Falls das nicht der Fall ist, kann es sonst passieren, dass Sicherheitsmaßnahmen durch Anwendende deaktiviert werden, da diese zunächst, bei falscher Konfiguration, bestimmte Funktionen des Embedded System behindern oder verhindern könnten.

Eine mögliche Gliederung für ein Security-Handbuch, die die IEC 62443-4-1 erfüllt, könnte wie folgt aussehen [ABB2023]:

1. Einleitung
2. Anwendbarkeit und Herausforderungen
3. Anforderungen (Verstehen des Produktentwurfs)
4. Wie das Produkt die Anforderungen erfüllt
   a. Inbetriebnahme
   b. Härtung des Systems
   c. Außerbetriebnahme
   d. Defense in Depth Konzept
5. Checkliste für Cyber-Vorfälle
6. Support
7. Glossar
8. Literaturverzeichnis

5.3 Produktverwendung (Betriebsphase)

In dieser Phase des SDL ist das Produkt, ein Embedded System, bereits an Kundinnen und Kunden ausgeliefert und mithilfe des Produkthandbuchs in Betrieb genommen worden. Sicherheitseinstellungen wurden ebenfalls gemäß den Empfehlungen des Security-Handbuchs umgesetzt.

Da Softwareschwachstellen, die entweder in Software Dritter oder des Herstellers unabsichtlich enthalten sind und in der Regel erst mit der Zeit bekannt werden, sollte auch während der Betriebsphase des Produktes ein Support für Sicherheitsfragen gewährleistet sein. Besonderer Fokus liegt auf dem Empfang und der Behandlung von Sicherheitsproblemen (Mängelbehandlung) sowie der Erstellung und Auslieferung von Sicherheitsupdates. Beide Punkte dürfen im Kontext des SDL nicht vernachlässigt werden und sind für eine Zertifizierung nach IEC 62443-4-1 zwingend umzusetzen.

Zusätzlich zu diesem Dokument sind ausführliche Informationen zum Thema Prozesse für die Behandlung von Mängeln und Offenlegung von Mängeln in den Standards [ISO_IEC_29147] und [ISO/IEC FDIS 30111] dokumentiert.

Mängelbehandlung
Die Mängelbehandlung bezeichnet einen Prozess, der mit dem Empfang eines Sicherheitsmangels beginnt und der Behandlung und Offenlegung des Sicherheitsmangels endet.

Empfang einer Mängelmeldung
Zunächst muss der Hersteller eine Möglichkeit schaffen, dass Mängel vertraulich und anonym gemeldet werden können. Hierfür kann beispielsweise eine Funktions-E-Mailadresse eingerichtet werden, die auf der Webseite des Unternehmens sowie im Security-Handbuch unter der Kapitelüberschrift „Support" veröffentlicht wird.

Das Schaffen einer anonyme Meldemöglichkeit ist wichtig, da neben Herstellern von Fremdkomponenten, Behörden, Kundinnen und Kunden sowie der eigenen regelmäßig stattfindenden Überprüfung des Produktes ebenfalls eine Meldung von unbekannten Dritten, wie beispielsweise Sicherheitsforschern erfolgen kann, die sich auf die Überprüfung von mit dem Internet verbundenen Komponenten spezialisiert haben. Der Empfang von neuen Mängelmeldungen muss regelmäßig geprüft werden. Hierfür bietet es sich eine automatische Benachrichtigung zu aktivieren, die erfolgt, wenn ein Mangel gemeldet wird.

Zusätzlich sollten Meldende über den Empfang und die weiteren Bearbeitungsschritte informiert werden. Falls dies nicht geschieht, hat es sich als de facto Standard entwickelt, das eine Schwachstelleninformation nach einer bestimmten Zeit, einer Frist die durch Meldende gesetzt wird, der Öffentlichkeit zur Kenntnis gegeben wird. Dies hat das Ziel das beim Hersteller zusätzlicher Druck und somit Handlungsbedarf entsteht und die Mängelbeseitigung nicht aufgeschoben wird.

Überprüfung der Mängelmeldung
Nach dem Empfang der Mängelmeldung gilt es diese im nächsten Schritt zeitnah zu verifizieren. Hierzu muss der beschriebene Mangel durch den Hersteller an der Produktversion, an der der Mangel gefunden wurde, getestet werden. Weiterhin muss getestet werden, ob der ermittelte Mangel ebenfalls in anderen Produkten zu finden ist, beispielsweise in einer Vorgängerversion oder der

Nachfolgeversion eines Embedded Systems. Handelt es sich bei der Überprüfung um einen Mangel, ist mit dem nächsten Schritt fortzufahren. Wurde hingegen kein Mangel festgestellt oder ermittelt, dass dieser Mangel durch eine falsche Konfiguration durch Kundinnen und Kunden eingetreten ist, gilt es die Person zu informieren, die den Mangel gemeldet hat, vorausgesetzt, die Person möchte dies und hat den Mangel nicht anonym gemeldet.

Bewertung des ermittelten Mangels
Nach der erfolgreichen Überprüfung des gemeldeten Mangels gilt es im nächsten Schritt eine Bewertung des Mangels am Embedded System durchzuführen. Hierbei muss konkret ermittelt werden, welchen Einfluss der Sicherheitsmangel auf das Defense in Depth Konzept des Produktes sowie die Komponenten im Umfeld des Produktes aufweist. Weiterhin ist die Grundursache des Problems sowie mögliche Seiteneffekte auf weitere Komponenten und Software herauszufinden, die hierdurch ebenfalls betroffen werden können.

Hierzu ist neben der Bedrohungsanalyse durch das STRIDE-Modell ebenfalls eine Bewertung des Schweregrads der Bedrohung mithilfe der Risikoanalyse (siehe Kapitel 5.2.3 Bedrohungs- und Risikoanalyse) für den ermittelten Sicherheitsmangel vorzunehmen. Dies gilt ebenfalls für alle Produkte und Produktversionen, die über diesen Sicherheitsmangel verfügen.

Behandlung sicherheitsbezogener Mängel
Nach der Bewertung des ermittelten Mangels folgt im nächsten Schritt die Behandlung. Diese sollte durch die Einspielung eines Sicherheitsupdates durchgeführt werden. Sofern das Aufspielen eines Sicherheitsupdates nicht möglich ist, stehen in Einzelfällen folgende Optionen zur Verfügung:

- Beheben des Mangels durch das Defense in Depth Konzept
- Die Anforderungen an das Produktumfeld werden erhöht, um den Angriffsvektor, der den Mangel bedroht, zu reduzieren
- Einführen von weiteren Schutzmaßnahmen, die ausschließlich den Sicherheitsmangel reduzieren
- Deaktivieren von Funktionen (Härtung) des Systems, in dessen Folge der Angriffsvektor des Sicherheitsmangels ebenfalls reduziert oder deaktiviert wird
- Nichtbehebung des Mangels, da das vorhandene Sicherheitsrisiko als gering eingestuft wird

Im Falle einer Beseitigung des Mangels ist ein Plan zu erstellen, in welchem Zeitraum die Behebung des Mangels zu erfolgen hat und in welchen Schritten hierbei vorzugehen ist.

Der Hersteller sollte Vorgehen und Maßnahmen bei der Behebung von Sicherheitsmängeln detailliert dokumentieren, damit im Falle einer wiederholten Behebung eines Mangels ein Best Practice vorhanden ist, dass ganz oder in Teilen bei der Beseitigung des neuen Sicherheitsmangels unterstützen kann. Eine allgemeine Dokumentation ist ebenfalls während des gesamten Prozesses durchzuführen, um im nächsten Schritt, der Offenlegung von Sicherheitsmängeln, Informationen an Kundinnen und Kunden weitergeben zu können.

Mängel offenlegen
Im Zuge des Prozesses der Behebung von Sicherheitsmängeln gilt es meldepflichtige und sicherheitsbezogene Mängel offenzulegen und Kundinnen und Kunden zu informieren. Informationen die Kundinnen und Kunden in diesem Kontext erhalten müssen sind Folgende:

- Eine detaillierte Problembeschreibung
- Die Mängelbewertung durch die Risikoanalyse
- Eine detaillierte Lösungsbeschreibung mit Hinweisen zur Installation von Sicherheitsupdates oder durchzuführenden Konfigurationseinstellungen

Unterstützung durch ein CERT
Um Mängel zu erkennen, zu bewerten und offenzulegen gibt es Unterstützungsmöglichkeiten. So ist es möglich hierzu ein Computer Emergency Response Team (CERT) zu verwenden. Es besteht die Möglichkeit entweder ein eigenes CERT aufzubauen und zu betreiben oder dies auszugliedern. Ein CERT hat den Vorteil, dass es sich hierbei um eine zentrale Anlaufstelle für Betreiber, Integratoren und Herstellern von Produkten handelt und diese sich untereinander austauschen können.

Zusätzlich werden hierüber anonymisiert Schwachstellen entgegengenommen sowie eine zeitnahe Bearbeitung durchgeführt. Gerade, wenn mehrere Unternehmen in einem CERT gebündelt werden, die häufig miteinander kooperieren, entsteht durch die zentrale Sammlung, Entgegenahme, Überprüfung und Bearbeitung von Mängeln ein Vorteil.

Der gesamte Prozess bis auf die Teilschritte der Bearbeitung von Sicherheitsmängeln kann zentralisiert durchgeführt werden. Da die Norm [IEC_62443-4-1] einen Prozess zur Mängelbehandlung fordert und dies eine zwingende Voraussetzung für eine Zertifizierung des SDL darstellt, bietet sich die Einführung von CERT an. Für Industrieunternehmen stellt der Verband der Elektrotechnik Elektronik Informationstechnik e.V. das CERT@VDE als zentralisierte Dienstleistung zur Verfügung. Für Bundesbehörden wird durch das Bundesamt für Sicherheit in der Informationstechnik das CERT-Bund angeboten.

Regelmäßige Überprüfung des Prozesses
In regelmäßigen Abständen gilt es auf Basis der ermittelten Best Practices und neuen Impulsen, beispielsweise durch Veröffentlichungen und dem Austausch mit anderen Unternehmen, den gesamten Prozess, von eingehenden Informationen eines Mangels bis hin zur Mängelbehandlung und Offenlegung des Prozesses, zu überprüfen. So werden bewährte Prozessschritte weitergeführt und ineffiziente Prozessschritte angepasst oder ausgetauscht.

Erstellung von Sicherheitsupdates
Wurde im Zuge der Bewertung und Behandlung eines Mangels oder mehrerer Mängel ermittelt, dass diese mithilfe eines Sicherheitsupdates zu schließen sind, ist nach der Erstellung des Sicherheitsupdates, dieses auf seine Eignung hin zu testen. Konkret muss überprüft werden, ob die Sicherheitslücken tatsächlich mit der Installation des Sicherheitsupdates geschlossen werden.

Außerdem muss überprüft werden, ob es nicht zu Seiteneffekten in der eigenen Software oder bei Komponenten von anderen Herstellern kommt, die in Verbindung mit dem Embedded System stehen. Es muss ausgeschlossen werden, dass das Sicherheitsupdate negative Auswirkungen beim Produkt bzw. in Verbindung stehenden Produkten hervorruft.

Auch während der Erstellung des Sicherheitsupdates gilt es, eine Dokumentation zu erstellen, die bei der Auslieferung von Sicherheitsupdates an die Produktbetreiber zu versenden ist. Diese muss folgende Punkte beinhalten:

- Die Produktversionsnummern, für welche das Sicherheitsupdate gilt
- Anweisungen, wie das Sicherheitsupdate manuell oder automatisiert installiert werden kann
- Eine Beschreibung möglicher Auswirkungen auf das Produkt, beispielsweise das nach einer erfolgreichen Installation ein Neustart des Produkts durchzuführen ist
- Mögliche Sicherheitsrisiken die entstehen, sofern das Sicherheitsupdate nicht installiert wird

Sofern das Sicherheitsupdate neben dem eigenen Embedded System ebenfalls Komponenten Dritter betrifft, ist dies zu dokumentieren. Konkret muss dargelegt werden, ob es nach der Installation des Sicherheitsupdates zu Kompatibilitätsproblemen mit anderen Komponenten kommen kann.

Auslieferung von Produktupdates
Die Auslieferung eines Produktupdates ist ein kritischer Ablauf, da hier Daten sicher über das Internet vom Hersteller zum Betreiber des Embedded Systems transportiert werden müssen. Es ist eine Manipulation sowohl auf Herstellerseite als auch auf Betreiberseite auszuschließen. Trotz dieser Herausforderungen gilt es ein Sicherheitsupdate zeitnah und sicher auszuliefern.

Um dies zu ermöglichen bietet es sich an, das auf Herstellerseite ein Server zur Verfügung steht, der die Updates zentralisiert für die Embedded Systems bereitstellt. Der Server muss mithilfe von Schutzmaßnahmen gewährleisten, dass ausschließlich lesend über ein definiertes Protokoll auf die Updates zugegriffen werden kann.

Zusätzlich sollte das zu aktualisierende Gerät sich vorher beim Server authentifizieren, um seine Echtheit bzw. die Legitimität der Anfrage nachzuweisen. Dies kann beispielsweise mithilfe von Login-Daten und Passwort oder über ein Zertifikat überprüft werden. Bei einem Zertifikat handelt es sich um einen Echtheitsnachweis, der durch eine Zertifizierungsstelle (engl. Certification Authority), dem Hersteller des Embedded Systems ausgestellt wird.

Nach der erfolgreichen Anfrage dürfen die Dateien des Sicherheitsupdates ausschließlich über eine verschlüsselte Verbindung vom Server zum Embedded System gesendet werden. Um eine Echtheit der Daten, die Integrität gewährleisten zu können, sind die Sicherheitsupdates digital zu signieren.

5.4 Produktaußerdienststellung

Wird ein Produkt, nachdem es viele Jahre oder Jahrzehnte aktiv war, außer Dienst gestellt, gilt es analog zu den vorangegangen Prozessschritten des SDL, folgende Punkte durch den Hersteller zu definieren.

Um ein Embedded System, das beispielsweise als Gateway zwischen Betreiberservern, die sich im Internet befinden und einer Produktionsanlage fungiert, außer Dienst zu stellen, ist zunächst darzulegen, wie das Produkt auf richtige Art und Weise aus der Produktivumgebung entfernt werden kann. Es ist darzustellen wie das System, zunächst heruntergefahren und anschließend physisch entfernt werden kann. Im Anschluss sind Dateien, die sich auf dem Embedded System befinden, zu entfernen. Dies gilt neben unternehmensspezifischen Daten ebenfalls für Konfigurationseinstellungen. Weiterhin sind die vorhandenen kryptografischen Schlüssel durch den Hersteller aus dem Produkt zu exportieren oder alternativ der sichere Datenbankspeicher physisch zu zerstören, beispielsweise, in dem dieser mithilfe eines Bohrers durchbohrt wird.

5.5 Zusammenfassung

Der vorliegende Musterprozess gibt einen Überblick über die wichtigsten Aspekte bei der Einführung eines gesicherten Entwicklungslebenszyklus im Unternehmen, wie er in der [IEC_62443-4-1] gefordert wird. Es ist wichtig mit den grundlegenden Prozessen wie beispielsweise der Fortbildung der Mitarbeitenden zu beginnen. Anschließend folgt der sichere Produktentwicklungsprozess mit dem Fokus auf die Beschreibung des Security Kontextes, die Beschreibung des Defense in Depth Konzeptes sowie die Bedrohungs- und Risikoanalyse. Auf dieser Basis werden im Anschluss Security-Anforderungen ermittelt, die anschließend umgesetzt und geprüft werden. Während der Betriebsphase sind Schwachstellen, die durch Dritte gemeldet werden, entgegenzunehmen und zu bearbeiten. Der sichere Entwicklungslebenszyklus endet schlussendlich mit der Außerbetriebnahme des Produktes. Neben der Beschreibung des Musterprozesses sind im Dokument ebenfalls Abbildungen, Texte und Tabellen integriert, die bei der Einführung der IEC 62443-4-1 unterstützen und verwendet werden können.

6 Verzeichnisse

6.1 Abbildungsverzeichnis

6.2 Tabellenverzeichnis

6.3. Literaturverzeichnis

[ABB2023]
ABB AG: White Paper - Cyber Security in the AC500 PLC family. Approach Cyber Security with Confidence. URL: https://library.e.abb.com/public/dae33e547f084eb-29cba2b65fad74065/White%20Paper%20-%20AC500%20Cyber%20Security.pdf, 30.03.2023.

[CDS2023]
Larry Conklin, Victoria Drake, Sven strittmatter: Threat Modeling Process. URL: https://owasp.org/www-community/Threat_Modeling_Process, 27.03.2023.

[DIN_EN_IEC_27002]
Deutsches Institut für Normung e. V.: DIN ISO/IEC 27002:2017 Informationstechnik - Sicherheitsverfahren - Leitfaden für Informationssicherheitsmaßnahmen, 2017.

[DIN_EN_ISO_27001]
Deutsches Institut für Normung e. V.: DIN ISO/IEC 27001:2017 Informationstechnik - Sicherheitsverfahren - Informationssicherheitsmanagementsysteme - Anforderungen (ISO/IEC 27001:2013 einschließlich Cor 1:2014 und Cor 2:2015), 2017.

[DKE2022]
DKE - Deutsche Kommission Elektrotechnik: Elektronik Informationstechnik DIN und VDE EC 62443: Die internationale Normenreihe für Cybersecurity in der Industrieautomatisierung. URL: https://www.dke.de/de/arbeitsfelder/industry/iec-62443-cybersecurity-industrieautomatisierung, 21.11.2022.

[GES2020]
Gessler, Ralf: Entwicklung eingebetteter Systeme. Vergleich von Entwicklungsprozessen für FGPA- und... Mikroprozessorsysteme Entwurf auf systemebene. Morgan Kaufmann, [S.l.], 2020.

[GMO2018]
Gunter, David G.; Medoff, Michael D.; O'Brien, Patrick C.: Implementing IEC 62443. A pragmatic approach to cybersecurity. Exida, Sellersville, PA, 2018.

[IEC_62443-1-1]
IEC - International Electrotechnical Commission: IEC TS 62443-1-1:2009 In-dustrial communication networks - Network and system security - Part 1-1.

[IEC_62443-1-2]
IEC - International Electrotechnical Commission: ISA-TR 62443-1-2 Security for industrial automation and control systems - Master Glossary.

[IEC_62443-1-3]
IEC - International Electrotechnical Commission: IEC/TS 62443-1-3 Security for industrial process measurement and control – Network and system security – Part 1-3.

[IEC_62443-1-4]
IEC - International Electrotechnical Commission: ISA-62443-1-4 Security for industrial automation and control systems Life Cycle and Use Cases.

[IEC_62443-2-1]
IEC - International Electrotechnical Commission: IEC 62443-2-1:2010 Industrial communication networks - Network and system security - Part 2-1.

[IEC_62443-2-2]
ISA - The International Society of Automation: ISA-62443-2-2 Security for industrial automation and control systems - Part 2-2.

[IEC_62443-2-3]
IEC - International Electrotechnical Commission: IEC TR 62443-2-3:2015 Security for industrial automation and control systems - Part 2-3.

[IEC_62443-2-4]
IEC - International Electrotechnical Commission: IEC 62443-2-4 Security for industrial automation and control systems – Network and system security – Part 2-4.

[IEC_62443-2-5]
IEC - International Electrotechnical Commission: IEC 62443-2-5 Implementation guidedance for IACS asset owners.

[IEC_62443-3-1]
IEC-International Electrotechnical Commission: Industrial communication networks - Network and system security - Part 3-1.

[IEC_62443-3-2]
IEC - International Electrotechnical Commission: IEC 62443-3-2:2018 Security for industrial automation and control systems - Part 3-2.

[IEC_62443-3-3]
IEC - International Electrotechnical Commission: ndustrial communication networks - Network and system security - Part 3-3.

[IEC_62443-4-1]
IEC - International Electrotechnical Commission: IEC 62443-4-1:2018 IT-Sicherheit für industrielle Automatisierungssysteme - Teil 4-1.

[IEC_62443-4-2]
DKE - Deutsche Kommission Elektrotechnik, Elektronik Informati-: DIN EN IEC 62443-4-2 IT-Sicherheit für industrielle Automatisierungssysteme - Teil 4-2.

[ISO/IEC FDIS 30111]
ISO - International Standardization Organization; IEC - International Electrotechnical Commission: ISO/IEC FDIS 30111:2019(E): Information technology — Security techniques — Vulnerability handling processes.

[ISO_9001]
ISO - International Standardization Organization: ISO 9001:2015 Quality management systems — Requirements.

[ISO_IEC_29147]
ISO - International Standardization Organization; IEC-International Electro-technical Commission: ISO/IEC 29147:2018(E): Information technology — Security techniques — Vulnerability disclosure.

[NIE2021]
Niemann, Karl-Heinz: Abgrenzung der IT-Sicherheitsnormenreihen ISO 27000 und IEC 62443: Eine Sicht auf automatisierungstechnische Anlagen der Fertigungs- und Prozessindustrie. Hochschule Hannover, 2021.

[SWI2004]
Swiderski, Frank; Snyder, Window: Threat modeling. Microsoft Press, Redmond, Wash., 2004.

[VDI/VDE 2182]
VDI/VDE-Gesellschaft Mess- und Automatisierungstechnik: VDI/VDE 2182 Blatt 1. Informationssicherheit in der industriellen Automatisierung, 2011.

[WAL2020]
Waldeck, Boris: Zertifizierter Entwicklungsprozess nach 62443-4-1 – Security by design, Lemgo, 2020.

Kontakt für Rückfragen

Jan-Niklas Puls
Experte für IT-Sicherheit

0511 9296 1629
puls@mitunsdigital.de
www.digitalzentrum-hannover.de

Mittelstand-Digital
Zentrum
Hannover