Bundesverband IT-Sicherheit e.V.

TeleTrusT
Pioneers in IT security.

Secure Platforms für Digitale Souveränität

Technische Empfehlungen für das europäische IT-Ökosystem

2021

Danksagung

Diese Publikation wurde im TeleTrusT-Arbeitskreis "Secure Platform", einem Untergremium der TeleTrusT-AG "Recht", erarbeitet. TeleTrusT bedankt sich bei den nachstehenden Personen für ihre Mitwirkung sowie für die aktive Mitgestaltung dieser Handreichung.

Projektleitung
Dr. André Kudra, esatus, Leiter der TeleTrusT-AG "Blockchain" und Leiter des TeleTrusT-AK "Secure Platform"

Autorenliste (Auszug)

Adolf, Alexander - Condition-ALPHA
Blunk, Rolf - OTARIS
Jäger, Dr. Hubert - Digital Trust Innovations
Kolmhofer, Prof. DI Robert - FH Hagenberg
Kudra, Dr. André - esatus
Mutz, Reinhard - World Privacy and Identity Association
Olfert, Sarah - esatus
Pfeiffer, Anna Katharina - esatus
Rieken, Dr. Ralf - Uniscon
Schwemm, Adrian - Institut für Internet-Sicherheit

Redaktion

Abou Nasser, Morad - Bundesverband IT-Sicherheit e.V. (TeleTrusT)
Mühlbauer, Dr. Holger - Bundesverband IT-Sicherheit e.V. (TeleTrusT)

In dieser Publikation werden zahlreiche Anglizismen verwendet, da sie sich in der zugrundeliegenden Fachdiskussion branchentypisch verfestigt haben.

Dieses Dokument dient als Anhaltspunkt und bietet einen Überblick. Er erhebt weder Anspruch auf Vollständigkeit noch auf die exakte Auslegung der bestehenden Rechtsvorschriften. Er darf nicht das Studium der relevanten Richtlinien, Gesetze und Verordnungen ersetzen. Desweiteren sind die Besonderheiten der jeweiligen Produkte sowie deren unterschiedliche Einsatzmöglichkeiten zu berücksichtigen. Insofern sind bei den im Dokument angesprochenen Beurteilungen und Vorgehensweisen eine Vielzahl weiterer Konstellationen denkbar.

Impressum

Herausgeber:

Bundesverband IT-Sicherheit e.V. (TeleTrusT)
Chausseestraße 17
10115 Berlin
Tel.: +49 30 400 54 310
Fax: +49 30 400 54 311
E-Mail: info@teletrust.de
https://www.teletrust.de

V 1.0_2021-06 DE

Inhalt

1 Überblick

Die Forderung nach *Digitaler Souveränität* für Deutschland und Europa ist inzwischen in der geschäftlichen und politischen Mainstream-Diskussion angekommen. Im April 2020 hat der Bundesverband IT-Sicherheit e.V. (TeleTrusT) durch seinen Arbeitskreis "Secure Platform", ein Untergremium der TeleTrusT-AG "Recht", sein Positionspapier *"Secure Platform: Digitale Souveränität als Motivation für ein sicheres IT-Ökosystem in Deutschland und Europa"* mit Forderungen an Entscheidungsträger aus Politik und Wirtschaft sowie Hersteller von IT- bzw. IT-Sicherheitsprodukten veröffentlicht. Angekündigt wurde dort die nächste Phase mit einem technischen Schwerpunkt: Die Erarbeitung einer architekturellen Empfehlung für sichere Plattformen, die für die Ausgestaltung kritischer IT-Komponenten angewendet werden sollte.

Das vorliegende Papier ***"Secure Platforms für Digitale Souveränität: Technische Empfehlungen für das europäische IT-Ökosystem"*** ist das Ergebnis. Es ist ein umfassendes Werk - in Breite und Tiefe - geworden, um der Problemstellung adäquat Rechnung zu tragen. Ausgangspunkt ist weiterhin die Förderung Digitaler Souveränität durch die **Stärkung bzw. Schaffung eines europäischen IT-Ökosystems**, um **manipulationssichere Verarbeitung** (Kapitel 2.1) technisch sicherzustellen. Es baut auf dem etablierten **IT-Schichtenmodell** (Kapitel 2.2) auf und erweitert es gezielt. Ein **Angriffsmodell** (Kapitel 4.1) mit staatlichen Akteuren, Lieferanten, Betreibern und externen Angreifern wird darübergelegt, um konkrete Anknüpfungspunkte für technische Empfehlungen abzuleiten.

Grundlegende Erkenntnis unter Berücksichtigung von Standardisierungsaktivitäten (Kapitel 3.1), risikoorientierten Beschaffungsentscheidungen (Kapitel 3.2) und "Security by Design"-Prinzipien (Kapitel 3.3) ist, dass im Ökosystem nicht überall die höchsten **Sicherheitskategorien** (Kapitel 4.2):

I. Robuste Versorgung mit mehreren unabhängigen Lieferanten je Komponente,
II. Verifizierbar manipulationssichere Technik,
III. Unabhängigkeit der Auditoren,

erreicht werden können. Eine Experteneinschätzung liefert im Ideal erreichbare Sicherheitskategorien pro Angriffsvektor und den erforderlichen Umsetzungsaufwand (Kapitel 5.1). Die Grobschätzung ergibt die wenig überraschende Aussage, dass **mehrere Tausend Personenjahre Arbeit** erforderlich sind, um das europäische Ökosystem zu erschaffen. Positiv: Es wird bei entsprechender Priorisierung und Fokussierung als **grundsätzlich möglich** erachtet.

Ausgehend von den wünschenswerten **Zieleigenschaften einer Architektur** von Secure Platforms (Kapitel 5.2) werden die **Empfehlungen für die technische Architektur** (Kapitel 6) abgeleitet. Einstiegspunkt dafür sind Entwurfsmuster und die Bestätigung der prinzipiellen Machbarkeit (Kapitel 6.1) mit anschließender Ableitung der **Referenzarchitektur** für die technischen Empfehlungen (Kapitel 6.2). Die **Entwurfsempfehlungen** (Kapitel 6.3) werden konkretisiert für:

- Gehäuse, Leiterplatten und Verbindungstechnik
- Rechner-Systemarchitektur
- Prozessortechnik
- Netzwerktechnik
- Betriebssystem
- Virtualisierung

Konkrete Zielvorstellung des Arbeitskreises im Sinne nächster Schritte sind **Secure Platform-Referenzimplementierungen** (Kapitel 7.1), die mit staatlicher Förderung anzugehen sind und die Secure Platform Machbarkeit grundsätzlich bestätigen. Derartige Referenzimplementierungen sind als Grundlage der Digitalen Souveränität anzusehen und müssen umgehend in die kommerzielle Entwicklung und Bereitstellung überführt werden. Der dringlichste Schritt (Kapitel 7.2) sind dann industrielle Deployments mit tatsächlicher **produktiver Nutzung der Technologien des eigenen Ökosystems**. Ein wesentlicher "weicher" Erfolgsfaktor ist die Absenkung des "zu großen Respekts" vor der Aufgabe. Trotz der zu erwartenden Anstrengungen ist es unvermeidlich, die europäischen Fähigkeiten in CPU-Design und Herstellung zu stärken, um die Digitale Souveränität nachhaltig sicherzustellen.

2 Digitale Souveränität durch ein eigenes europäisches IT-Ökosystem

Informationstechnik (IT) ist fest in jedem Bereich des Alltags verankert und Grundlage der Digitalisierung. "Digitale Souveränität" ist eine entscheidende Vorbedingung für die Wettbewerbsfähigkeit Europas, gerade auch mit Blick auf Industrie 4.0, aber ebenso für den Betrieb Kritischer Infrastrukturen (KRITIS) oder Vorhaben wie GAIA-X und 5G. Im ersten Positionspapier "Secure Platform: Digitale Souveränität als Motivation für ein sicheres IT-Ökosystem in Deutschland und Europa" des TeleTrusT-Arbeitskreises "Secure Platform" des Bundesverbandes IT-Sicherheit e. V. (TeleTrusT) vom April 2020 wurde festgehalten, dass diese Digitale Souveränität aus mehreren Gründen nicht gegeben ist:

1. Es bestehen starke Abhängigkeiten von auswärtigen Technologieproduzenten.
2. Eingesetzte Technologien sind komplexe, intransparente Systeme.
3. Nachhaltige IT-Sicherheit scheint kein Entscheidungskriterium zu sein.

Es wurde festgestellt, dass es weder realistisch, notwendig noch wünschenswert ist, alle Elemente einer digitalen Infrastruktur von europäischen Herstellern entwickeln und fertigen lassen zu wollen. Eine erfolgversprechende und zielführende Strategie ist jedoch, ausgewählte Themenfelder durch europäische Anbieter zu besetzen. Die Wiedererlangung Digitaler Souveränität kann erreicht werden durch:

- Reduktion digitaler Abhängigkeit von außereuropäischen Anbietern,
- Abkehr von unzureichenden Designparadigmen, sowie
- Begünstigung eines verantwortungsbewussteren Käuferverhaltens.

Das vorliegende Positionspapier widmet sich vorrangig der technischen Diskussion und der Ableitung von Empfehlungen gemäß der Auflösung unzureichenderer Designparadigmen (Punkt 2) sowie perspektivisch der Stärkung bzw. Wiederansiedelung eines europäischen IT-Ökosystems im Sinne der Reduktion von Abhängigkeiten zu außereuropäischen Anbietern (Punkt 1).

2.1 Zielsetzung: Manipulationssichere Verarbeitung technisch sicherstellen

Digitale Souveränität bedeutet, dass Deutschland und die EU die wesentlichen Werte und Rechte von Bürgern, Behörden und Wirtschaft eigenständig schützen und wahren können. Digitale Souveränität bedeutet auch, dass Behörden und Unternehmen digitale Dienste souverän nutzen können, d.h. mit hoher Zuverlässigkeit die vollständige und dauerhafte Kontrolle über die Datenverarbeitung nicht nur auf dem Papier, sondern auch materiell behalten können. Ein konstituierender Faktor Digitaler Souveränität ist die manipulationssichere Verarbeitung von Daten. Auf Käufer- bzw. Anwenderseite werden Vertrauenswürdigkeit und Sicherheit von IT fälschlicherweise als Selbstverständlichkeit angesehen. Ein starker Schutz gegen unerwünschte Manipulation ist häufig kein primäres Designziel von heutigen IT-Systemen:

- Design und Produktion aktueller Systeme sind durch die Paradigmen "maximale Funktionalität" und "geringe Time-to-Market" geprägt.
- Aktuelle IT-Systeme sind vielschichtige Gebilde aus Hard- und Software, deren Komplexität selbst für Experten oft schwer durchdringbar ist.
- Es ist kaum feststellbar, ob ein Produkt lediglich die gewünschten Funktionalitäten bereitstellt oder ob Schwachstellen und versteckte "Back Doors" vorhanden sind.
- Risiken der Lieferkette wie bspw. mögliche Einflussnahmen auf IT-Komponenten in der Kette werden nicht analysiert und als Entscheidungsgrundlage einbezogen.
- Ist ein Produkt erst einmal auf dem Markt, werden sicherheitsrelevante Updates vernachlässigt.

Fakt ist: Das allgemeine Sicherheitsniveau und die Vertrauenswürdigkeit aktuell verbreiteter Systeme sind deren gesellschaftlicher Bedeutung selten angemessen. Übersetzt in die klassischen Schutzziele der Informationssicherheit bedeutet dies:

- Die Vertraulichkeit (Confidentiality) ist beeinträchtigt, vertrauliche Daten gelangen in unerwünschte Hände und können im Sinne des Angreifers, bspw. Durch Industriespionage, oder auch spezifisch gegen den Urheber, bspw. durch Erpressung, eingesetzt werden.
- Die Integrität (Integrity) ist beeinträchtigt, Daten werden, schlimmstenfalls unbemerkt, verändert und beeinflussen die Geschäftsausübung oder führen durch falsche Betriebsparameter zur Beschädigung bzw. Zerstörung physischer Infrastrukturen.
- Die Verfügbarkeit (Availability) ist beeinträchtigt, Daten werden extrahiert bzw. vernichtet oder Systeme unerreichbar bzw. abgeschaltet (Kill Switch), ebenso ist Kontrollverlust mit Steuerungsübernahme möglich.

Die CIA-Triade - Vertraulichkeit, Integrität, Verfügbarkeit - beeinträchtigende Manipulationen können insbesondere durch systemische Schwachstellen, Programmierfehler, eingebaute Hintertüren und Fernsteuerungsmechanismen sowie eventuelle Überwachungsfunktionen ermöglicht werden. Diese können sich praktisch auf jeder Ebene des IT-Systems befinden, auch in der Hardware, d.h. den verbauten Chips. Durch fehlende Manipulationssicherheit bzw. -detektion bestehen an verschiedenen neuralgischen Punkten in IT-Systemen bemerkenswerte Kompromittierungspotenziale, die im Kapitel 4 dezidiert aufgeschlüsselt werden. Auf dieser Basis werden mit der in Kapitel 3 beschriebenen Grundsätzen sicherheitstechnischer Gestaltung, die Anforderungen an eine sichere technische Architektur in Kapitel 5 beschrieben. Dabei wird die aktuelle Situation der idealen Wunschvorstellung gegenübergestellt und in Kapitel 6 hinsichtlich der Mach- und Umsetzbarkeit kritisch untersucht. Dabei werden Möglichkeiten aufgezeigt, wie eine manipulationssichere Datenverarbeitung im Sinne aller Komponenten der CIA-Triade erreicht werden kann. In Kapitel 7 werden dringliche Maßnahmen als auch Forderungen an die Bundesregierung formuliert. Das Kapitel 8 rundet mit einem Exkurs in die Konzeption eines sicheren Endanwenderarbeitsplatzes das Positionspapier ab.

2.2 Im Fokus: Backend-Komponenten des IT-Schichtenmodells

Im ersten Positionspapier wurde ein einfaches IT-Schichtenmodell definiert sowie analytisch und argumentativ verwendet. Abbildung 1 zeigt in die wesentlichen Elemente einer digitalen Infrastruktur und deren Herkunft.

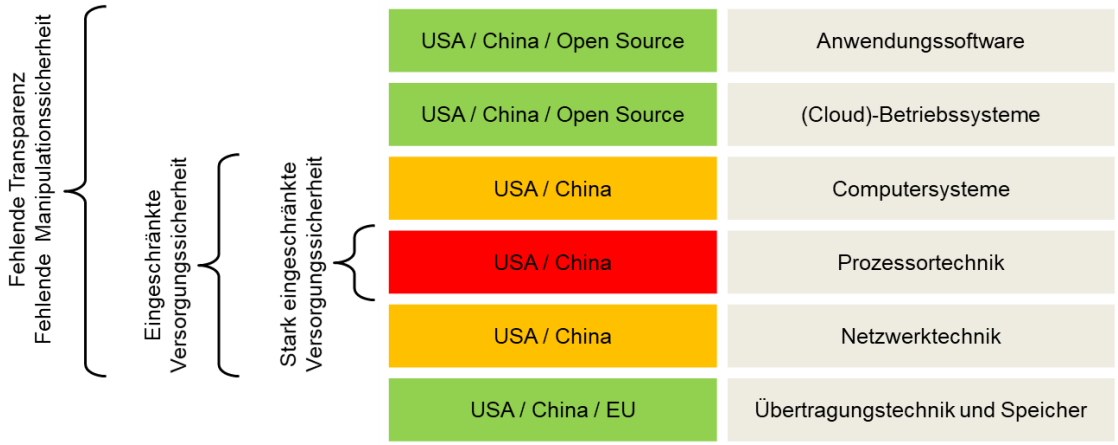

Abbildung 1: IT-Abhängigkeiten im internationalen Kontext

Darin wird deutlich, dass aktuelle Abhängigkeiten der Digitalen Souveränität entgegenstehen:

- Eine vollständige Abhängigkeit von außereuropäischen Lieferanten besteht bei Netzwerktechnik, Prozessortechnik und Computersystemen.
- In der Übertragungstechnik gibt es wettbewerbsfähige europäische Angebote.
- Bei Betriebssystemen und Anwendungssoftware bietet Open Source realistische Alternativen zu den dominanten Anbietern.

Diese Abhängigkeiten führen im Rahmen der fortschreitenden Digitalisierung und Vernetzung von Wirtschaft, Verwaltung und Gesellschaft zu signifikanten Risiken hinsichtlich des Schutzes der wesentlichen Werte der Digitalen Souveränität und somit zu wachsenden Risiken bezüglich der Gewährleistung der unabhängigen Funktion des öffentlichen Lebens und der Wirtschaft. Da die Abhängigkeit Deutschlands und der EU von außereuropäischen Lieferanten über Jahrzehnte gewachsen ist, sind Lösungen erforderlich, die mit tragbarem

Aufwand innerhalb eines Zeitraums von wenigen Jahren zu einer signifikanten Verbesserung führen. Folgende Umsetzungsoptionen, siehe auch Abbildung 2, werden als realistisch erachtet:

1. Förderung des Einsatzes von Open Source Software in allen Bereichen von Wirtschaft und Gesellschaft.
2. Auswahl und Förderung alternativer Angebote in den Bereichen Netzwerktechnik und Computersysteme inklusive der Schlüsselkomponenten aus europäischen Quellen, für welche die Bereitstellung aus den Quellen USA und China als nicht ausreichend zuverlässig bewertet wird.
3. Förderung von Entwicklung und Anwendung von Konzepten und Lösungen zur Realisierung von Transparenz und Manipulationssicherheit bestehender Systeme mittels automatischer Systeme für separate Kontrolle (Auditierung) und Steuerung (Confidential und Zero Trust Computing).

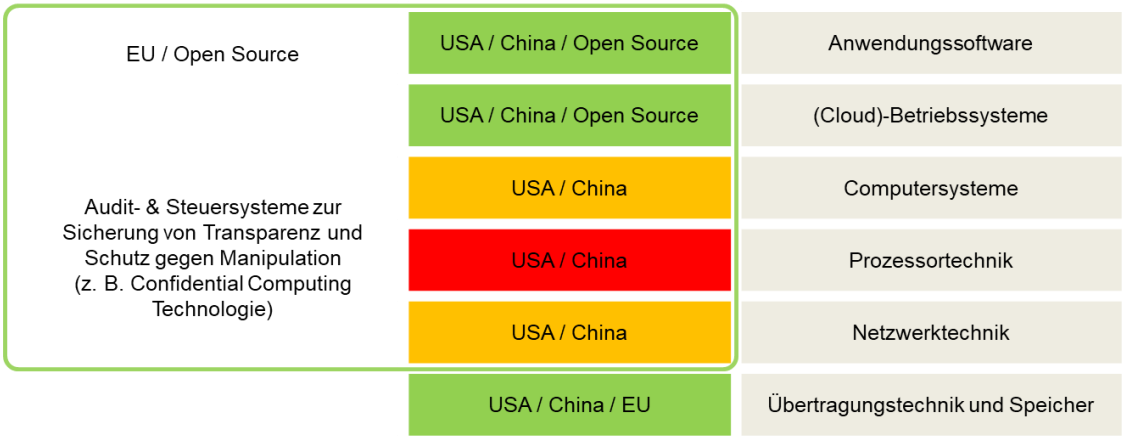

Abbildung 2: Umsetzungsoptionen zur Reduzierung der Abhängigkeit

Unter Berücksichtigung dieser Umsetzungsoptionen und des IT-Schichtenmodells spricht der TeleTrusT-Arbeitskreis Secure Platform in diesem zweiten Positionspapier architekturelle Empfehlungen aus. Thematisch begrenzt werden diese auf Backend-Komponenten in Servern (Prozessoren, Systeme, Betriebssysteme) plus Netzwerktechnik, wie diese bspw. für die Implementierung von KRITIS-Komponenten, GAIA-X Knoten und allgemein Cloud-Infrastrukturen benötigt werden.

3 Grundsätze sicherheitstechnischer Gestaltung

3.1 Standardisierungsaktivitäten für sichere Plattformen

Für die Gestaltung von sicheren Plattformen sind sowohl Sicherheits- als auch Datenschutzstandards relevant. Bei den Sicherheitsstandards werden zum Standard ISO/IEC 27001 für Informations-Sicherheits-Managementsysteme mit seinem normativen Anhang A und den Umsetzungsempfehlungen in ISO/IEC 27002 laufend neue "Codes of Practice" für spezifische Teilaspekte hinzugefügt. Für das Cloud-Computing ist das beispielsweise der ISO/IEC 27017 und unter Berücksichtigung des Datenschutzes der ISO/IEC 27018. Ein bekanntes Profil für diese Standardreihe ist der IT-Grundschutz des Bundesamt für Sicherheit in der Informationstechnik (BSI), der auch auf den Aufbau und den Betrieb von Plattformen angewendet werden kann.

Für Prüfungen und Testierungen entsprechend der Vorgehensweise von Wirtschaftsprüfern gemäß dem "International Standard on Assurance Engagements" (ISAE) 3000 bzw. dem mit diesem in Einklang stehenden deutsche Prüfungsstandard (PS) 860 "IT-Prüfung außerhalb der Abschlussprüfung" des Instituts der Wirtschaftsprüfer (IDW) eignet sich der BSI C5 Standard (Cloud Computing Compliance Criteria Catalogue), der auf Anforderungen der zuvor genannten ISO-Standards sowie von weiteren amerikanischen Standards, u.a. der Cloud Controls Matrix der Cloud Security Alliance aufbaut.

Die Standardisierungen der Sicherheit im Bereich von Industrie 4.0 beziehen sich in der Hauptsache auf die Reihe IEC 62443 zur Cyber-Security in der Industrieautomatisierung. Im Fokus steht die Verbesserung der Integrität und Verfügbarkeit von Komponenten und Systemen in der Industrieautomatisierung. In IEC 62443-3-3 werden bereits "Security Levels SL-1, SL-2 und SL-3" definiert. Im Rahmen der Standardisierungstätigkeiten der "International Data Space Association" (IDSA) wird aufbauend auf IEC 62443 mit der DIN Spec 27070 der Fokus auf die Vertraulichkeit als Schutzziel ausgedehnt und Schutzprofile (Basis, Trust und Trust Plus) definiert. Das Profil "Trust Plus" verlangt nicht nur Schutz gegen versehentliche Fehlbedienungen durch Administratoren, sondern darüber hinaus Schutzmaßnahmen gegen vorsätzliche Manipulationen des Systems durch Administratoren, die gleichzeitig verhindern, dass Manipulationen durch externe Angreifer vorgenommen werden können, die Administratorenzugangsdaten erbeuten (Protection against Theft of Privilege).

Da die Umsetzung der zuvor genannten ISO/IEC-Empfehlungen nur im Rahmen einer Zertifizierung oder Testierung eines Informations-Sicherheits-Managementsystems bestätigt werden können, hat für die Zwecke einer Datenschutz-Zertifizierungen gemäß dem Artikel 42 der EU-Datenschutzgrundverordnung (DSGVO) das vom Bundesministerium für Wirtschaft und Energie (BMWi) geförderte Projekt AUDITOR einen eigenen Prüfstandard für Datenverarbeitungsvorgänge mit Cloud-Computing ausgearbeitet. Dieser wird als DIN Spec 27557 dokumentiert. Eine solche Datenschutz-Zertifizierung hat den Vorteil, dass die Cloud-Nutzer ihre Kontrollpflichten rechtsverbindlich als erfüllt betrachten können, sofern sie einen Cloud-Anbieter mit einem solchen Zertifikat auswählen. Das Zertifikat kennt drei Schutzklassen (1 bis 3). Wenn ein Cloud-Nutzer den Schutzbedarf, der durch ihn mit Hilfe der Cloud verarbeiteten Daten feststellt, und einen Cloud-Dienst auswählt, der diese Schutzklasse erfüllt, dann ist ein solches Datenschutz-Zertifikat nicht nur ein Gütesiegel, sondern ein Zertifikat mit verbindlicher Rechtsfolge. Die DIN Spec 27557 fordert im Bereich der Zugriffskontrolle (Kriterium 2.4) für die höchste Schutzklasse 3, dass die Zugriffskontrolle regelmäßig durch manipulationssichere technische Maßnahmen zur Prävention und aktiven Erkennung von Angriffen zu erfolgen hat. Weiter erläutert die DIN Spec 27557, dass technische Maßnahmen dann manipulationssicher sind, wenn sie nur durch das Zusammenwirken von mehreren unabhängigen Parteien verändert werden können.

Die in Unterkapitel 2.1 formulierte Zielsetzung einer technisch-architektonischen Empfehlung für den souveränen Aufbau und Betrieb einer sicheren Plattform baut auf den bestehenden und den gerade entstehenden Standards auf, geht aber über diese noch hinaus, da für Souveränität nicht nur hohe und höchste Vertraulichkeit und Integrität, sondern auch Offenheit und Transparenz sowie Austauschbarkeit von Komponenten durch Standardisierung und Modularität gefordert sind.

3.2 Risikoorientierte Entscheidung für IT-Produkte auf Basis von Schutzbedarf

Grundsätzlich erfolgt im Organisationskontext bei der Beschaffung von IT-Produkten - implizit oder explizit - eine risikoorientierte Entscheidung. Auch wenn kein formalisierter Risikomanagementprozess existiert: Es

herrscht bei Entscheidern oft die einseitige Einschätzung, dass im Handel bzw. aus "seriösen Quellen" verfügbare Produkte schon sicher sein werden. Für den "formellen Weg" sind aus der Standardisierung für das Risikomanagement Best Practice Frameworks wie der BSI-Standard 200-3 "Risikoanalyse auf der Basis von IT-Grundschutz" oder die ISO/IEC 27005 "Information technology - Security techniques - Information security risk management" hervorgegangen. Diese kommen in der Praxis regelmäßig zur Anwendung, insbesondere in größeren Organisationen. Elementare Komponenten solcher Risikomanagementprozesse sind

- eine Schutzbedarfsfeststellung für relevante IT-Objekte bzgl. der Grundwerte Vertraulichkeit, Integrität und Verfügbarkeit (CIA-Triade) als Grundvoraussetzung,
- eine Gefährdungsanalyse, aus der eine Übersicht der möglichen Gefährdungen hervorgeht,
- eine Risikoeinstufung durch die Ermittlung von Eintrittshäufigkeit (bzw. Eintrittswahrscheinlichkeit, Likelihood) und Schadenshöhe (bzw. Schadensauswirkung, Impact),
- eine Vorgehensweise für die Risikobehandlung gemäß des Risikoappetits der Organisation (bspw. Vermeidung, Reduktion, Transfer, Akzeptanz).

Das vorliegende Werk beinhaltet eine elaborierte Auseinandersetzung mit Gefährdungen auf einem für die gängige Praxis eher unüblichen Niveau. Aus der Erfahrung der IT-Sicherheitsexperten des Bundesverbandes geht eindeutig hervor, dass die Risiken, die von den Gefährdern

- staatlicher Akteure (eigener oder fremder Staat),
- Lieferanten von Hard- und Software,
- Plattformbetreiber, die Privilegien missbrauchen können, und
- hochqualifizierter Akteure aus organisierter Kriminalität oder Terrorismus

ausgehen, meist gezielt "wegignoriert" werden. Die Wahrscheinlichkeit, dass etwas von diesen Ebenen passiert, wird als sehr gering eingeschätzt bzw. als "Da kann man sowieso nichts machen!" abgetan. Dieses Papier begibt sich beabsichtigt auf den ungewöhnlichen Tiefgang der detaillierten Betrachtung und Analyse. Damit wird einerseits die strategische Gefährdung verdeutlicht. Andererseits werden konkrete Maßnahmen zur Gefahrreduktion bzw. Stärkung der europäischen Position vorgeschlagen und somit aufgezeigt, dass und in welchem Rahmen doch "etwas" machbar ist.

Dabei geht es nicht nur um die strategisch wichtige Digitale Souveränität, sondern auch um die für den technischen Datenschutz notwendigen Maßnahmen bei Datenverarbeitungstätigkeiten mit der höchsten Schutzbedarfsklasse wie etwa Verarbeitungstätigkeiten besonders schutzwürdiger Daten oder Verarbeitungstätigkeiten großer Datenbestände (KI/Big Data), mit denen Persönlichkeitsprofile erstellt werden können.

3.3 Referenz zu Security-by-Design-Prinzipien und Anforderungen

Entsprechend der unter Kapitel 2 aufgeführten Zielsetzung und der Voraussetzungen für eine angemessene IT-Souveränität in Deutschland und der EU stellt sich auf der Grundlage der in Kapitel 3 und in 4.1 tiefergehend beschriebenen Gefährdungen und Voraussetzungen die Frage, ob und inwieweit die - u.a. auch vom BSI in den letzten Jahren vielfach geforderte - strikte Einhaltung der Security-by-Design-Prinzipien uns dem Ziel der IT-Souveränität näher bringen kann.

In den oben bereits erwähnten Kapiteln 2 und 3 werden sowohl die relevanten IT-Schichten mit ihren jeweiligen Systemelementen und den sie verbindenden Kommunikationswegen als auch derzeit mögliche Angriffsmodelle allgemein und ohne spezifische Bezugnahme auf branchen- bzw. technologiespezifische Details hinreichend beschrieben.

Die konsequente Einhaltung der Security-by-Design-Prinzipien - d.h. die Berücksichtigung von Anforderungen der Informationssicherheit und des Datenschutzes von Anfang an und bis zum Ende des jeweiligen Lebenszyklus von Produkten, Dienstleistungen und Prozessen zu jedem Zeitpunkt und aus der Sicht aller relevanten Stakeholder bestmöglich und integrativ - ist in einem "Sicheren Lebenszyklus-Modell" gemäß der generischen Darstellung im nachfolgenden Schaubild möglich:

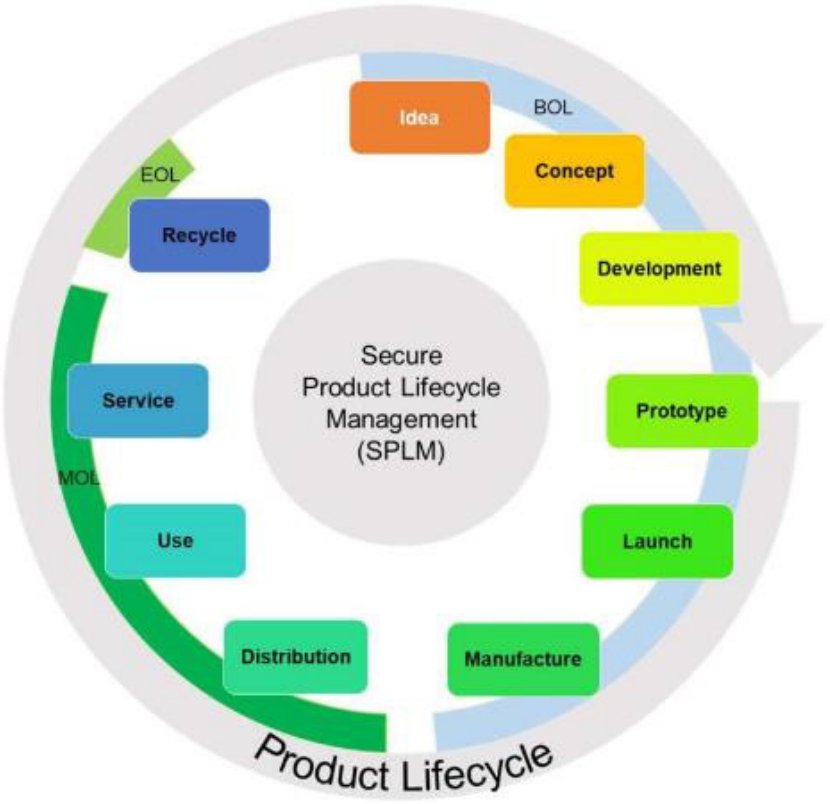

Abbildung 3: Secure Product Lifecycle Management[1]

Laut der im August 2020 veröffentlichten TeleTrusT-Handreichung "Security by Design - Leitfaden für Entscheider" hat sich im Rahmen eines solchen sicheren Lebenszyklus-Modells in der Praxis die Durchführung eines Bedrohungsmodellierungs-Workshops (Threat Modeling) gemeinsam mit allen relevanten Stakeholdern während der frühen Ideen- bzw. Konzept-Phase als eine qualitätsentscheidende Kernmaßnahme erwiesen.

Als Ergebnis eines solchen Bedrohungsmodellierungs-Workshops liegen spezifisch dokumentierte und von allen Stakeholdern mitgetragene Schutzziele und als geeignet identifizierte Sicherheits- und Datenschutzmaßnahmen (gemäß jeweils aktuellem Stand der Technik) als jederzeit zu überprüfende Soll-Vorgaben für den gesamten weiteren Lebenszyklus verbindlich vor.

Das grundlegende Vorgehen in einem Security-by-Design konformen Bedrohungsmodellierungs-Workshop zeigt das nachfolgende Schaubild:

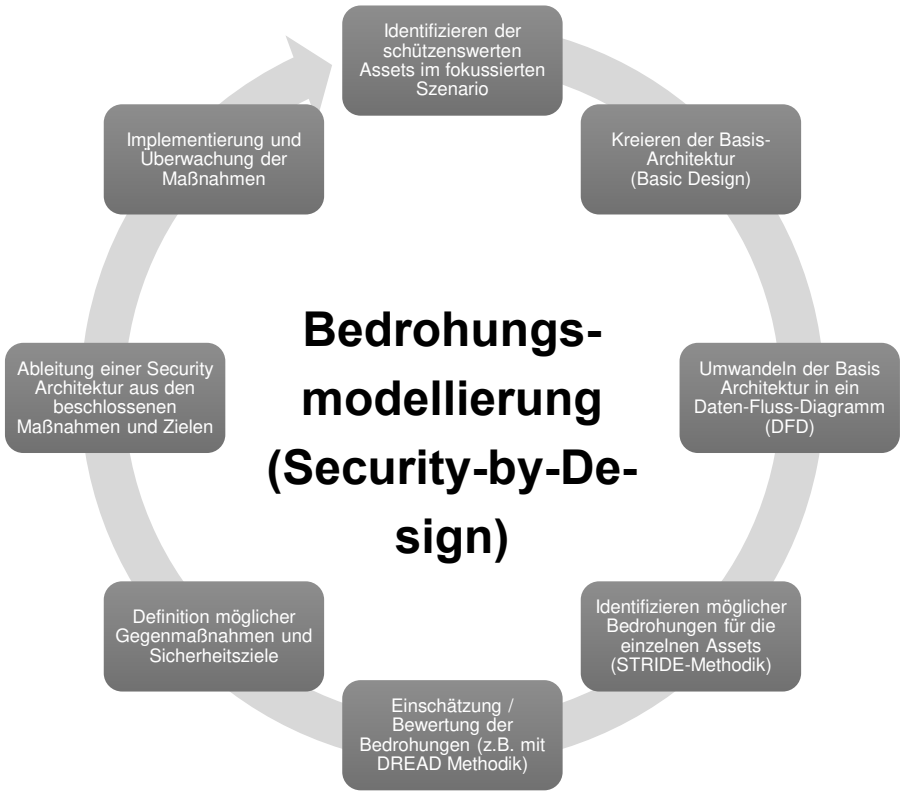

Abbildung 4: Bedrohungsmodellierung nach Security by Design

Angewandt werden entsprechende Security-by-Design Vorgehensmodelle bislang beinahe ausschließlich in ausgewählten Software-Entwicklungs-Lebenszyklen. Um den oben beschriebenen Nutzen daraus auch für die angestrebte zunehmende "Digitale Souveränität durch ein eigenes europäisches IT-Ökosystem" zu erreichen, sollte das Prozess-Vorgehen gemäß Security-by-Design "von Anfang an" auch bereits beim Design und der Auswahlentscheidung für IT-System-Elemente (IT-Schichtenmodell) einschließlich der Prozessor-Architekturen und Betriebssysteme konsequent angewandt werden.

Hierzu ist das beschriebene Security-by-Design Vorgehen beginnend mit einer Bedrohungsmodellierung in der Concept-Phase auch auf die einzelnen Ebenen und Elemente des IT-Schichtenmodells (siehe Kapitel 2.2) anzuwenden und durchzuführen.

Gegenwärtige Beschaffungsprozesse für IT-Systemelemente und Infrastrukturkomponenten (einschl. Prozessor-Architekturen und Operating Systems) werden in privatwirtschaftlichen Unternehmungen und in der öffentlichen Verwaltung in erster Linie nach Wirtschaftlichkeitskriterien und Funktionalitäten ausgeschrieben und entschieden.
Zukünftig sind zur Erreichung der Souveränitäts-Zielsetzungen vielmehr auch die im Security-by-Design Vorgehen zu Beginn erarbeiteten Sicherheits- und Datenschutz-Anforderungen als zwingend notwendige Vorgaben von den Anbietern zu verlangen und nur angemessen "sichere" IT-Komponenten zu beschaffen und einzusetzen. Von Seiten der Politik sind hierfür die Rahmenbedingungen - in erster Linie für Betreiber kritischer Infrastrukturen, ebenso aber auch u.a. für Anbieter und Betreiber von IoT-Systemen (Internet of Things) und Lösungen - verbindlich vorzugeben.

In einer weiteren Dokument-Version des TeleTrusT Arbeitskreises Security-by-Design ist die Beschreibung von ausgewählten Szenarien zu spezifischen Branchen- und Technologie-Umfeldern derzeit in Bearbeitung. Die Veröffentlichung der Folgeversion des Security-by-Design Dokumentes ist für das laufende Jahr 2021 geplant.

Wichtig erscheint hierbei nach ersten Recherchen und Erhebungen, neben international tätigen Großindustrie-Unternehmungen sowohl innovative kleine und mittelständische Unternehmen (KMUs) - z.B. in spezifischen

industriellen IoT-Anwendungsbereichen und in der Fertigungsautomation - als auch öffentliche Verwaltungen mit Bürger-Services auf Bundes- und Landesebene mit einzubeziehen.

Die Ergebnisse sollten in Form von noch zu identifizierenden Studienergebnissen den Entscheider-Ebenen in der Politik als Grundlage für die Erarbeitung und Entscheidung angemessener Rahmenbedingungen zur Erreichung der angestrebten digitalen IT-Souveränität dienen. Außerdem sollten auf dieser Basis hochinnovative Forschungseinrichtungen und international wettbewerbsfähige Wirtschaftsunternehmen innerhalb Deutschlands und der EU mit kurz-/mittel- und langfristigen Zielszenarien etabliert werden.

4 Angriffsmodell und Sicherheitskategorien

4.1 Angriffsmodell und Einschränkungen der Souveränität

"High Performance Computing" ist der Dreh- und Angelpunkt der Digitalisierung. Die schnell wachsenden und flexibel nutzbaren Rechenplattformen (Computing Platforms) machen die Softwareentwicklung und die Bereitstellung von Software as a Service (SaaS) relativ einfach und schnell. Insbesondere die durch die von großen Plattform-Anbietern, den so genannten "Hyperscalern", über das bloße "Hosting" hinaus bereitgestellten proprietären Daten- und Infrastrukturdienste, etwa Datenbankdienste oder Redundanzstrukturen, erleichtern den Software-Ingenieuren die Arbeit und verkürzen die Produktentstehungszyklen für SaaS-Angebote signifikant.

Allerdings locken die Plattform-Anbieter auf diese Weise ihre Kunden, die SaaS-Anbieter, in eine Falle. Sobald die Software der SaaS-Anbieter u.a. auf den proprietären Daten- und Infrastrukturdiensten aufbauend etabliert sind, können die SaaS-Anbieter nicht ohne große Migrationskosten zu einem anderen Plattform-Anbieter wechseln. Sie verfangen sich im so genannten "Vendor Lock-in". So verfestigt sich im Moment ein weltumspannendes Oligopol der großen Plattform-Anbieter mit der entsprechenden Verschiebung der Gewinnmargen von den SaaS-Anbietern hin zu den Plattform-Anbietern.

Die Stärke der Hyperscaler resultiert in erster Linie aus dem gebündelten Angebot der Computing Platform, der Systemsoftware, der Daten- und Infrastrukturdienste und teilweise sogar der Software-Dienste (SaaS) aus einer Hand. Dabei handelt es sich bei der Systemhardware teilweise um eigene Assemblierungen der Hyperscaler, bei der Systemsoftware, den Daten- und Infrastrukturdiensten und der "Userland Software-Dienste" jeweils um eine Mischung von Open Source Software und eigenentwickelter proprietärer Software.

Für die Software-Entwickler und SaaS Provider, deren Aufgaben häufig ineinandergreifen und daher oft gemeinsam mit dem Begriff "DevOps" bezeichnet werden, ist es viel einfacher, auf die gebündelten Angebote der Hyperscaler zurückzugreifen, anstatt die Dienste und die Infrastruktur auf nackten Hosting-Leistungen aufzubauen und mit den notwendigen Prozessen selber zu betreiben.

Jedoch auf der anderen Seite, auf der Seite der Hyperscaler und deren kleineren Konkurrenten, ist die Bereitstellung der Plattform-Dienste eine komplexe Aufgabe mit entweder einer hohen Wertschöpfungstiefe oder vielen Abhängigkeiten von Lieferanten, und den damit einher gehenden Einschränkungen der Souveränität. Diese Komplexität bringt zudem eine Fülle von Angriffsmöglichkeiten mit sich. In diesem Kapitel werden die einzelnen Abhängigkeiten und Angriffsmöglichkeiten jeweils kurz erläutert.

In Abbildung 5 sind rund um eine zentral abgebildete Computing Platform die Lieferanten, Kunden und Nutzer, sowie potenziell eingreifende externe Angreifer oder staatliche Akteure angeordnet. Hier ist noch nicht die in dieser Handreichung vorgeschlagene Secure Platform und ihre Rolle im Ökosystem einer Daten- und Dienste-Infrastruktur, wie etwa GAIA-X abgebildet, sondern vorerst eine herkömmlich gestaltete und betriebene Computing Platform und das von Hyperscalern gebildete Bündel aus Rollen und Funktionen. In Kapitel 6 wird zur Darstellung der zusätzlichen Maßnahmen zur Sicherung der Plattform auf dieselbe Abbildung aufgebaut und die Secure Platform sowie ihre Rolle im Ökosystem einer offenen und föderierten Daten- und Dienste-Infrastruktur beschrieben.

Die lilafarbigen Linien bezeichnen die Lieferanten-Kunden-Verhältnisse. Die Angriffsvektoren sind als gestrichelte Pfeile angedeutet und zum Referenzieren im Weiteren mit einer Buchstaben-Nummern-Kennzeichnung versehen.

Die Wertschöpfungskette beginnt in Abbildung 5 links oben bei den "Processor Chip Manufacturers", die ihre Chips an die Hersteller der Server, Speicherelemente und Netzelemente wie Router und Switches (System Hardware Providers) liefern. Diese beliefern die komplett zusammengebaute Hardware im Falle von Servern und Speicherelementen i.d.R. zusammen mit einem Basic Input/Output System (BIOS), im Falle von Netzelementen i.d.R. mit der kompletten Betriebssoftware an die "Platform Provider".

Zum Betrieb der Serververbunde und Speichersysteme ist zusätzlich umfangreiche Systemsoftware erforderlich, die teilweise proprietär und teilweise quelloffen, d.h. als Open Source Software vorliegt und von den Platform Providern genutzt wird.

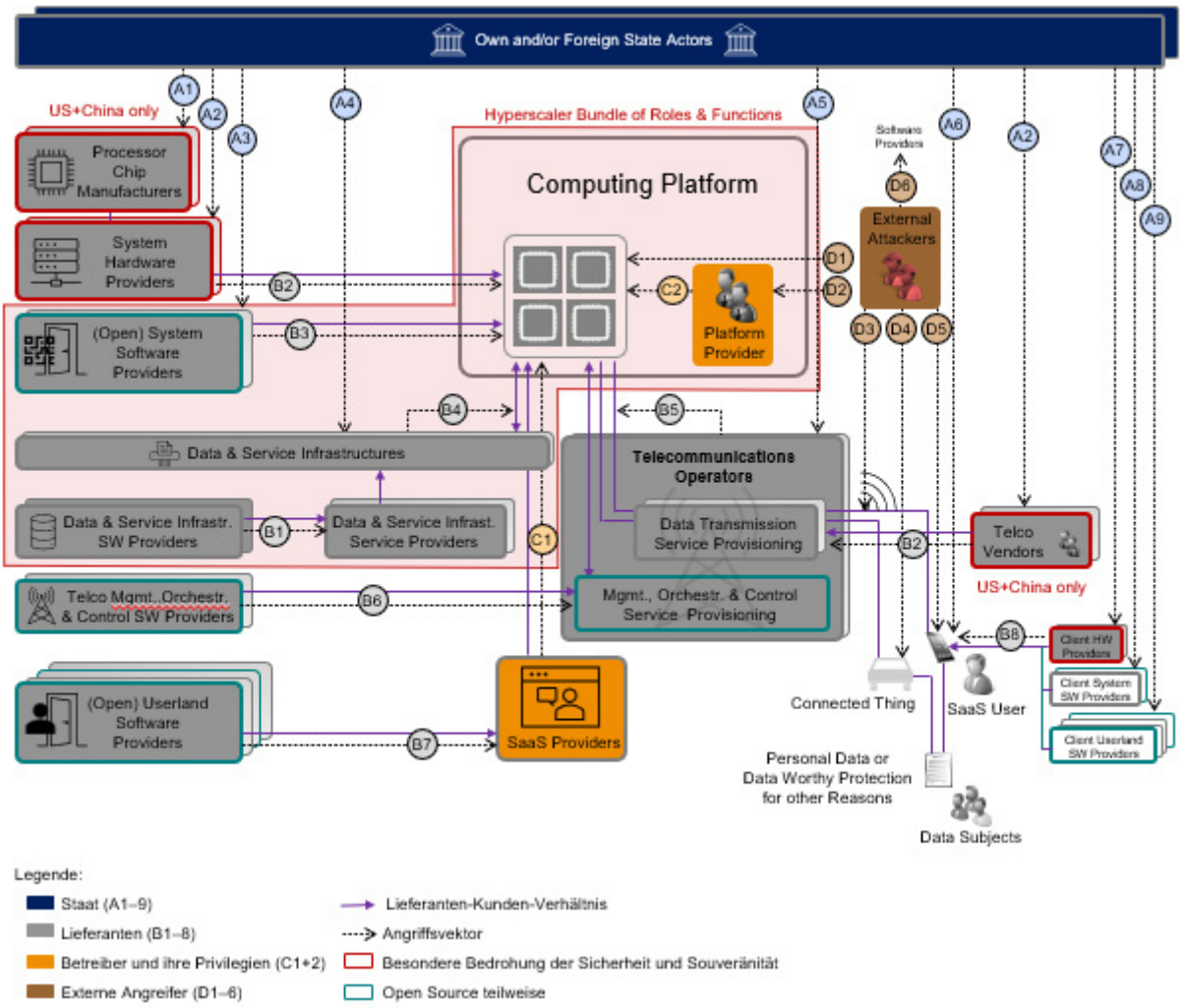

Abbildung 5: Die Angriffsvektoren und die Bedrohungen der Souveränität beim Betrieb einer Computing Platform sowie das Bündel aus Rollen und Funktionen, das von großen Plattform-Anbietern (Hyperscaler) geschnürt wird.

Ergänzt wird das Angebot der Plattform-Anbieter durch Daten- und Infrastruktur-Dienste, die i.d.R. von den Hyperscalern selbst entwickelt und betrieben werden.

Kunden dieser Plattform-Angebote sind sowohl die "Userland Software Provider" als auch die Software Dienste-Anbieter (SaaS Provider), welche die Userland Software ihren Kunden und deren Nutzern zur Verfügung stellen. Teilweise bedienen sich die SaaS Provider auch quelloffener bzw. Open Source Software um sie als Software-Dienst anzubieten. Teilweise werden auch die eigenen Plattformen für den Betrieb der eigenen "Userland Software" genutzt (Beispiel Microsoft: Azure für O365).

Am Ende der Wertschöpfungskette sind die Nutzer der Software-Dienste technisch mit der Computing Platform über die Netze der Telekommunikationsanbieter, fix und/oder mobil (Telecommunications Operators), verbunden. Die Nutzer können Menschen (User) oder Maschinen, Sensoren oder andere "Things" sein.

Die Technik bei den Telekommunikationsanbietern bedient sich durch die Virtualisierung der Netzfunktionen zunehmend ebenfalls informationstechnischer Systeme, die als Computing Platforms bezeichnet werden. Dies sind in modernen Architekturen die Dienste des Managements, der Orchestrierung und Steuerung (Control) der eigentlichen Netzelemente, die für die Übermittlung der Daten sorgen.

Die Softwaredienste werden schließlich auf Endgeräten genutzt, die von Herstellern i.d.R. zusammen mit Systemsoftware ausgeliefert werden (Client Hardware and Client System Software Providers) und auf denen "Client Userland Software" die Nutzung der Softwaredienste erlaubt.

Durch die Softwaredienste (SaaS) werden Daten verarbeitet, die entweder personenbezogen und/oder anderweitig schutzwürdig sind. Der Grad des Schutzbedarfs variiert zwischen verschiedenen Anwendungen, wie in

Kapitel 1 beschrieben. Schließlich sind in Abbildung 3 die betroffenen Personen (Data Subjects) als diejenigen abgebildet, deren Persönlichkeitsrechte durch die Maßnahmen der IT-Sicherheit und des Datenschutzes risikoangemessen gewahrt werden müssen.

Die potenziellen Zugriffe durch Akteure des eigenen Staates oder fremder Staaten können auf den Wegen A1 bis A9 erfolgen.

A1: In den Rechtssystemen zumindest der USA und China ist es jeweils vorgesehen, dass die Regierungsadministrationen mit den Herstellern von Prozessoren und anderen für die Herstellung der Server, Speicher und Netzelemente notwendigen Halbleiter-Chips, Absprachen zur Einrichtung so genannter Hintertüren (Backdoors) treffen dürfen, über die unter bestimmten Umständen Zugriff auf die in den Schaltkreisen verarbeiteten Daten erlangt werden kann.

A2: Dasselbe gilt auch für die Herstellung von zusammengesetzten Systemen wie Servern, Speicher- oder Netzelementen sowie Telekommunikations-Equipment.

A3: Die Server und Speicher- bzw. Netzelemente können nur mit umfangreicher Betriebssystem- und weiterer Betriebssoftware betrieben werden. Diese ist zu einem bedeutenden Anteil Open Source Software, aber auch noch beträchtlich proprietäre Software. Auch in diesen Komponenten könnten staatliche Akteure über Hintertüren Zugriff auf durch die Systeme verarbeitete Daten erlangen.

A4: In analoger Weise gilt das für die Software der Daten- und Infrastrukturdienste.

A5: Dieser Zugriffsvektor beschreibt die Möglichkeiten der Telekommunikationsüberwachung. Diese betreffen bei Ende-zu-Ende-Verschlüsselung der Datenübertragung i.d.R. die Verkehrs- oder Verbindungsdaten, also die Informationen, welches Endgerät wann zu welchen Dienstanbietern Verbindung hatte. Die Verpflichtung der Telekommunikationsanbieter zur Speicherung solcher Überwachungsdaten wird in vielen Rechtssystemen auch "auf Vorrat", also ohne Anfangsverdacht als verhältnismäßig betrachtet, ist jedoch in Europa umstritten. Allerdings sehen die Sicherheitsbehörden den juristischen Streit um diese Speicherpflicht gelassen, da die Daten i.d.R. ohnehin als Abrechnungsdaten gespeichert werden. Die Überwachung der Kommunikationsinhalte durch Auskunftsersuchen bei den Telekommunikationsanbietern ist technisch bei starker Ende-zu-Ende-Verschlüsselung praktisch nicht möglich, anderenfalls jedoch mit gerichtlicher Anordnung Praxis.

A6: Um schwere Straftaten und in weniger rechtsstaatlichen Systemen auch andere Sachverhalte aufzuklären, bedienen sich staatliche Akteure auch der Quellenüberwachung, d.h. dem Zugriff auf die Informationen und Daten über die Quelle (oder Senke) der Daten, also an den Endgeräten (Clients).

A7: Sofern kein physischer Zugriff und die notwendigen Zugangsdaten auf das Endgerät vorliegen, ist die Quellenüberwachung nur über den staatlichen Akteuren bekannte Fehler oder Hintertüren in der "Client Hardware",

A8: der "Client Systemsoftware" oder

A9: der "Client Userland Software" möglich.

Mit den Bezeichnungen B1 bis B8 sind die Angriffe auf die Vertraulichkeit und/oder die Integrität versehen, die von den Lieferanten der Computing Platform ausgehen könnten. Diese können teilweise technisch identisch zu den Zugriffsoptionen durch staatliche Akteure sein. Insbesondere korrespondieren die Angriffe B2-B6 mit der technischen Implementierung der Zugriffe/Angriffe A2-A6. Folgende Erläuterungen ergänzen die zuvor für A1-A8 gemachten Angaben:

B2 und B8:
Bei den Lieferungen der Hardware Provider B2 und teilweise B8 können dies Hintertüren in der Hardware implementiert sein.

B1, B3-B8:
Bei den Software Providern B1, B3 bis B7 und teilweise B8 kann die Software nicht-intendierte Fehler enthalten, die, falls sie als "Exploits" entdeckt werden, zu Angriffszwecken gebraucht werden können oder es können in der Software absichtlich eingebaute Hintertüren vorhanden sein, mit denen sabotiert werden kann oder die Vertraulichkeit sowie die Integrität der verarbeiteten Daten verletzt werden können.

Mit den Bezeichnungen C1 und C2 sind der Missbrauch von Betreiberprivilegien markiert.

C1: Die SaaS-Provider verfügen gewöhnlich über die Souveränität, die Software, die den Kunden und Nutzern zur Verfügung gestellt werden soll, auszuwählen und das Deployment vorzunehmen. Dabei sollten die SaaS-Provider die Kontrolle darüber ausüben, welche Regeln zur sicheren Entwicklung dieser "Userland Software" angewendet werden und welche Software-Komponenten tatsächlich zum Einsatz kommen. Darüber hinaus sollte der SaaS-Provider kontrollieren, welche Daten-Logs angelegt werden und dass ein Rollen- und Rechtekonzept durch die "Businenss Logic" der Userland Software vorgesehen und implementiert ist, das unberechtigten Zugriffe unterbindet.

    Fahrlässig oder arglistig kann jedoch das Rollen- und Rechtekonzept verletzt werden und so fahrlässig zu "Breaches" führen oder zu "Insider"-Angriffen genutzt werden.

C2: Die "Platform Provider" haben i.d.R. ebenfalls ein Rechte- und Rollenkonzept zur Umsetzung der betrieblichen Aufgaben implementiert. Dabei haben die Mitarbeiter die gewissen Rollen des Platform Providers angehören, d.h. beispielsweise die Administratoren der Systemsoftware, die Administratoren der Router oder die Administratoren der Server, privilegierten Zugang zu den Netzelementen bzw. Servern der Computing Platform und damit Souveränität über die auf den Servern ausgeführte Software zu privilegierten Zugriff auf die im Arbeitsspeicher, in den Systembussen und der CPU befindlichen Daten, darunter auch die Daten, die von den Kunden der SaaS-Provider verarbeiteten Daten.

    Dieser privilegierte Zugang und die privilegierten Zugriffsmöglichkeiten können durch Missachtung der Geheimhaltungsverpflichtungen und anderen Vereinbarungen fahrlässig zu Datenlecks führen oder arglistig zu Datendiebstahl oder Verletzung der Integrität der Daten missbraucht werden.

Die vierte Kategorie der Angriffsvektoren beschreibt die externen Angriffe D1 bis D6.
Externe Angreifer ("External Attackers") sind nur zum kleineren Teil "Scipt Kiddies", also die mit einem Heldenmythos umrankten Hackerpersönlichkeiten, die "sportlich" und in der Hauptsache für den eigenen Ruhm arbeiten. In der Mehrzahl sind die externen Angreifer hoch arbeitsteilig organisierte Cyberkriminelle, die motiviert durch rechtswidrig erwirtschaftende Gewinne agieren.

D1: Der Angreifer versucht direkt die Computing Platform zu penetrieren. Hierfür kann er die öffentlich ansprechbaren Internet-Adressen und -Ports der Plattform nutzen. Wenn die Firewalls nur die Adressen und Ports, die für die über die Plattform angebotenen Dienste erforderlich sind, nicht blockieren, werden diese Angriffe über reguläre Zugänge zu einem dieser Dienste ausgeführt. Es können aber auch "physische" Angriffe auf die Plattform direkt dieser Kategorie zugerechnet werden.

D2: Da der Perimeterschutz zur Abwehr der Angriffe D1 sich kontinuierlich verbessert hat, sind inzwischen oft die externen Angriffe über die Infrastruktur und die Privilegien der Platform Provider erfolgsversprechender. Wenn es Angreifern gelingt in die IT des Plattformbetreibers einzudringen, so können mit unterschiedlichen Angriffsmethoden, die von "Key Loggern" bis zu "Social Engineering" reichen, die Privilegien des Anbieters erbeutet und damit analog zu C2 der Betrieb sabotiert oder die Vertraulichkeit und/oder die Integrität der verarbeiteten Daten verletzt werden.

D3: Der Vollständigkeit halber ist der Angriff auf die Übertragung der Daten von den Nutzern zur Cloud und von der Cloud zu den Nutzern hier aufgeführt. Sofern die Verschlüsselung der Datenübertragung sorgfältig ausgeführt ist, ist der Erfolg von Versuchen Daten während der Übertragung auszuspähen oder zu verändern unwahrscheinlich.

D4 und D5:
    Viele der Angriffe auf die durch die Kunden und Nutzer in der Computing Platform verarbeiteten Daten erfolgen über die Endgeräte, d.h. entweder über die "Things" des "Internet of Things" oder die Zugangsgeräte der Nutzer (Smartphone, Tablett, PC, etc.). Die Methoden der Angreifer in diesem Bereich sind ebenfalls sehr vielfältig und reichen von über Social Engineering eingeschleuster Schadsoftware über die Penetration der Geräte über den Angreifern bekannte Schwachstellen in der Client System- und Client Userland Software.

D6: In dieser Angriffskategorie sind die Angriffe von Cyberkriminellen zusammengefasst, die Schadsoftware oder Hintertüren über die "Software Provider" einschleusen, sowohl bei den Entwicklern der "System Software", der "Data & Service Infrastructure Software" als auch der "Userland Software".

In Abbildung 5 sind die besonderen Bedrohungen der Sicherheit und der Souveränität des Betriebs der digitalen Infrastruktur, wie in Kapitel 2 eingeführt, rot umrandet und rot beschriftet.

4.2 Sicherheitskategorien als Struktur zum Systemvergleich

Die Gesamtsicherheit einer sicheren Plattform (Secure Platform) ist durch die ärgsten Schwachstellen in dem in Abbildung 5 gezeigten komplexen Gesamtsystem, also den Angriffsvektoren mit der höchsten Erfolgswahrscheinlichkeit, bestimmt. Daher muss bei der Analyse eine umfassende Betrachtung aller bekannten Angriffsvektoren erfolgen, damit nicht ausgerechnet die gefährlichsten Angriffsvektoren unbeachtet bleiben.

Desweiteren sollen die in Kapitel 5 und 6 formulierten Architekturempfehlungen nicht eingeengt auf die Lösungen einzelner Anbieter am Markt abheben, sondern vielmehr unter Berücksichtigung des Standes der Technik die implementierungsneutralen Empfehlungen zeigen, welche Anforderungen an eine Secure Platform gestellt werden, und wie diese erfüllt werden können.

Um zu diesen Architekturempfehlungen für Secure Platforms zu gelangen, wird eine Methode zum Vergleich verschiedener Sicherheitsarchitekturen benötigt. Dazu ist die Kenntnis der Schutzwirkung von bekannten Schutzmaßnahmen erforderlich. Dies ist im Einzelfall aufwändig und oft von nur schwer quantifizierbaren Faktoren, meistens sogar vom Verhalten menschlicher Individuen, abhängig.

Jedoch kann für die meisten Paare aus Angriffsvektoren einerseits und andererseits den Maßnahmen, wie diesen begegnet werden kann, ein dominierender Einflussfaktor auf die Angriffs- und Angriffserfolgswahrscheinlichkeit identifiziert werden. Dieser dominierende Einflussfaktor wird, wie in Abbildung 6 illustriert, zur Kategorisierung der erzielbaren Sicherheit für bestimmte Angriffsvektor-Maßnahmenpaare herangezogen, die dann für einen Vergleich verschiedener Sicherheitsarchitekturen dienen. Die Kategorien sind nach den Einflussfaktoren geordnet, welche die Angriffs- und Angriffserfolgswahrscheinlichkeit nach Anwendung von Schutzmaßnahmen für je einen Angriffsvektor dominant bestimmen. Sie sind für den in Kapitel 5 angestellten Vergleich verschiedener Sicherheitsarchitekturen nützlich.

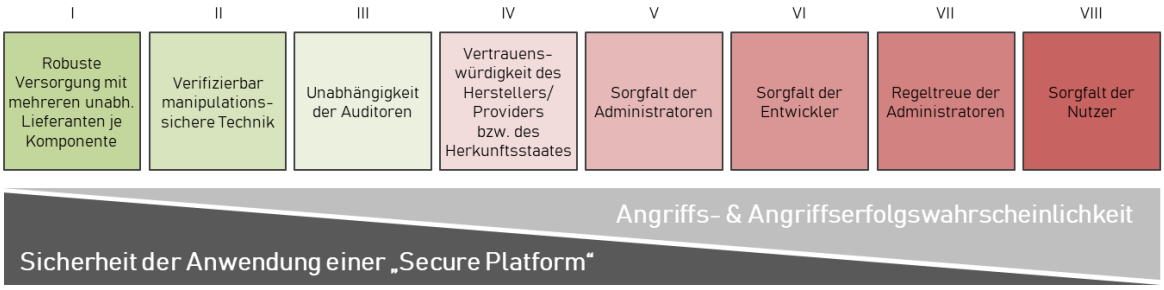

Abbildung 6: Sicherheitskategorien

Sicherheitskategorie I: Mit einer robusten Versorgung der Lieferkette mit mehreren unabhängigen Lieferanten könnte am besten ausgeschlossen werden, dass Sabotageakte aus einer Monopol- oder Oligopol-Stellung eines Lieferanten heraus verübt werden. Die Angriffserfolgswahrscheinlichkeit eines solchen hypothetischen Sabotagevorhabens wäre zwar sehr hoch, jedoch ist die Wahrscheinlichkeit einer solchen Sabotage sehr klein. Sie käme in einer Gesellschaft mit einem hohen Grad an Digitalisierung einer Kriegserklärung gleich. Ein Interesse an einer robusten Versorgung mit Prozessorchips, Systemhardware und Systemsoftware ist gleichwohl aus wirtschaftlicher Hinsicht erstrebenswert.

Sicherheitskategorie II: Verifizierbar manipulationssichere technische Komponenten stellen einen guten Schutz gegen Angriffe dar, die ansonsten durch Veränderungen der technischen Systeme (durch Änderung der Konfiguration, der Programmierung oder der Hardware) möglich würden. Sobald diese Veränderungen von einzelnen Personen oder mehrerer Personen aus einer Organisation möglich sind, sind andere, der nachfolgend genannten Faktoren für die Bestimmung der Erfolgswahrscheinlichkeit der Angriffe dominant und es handelt sich nicht um manipulationssichere technische Subsysteme.

Die Verifizierbarkeit der Eigenschaften von technischen Komponenten ist in unterschiedlichem Maß gegeben, je nachdem, ob nur das äußere Verhalten einer Komponente beobachtet werden kann (Black Box) oder ob auch der innere Aufbau und die Funktionsweise der Komponente durch Dritte nachvollzogen werden kann

(White Box). In diesem Zusammenhang sind die Quelloffenheit (Open Source) bei Hardware und Software sowie die Anwendung von Maßnahmen zur sicheren Softwareentwicklung von Bedeutung.

Sicherheitskategorie III: Wenn ein Auditor die Eigenschaften einer technischen Komponente inspiziert und verifiziert, so handelt er idealerweise völlig unabhängig von den Herstellern der Komponente. Jedoch ist die Gewährleistung der Unabhängigkeit fachlich wie wirtschaftlich in der Praxis nicht immer vollständig gegeben. Eine gar vorsätzliche und arglistige Koalition zwischen dem Hersteller und einem oder mehrerer Auditoren könnte die Sicherheit von Maßnahmen unterlaufen, die gerade auf der Unabhängigkeit der Akteure beruhen. Die Wahrscheinlichkeit für solche arglistige Koalitionen ist bei einer geeigneten Auswahl der prüfenden Organisationen sehr gering.

Sicherheitskategorie IV: Oft jedoch ist eine Verifizierbarkeit der Eigenschaften von technischen Komponenten aus wirtschaftlichen Gegebenheiten oder technischen Gründen der Komplexität gar nicht möglich. Dann ist der dominante Faktor für die Wahrscheinlichkeit von z.B. verdeckten Hintertüren die Vertrauenswürdigkeit des Herstellers. Dem Wahrheitsgehalt der diesbezüglichen Erklärungen der Hersteller eines Subsystems oder Systems können jedoch - je nach Rechtsraum - Rechtsvorschriften zum heimlichen Einbau von Hintertüren trotz der hohen Vertrauenswürdigkeit des Herstellers entgegenstehen.

Sicherheitskategorie V: Viele Sicherheitsmaßnahmen bauen auf die Sorgfalt der sie implementierenden Administratoren auf. Dies betrifft Systemkonfigurationen aber auch das Ausführen von eskalierenden Schritten bei reaktiven Sicherheitsmaßnahmen, wie etwa der Überwachung der Administratoren-Kollegen mit Zugangs- und Zugriffsprivilegien.

Sicherheitskategorie VI: Die Sicherheit der technischen Systeme wird klassisch besonders im Bereich der Software durch Implementierungsfehler und -schwächen stark gemindert. Dem kann nur durch geeignete Methoden der Entwicklung, die eine hohe Sorgfalt fördern und je nach Methode sogar erzwingen, erzielt werden.

Sicherheitskategorie VII: Liegen bei Administratoren kriminelle Arglist oder Erpressung vor, so ist die Angriffserfolgswahrscheinlichkeit von Insider-Angriffen dadurch dominiert und sehr hoch.

Sicherheitskategorie VIII: Kategorie VII kann lediglich durch die massenhaft fehlende Sorgfalt bei den Nutzern übertroffen werden. Es ist ein soziologisches Phänomen - oft auch als die "Attitude Behavior Gap" bezeichnete Diskrepanz zwischen Einsicht in notwendiges Handeln einerseits und dem praktizierten Handeln andererseits, dass Nutzer Sicherheitsregeln verletzen. Es muss Ziel der Gestaltung der Nutzerschnittstellen sein, dass Nutzer gar nicht fahrlässig handeln können. In diese Kategorie fallen nur Paare aus Angriffsvektoren und Maßnahmen, deren Sicherheit sich dominant durch die Sorgfalt der Nutzer bestimmt.

Dabei geht es nicht nur um die strategisch wichtige Digitale Souveränität, sondern auch um die für den technischen Datenschutz notwendigen Maßnahmen bei Datenverarbeitungstätigkeiten mit der höchsten Schutzbedarfsklasse wie etwa Verarbeitungstätigkeiten besonders schutzwürdiger Daten oder Verarbeitungstätigkeiten großer Datenbestände (Künstliche Intelligenz/Big Data), mit denen Persönlichkeitsprofile erstellt werden können.

5 Anforderungen an eine technische Architektur

Die Empfehlungen für eine technische Architektur einer Secure Platform sind so aufgebaut,

- dass zunächst aus der Betrachtung der Angriffsvektoren jeweils eine für eine ideale Secure Platform aus dem Stand der Wissenschaft und Technik als implementierbar bekannte Sicherheitskategorie gemäß Kapitel 4 als Anforderung formuliert wird,
- dann die dafür notwendigen Zieleigenschaften der Secure Platform definiert werden,
- sodann die für die Implementierung zur Verfügung stehenden Entwurfsmuster vorgestellt werden
- und schließlich eine übersichtliche Referenzarchitektur präsentiert wird.

Mit Hilfe der Ergebnisse wird dann auf die Anforderungstabelle zurückkommend eine Einschätzung vorgelegt, mit welchem grob geschätzten Aufwand in den nächsten Jahren den Angriffsvektoren begegnet werden könnte.

5.1 Das Idealkonzept einer Secure Platform

In Tabelle 1 sind in der ersten Spalte "Angriffsvektoren" die in Kapitel 4.1 und Abbildung 5 eingeführten Bedrohungen der Sicherheit und der Souveränität von Computing Platforms mit derselben Bezeichnungsnummerierung zeilenweise aufgeführt.

In der zweiten Spalte "Aktuelle Situation Computing Platforms" sind mit römischen Ziffern entsprechend Abbildung 6 in Unterkapitel 4.2 die Einflussfaktoren aufgeführt, welche die Angriffs- und Angriffserfolgswahrscheinlichkeit nach Anwendung von Schutzmaßnahmen je einem Angriffsvektor in den aktuellen Computing Plattformen dominant bestimmen.

In der dritten Spalte "Anforderungen an das Idealkonzept Secure Platform" sind die Einflussfaktoren gemäß derselben Nummerierung mit römischen Ziffern angegeben, die dann dominant die Sicherheit bestimmen würden, wenn die der Wissenschaft und Technik bekannten Maßnahmen bereits zum Einsatz kämen.

Die vierte Spalte "Umsetzbarkeit" gibt die Einschätzung der Autoren dieses Positionspapiers wieder, welcher Implementierungsaufwand in etwa erforderlich wäre, um von der aktuellen Situation zu den Anforderungen des Idealkonzepts innerhalb der nächsten Jahre zu gelangen. Die Aufwände sind grob in drei Stufen, kleiner 10 Personenjahre (PY), kleiner 100 PY und größer 100 PY angegeben. Voraussetzung ist, dass die öffentliche Hand die Implementierung von Secure Platforms unterstützt und fördert.

Angriffsvektoren		Aktuelle Situation Computing Platforms	Anforderung an das Idealkonzept Secure Platform	Umsetzbarkeit
Staat	A1: Backdoor → Prozessor/Chip	IV	I	> 100 + PY
	A2: Backdoor → System Hardware	IV	I	> 100 + PY
	A3: Backdoor → (Open) System Software	IV	III	< 100 PY
	A4: Backdoor → Daten/Infrastrukturdienst	IV	III	< 100 PY
	A5: Überwachung → TK	V	IV	< 100 PY
	A6: Quellenüberwachung → Device SaaS User	V	IV	< 100 PY
	A7: Fehler/Backdoors → Client Hardware	V	IV	< 100 PY
	A8: Fehler/Backdoors → Client Systemsoftware	V	IV	< 100 PY
	A9: Fehler/Backdoors → Client Userland Software	V	IV	< 100 PY
Lieferanten	B1: Exploits → D&S Infrast. Service Provider	IV	III	> 100 + PY
	B2: Hardware Backdoor → Computing Platform	IV	III	> 100 + PY
	B3: Exploits → Computing Platform	IV	III	> 100 + PY
	B4: Exploits/Backdoor → D&S Infrast. & Computing Platform	IV	III	> 100 + PY
	B5: Exploits/Backdoor → TK & Computing Platform	IV	III	> 100 + PY
	B6: Exploits/Backdoor → Mgmt. O&C Service Provisioning	IV	III	> 100 + PY
	B7: Exploits/Backdoor → SaaS Providers	IV	III	< 100 PY
	B8: Backdoor → Device SaaS User	IV	III	< 100 PY
Betrei-her	C1: Missbrauch IAM → Computing Platform	IV	III	< 100 PY
	C2: Missbrauch PAM → Computing Platform	VII	II	< 100 PY
Externe Angriffe	D1: Penetration → Computing Platform	V	II	< 100 PY
	D2: Privilegien erbeuten → Platform Provider	V	II	< 100 PY
	D3: Datenübertragung → SaaS User	IV	IV	< 10 PY
	D4: Angriff → IoT SaaS User	VIII	VI	< 100 PY
	D5: Angriff → Device SaaS User	VIII	VI	< 100 PY
	D6: Schadsoftware/Backdoors → Software Provider	VI	III	< 100 PY

Tabelle 1: Gegenüberstellung aktueller Sicherheitsarchitektur von Computing Platforms und dem Ideal einer Secure Platform anhand der definierten Angriffsvektoren und Sicherheitskategorien (Kapitel 4)

5.2 Zieleigenschaften einer Architektur für Secure Platforms

Wie aus den vorherigen Kapiteln abgeleitet, müssen neben einer konkurrenzfähigen Rechenleistung (hinreichend Terrra- und PetaFLOPS und Arbeitsspeicher) zur Erledigung der Datenverarbeitungsaufgaben folgende, für die Sicherheit strategisch wichtige Zieleigenschaften auch auf technischer Ebene erreicht werden:

- **Erneuer- und Austauschbarkeit**
  Ebenso wie Lieferanten durch alternative Anbieter ersetzt werden können müssen, muss dies auch für alle Hard- und Softwarekomponenten möglich sein. Nur so kann ein Betreiber seine Systeme jederzeit in einem vertrauenswürdigen Zustand betreiben. Wird eine schwerwiegende Schwachstelle bekannt oder kommen aus anderen Gründen Zweifel an der Vertrauenswürdigkeit eines Herstellers auf, so kann die betroffene Hard- oder Softwarekomponente gegen die eines anderen Herstellers ausgetauscht werden.

- **Hinreichende Isolation verschiedener Datenverarbeitungsaufgaben (Workloads)**
  Da die Secure Platform einer großen Anzahl wechselnder Benutzer ermöglichen soll, eine Vielzahl diverser, schnell wechselnder Anwendungen zu betreiben, müssen die verschiedenen Datenverarbeitungsaufgaben gut genug voneinander isoliert sein, so dass keine Anwendung einen ungewollten Zugriff auf Daten einer anderen Anwendung erlangen kann. In solchen Multi-Tenant-Systemen ist stets davon auszugehen, dass eine Nachbar-Anwendung von Cyberkriminellen verwendet werden könnte, um unbefugten Zugriff auf die durch andere Nutzer in der Plattform verarbeitete Daten zu erlangen.

- **Zugriffsschutz, keine unnötigen privilegierten Zugriffsmöglichkeiten**
  Neben möglicherweise gefährlichen Nachbaranwendungen, stellen auch die privilegierten Zugänge der Systemadministratoren, der Server-, Netzwerk-, Software-, Datenbank- und Storageadministratoren eine Gefahr für die Unversehrtheit der Vertraulichkeit, der Integrität und der Verfügbarkeit der mit der Plattform verarbeiteten Daten dar. Nicht nur weil die Administratoren den Zugriff missbrauchen könnten, sondern auch weil externe Angreifer deren Zugriffsmöglichkeiten erbeuten könnten.

  Manuelle Eingriffe durch Administratoren, die betrieblich einen privilegierten Zugriff auf die Software und/oder die Daten erfordern, sind für moderne, skalierende Cloud-Anwendungen nicht mehr notwendig, da aus ökonomischen Gründen ohnehin die vollständige Automatisierung aller Schritte der Datenverarbeitung angestrebt wird und für den harten Wettbewerb ein tiefes "Debugging" für hohe Nutzerzufriedenheit i.d.R. zeitlich vor dem Massen-Roll-Out und nicht an der auf der Plattform eingesetzten Live-Software zu erfolgen hat.

  Insbesondere für Anwendungen, die sich der Methoden des maschinellen Lernens oder Künstlicher Intelligenz bedienen, ist eine privilegierte Zugriffsmöglichkeit oft problematisch, da mit der Platform eine große Menge an meistens personenbezogenen Daten verarbeitet werden, für deren Verarbeitung der Erlaubnistatbestand oder andere Rechtsgrundlagen nicht oder nur aufwändig zu beschaffen sind. Kann jedoch der privilegierte Zugriff ausgeschlossen werden, so muss lediglich für die Weiterverarbeitung der Ergebnisse der Datenverarbeitung ein entsprechender Erlaubnistatbestand vorliegen. Dies erleichtert die Anwendbarkeit von Big Data grundlegend.

  Es ist also eine Zieleigenschaft der Secure Platform, privilegierte Zugriffe von vorneherein auszuschließen.

- **Auditierbarkeit**
  Durch Testen kann stets nur die Anwesenheit von Fehlern und Schwachstellen nachgewiesen werden, niemals jedoch deren Abwesenheit ("program testing can be a very effective way to show the presence of bugs, but it is hopelessly inadequate for showing their absence"[3]). Je komplexer ein System ist, aus je mehr Komponenten es also besteht, und je vielfältiger deren Zusammenspiel ist, desto geringer ist die praktisch erreichbare Abdeckung der Menge aller möglichen Systemzustände durch Tests und andere Untersuchungen. Daraus folgt im Umkehrschluss, dass mit der Komplexität auch die Anzahl der unentdeckt im Produkt verbleibenden Fehler und Schwachstellen zunehmen muss. Deshalb ist die Komplexität der schlimmste Feind der Sicherheit ("complexity is the worst enemy of security"[3]) und es gilt sie daher auf das unvermeidbare Maß zu reduzieren.

- **Manipulationssicherheit**
  Sind durch die technisch-organisatorische Gestaltung der Secure Platform ein genügend hoher Grad an Erneuer- und Austauschbarkeit, eine hinreichende Isolation der Datenverarbeitungsaufgaben (Workloads) voneinander, ein hinreichender Zugriffsschutz und die Auditierbarkeit gegeben, dann könnten durch Manipulationen der Secure Platform genau diese Zieleigenschaften wieder verlorengehen. Daher muss eine Secure Platform sicher gegen Manipulationen gestaltet sein. Die bedeutet, dass sicherheitstechnisch relevante Eigenschaften einer Secure Platform nur durch das Zusammenwirken von mehreren unabhängigen Parteien verändert werden können dürfen.

- **Gegebenenfalls Sabotagesicherheit**
  Ist eine Secure Platform manipulationssicher mit technischem Zugriffsschutz ausgestattet, dann ist es einzelnen Mitarbeitern des Plattform Providers zwar nicht mehr möglich die Vertraulichkeit und/oder Integrität von Daten oder Software, die mit der Platform verarbeitet werden, zu verletzen, es ist aber immer noch möglich, den Betrieb der Plattform zu sabotieren. Da eine Teilmenge der kritischen Anwendungen einer digitalisierten Gesellschaft auch einen hohen Schutzbedarf gegen Sabotage aufweisen, soll die Referenzarchitektur der Secure Platform auch für eine ggf. erforderliche sabotagesichere Ausgestaltung geeignet sein.

6 Empfehlungen für eine technische Architektur

Die Empfehlungen für eine technische Architektur bestehen aus einer Sammlung der Entwurfsmuster und einer Einschätzung der prinzipiellen Machbarkeit dieser. Dann wird eine Referenzarchitektur mit den für den Aufbau einer Secure Platform notwendigen Komponenten und deren Zusammenwirken eingeführt. Und schließlich werden weiter detaillierende Entwurfsempfehlungen als Konkretisierung von IEC 62443-4-2 abgegeben.

6.1 Entwurfsmuster und prinzipielle Machbarkeit

Mit den in den nachfolgenden Abschnitten beschriebenen Entwurfsmustern können die in Kapitel 4.2 geforderten Zieleigenschaften implementiert werden:

- **Ausschließliche Verwendung standardisierter Schnittstellen nach offenen Standards**
  Durch die ausschließliche Verwendung standardisierter mechanischer, elektrischer, und datentechnischer Schnittstellen werden Komponenten und (Sub-) Systeme prinzipiell austauschbar. Die Konformität eines Produktes mit dem Standard garantiert dem Betreiber ein Mindestmaß an Interoperabilität zwischen Produkten verschiedener Hersteller. Bei Ausfall eines Herstellers, oder einer seiner Produktlinien muss nicht die Systemarchitektur angepasst werden, sondern ein Produkt eines anderen Herstellers, welcher dieselben, standardisierten Schnittstellen bietet, kann im Idealfall sofort das ausgefallene Produkt ersetzen (drop-in replacement).

  Um hierbei Abhängigkeiten von bestimmten Herstellern zu vermeiden, ist es von entscheidender Bedeutung, darauf zu achten, nur offene Standards zu verwenden und nicht erst im dringenden Bedarfsfall, sondern schon von vorneherein zwei oder mehrere alternative Implementierungen parallel zueinander einzusetzen. Als offene Standards können Standards angesehen werden, die allen folgenden Anforderungen genügen:

  Während ihrer Entwicklung:

  - **Offenheit:** Die Entwicklung erfolgt auf der Basis offener Prozesse zur Entscheidungsfindung, an denen sich alle interessierten Marktteilnehmer aus dem oder den betroffenen Märkten beteiligen können.
  - **Konsens:** Die Prozesse zur Entscheidungsfindung fußen auf Zusammenarbeit und Konsensbildung und bevorzugen niemanden.
  - **Transparenz:**
    - Alle Informationen rund um technische Diskussionen und die Entscheidungsfindung werden dokumentiert, und in wieder auffindbarer Form archiviert.
    - Informationen über neue Standardisierungsaktivitäten sind öffentlich, und werden leicht auffindbar und zugänglich publiziert.
    - Die Teilnahme aller relevanten Kategorien interessierter und betroffener Organisationen am Standardisierungsprozess wird angestrebt, auch im Hinblick auf Erreichung eines Gleichgewichts.

  Während ihrer Anwendbarkeit:

  - **Wartung:** Eine fortwährende Pflege und Wartung der Standards sind über einen langen Zeitraum garantiert.
  - **Verfügbarkeit:** Die Standards stehen zur Implementierung und Verwendung zu angemessenen Bedingungen (einschließlich einer angemessenen Gebühr oder kostenlos) öffentlich zur Verfügung.
  - **Lizensierbarkeit:** Rechte an geistigem Eigentum (Intellectual Property Rights, Abk. IPR), die für die Umsetzung der Standards technisch wesentlich sind, werden rechtzeitig deklariert, und können zu fairen, angemessenen und nichtdiskriminierenden Bedingungen (fair, reasonable, and non-discriminatory, Abk. FRAND) lizenziert werden, gegen eine angemessene Gebühr oder kostenlos.
  - **Relevanz:** Die Standards sind effektiv und relevant und entsprechen den Marktanforderungen, sowie ggf. regulatorischen Anforderungen.
  - **Neutralität und Stabilität:** Die Standards schränken die Möglichkeiten für Implementierende, auf deren Grundlage in Wettbewerb zu treten und Innovationen zu entwickeln, nicht wesentlich ein.

- **Qualität:** Die Qualität und der Detaillierungsgrad der Standards reichen aus, um eine Vielzahl konkurrierender Implementierungen interoperabler Produkte und Dienstleistungen zu entwickeln.

- **Einsatz von Open Source Hardware und Software**
  Wenn für Software der vollständige Quellcode, bzw. für Hardware die vollständigen Konstruktionsdaten vorliegen, bestehen große Vorteile bezüglich der o.g., übergeordneten Ziele:

  - Wird ein hinreichend schwerwiegender Fehler oder Schwachstelle in einem Open Source Produkt entdeckt und kann der Hersteller diese nicht oder nicht schnell genug beseitigen, so kann der Betreiber des Systems entweder, bei Vorhandensein der entsprechenden Fähigkeiten, den Fehler selbst beseitigen oder andernfalls einen geeigneten Dienstleister damit beauftragen.

    Meist werden Open Source Produkte von einer Gemeinschaft von Interessierten gewartet und weiterentwickelt. Informieren Betreiber diese Gemeinschaft über Probleme, so werden diese regelmäßig behoben. Hierdurch profitieren sowohl die Interessierten (mit der Qualität steigt ihr Ansehen), der Entdecker des Problems (er erhält eine Lösung für sein Problem), sowie auch alle anderen, die dasselbe Produkt einsetzen (das Problem wird behoben, bevor es sich bei Ihnen auswirken konnte).

    Damit ist für Open Source Software die Erneuer- und Austauschbarkeit stets gegeben.

  - Eine gute Auditierbarkeit ist durch das Vorliegen der vollständigen Konstruktionsdaten regelmäßig besser gegeben, als wenn die Konstruktionsdaten aus Gründen der Wahrung von Betriebsgeheimnissen nur teilweise offengelegt werden.

- **Entwicklung angepasster Systemarchitekturen zur Reduktion der Komplexität**
  Im Gegensatz zu Mainframe-Systemen, die nur in vergleichsweise geringen Stückzahlen eingesetzt werden, sind die CPU- und Rechnerarchitekturen der heute in großen Mengen produzierten und eingesetzten Rechner, d.h. der sogenannten Blade Server in Rechenzentren und Laptop- und Desktop-Rechnern für Büroeinsatz und Endverbraucher, aus den Micro- und Mini-Rechnern der 1970er Jahre entstanden. Ziel dieser Architekturen war es, ein System zu schaffen, mit dem ein einzelner Benutzer oder eine kleine Anzahl von Benutzern eine einzige Hauptanwendung betreibt, wie z. B. die Warenwirtschaft in einem mittelständischen Unternehmen, oder die Verarbeitung von Messwerten in einem Labor. Da solche Rechnersysteme heute aber längst wie Mainframe-Rechner benutzt werden (eine große Anzahl wechselnder Benutzer betreibt eine Vielzahl diverser, schnell wechselnder Anwendungen), ist diese Grundannahme über die von der Architektur abzudeckenden Anwendungsfälle längst nicht mehr zutreffend.

  Als Anpassungsstrategie hierfür haben die CPU-Hersteller versucht die Ausführungsgeschwindigkeiten immer weiter zu erhöhen. Allerdings haben sie hierzu nicht die Architektur als Ganzes angepasst oder neu entworfen, sondern nur wiederholt inkrementelle Verbesserungen rund um die Ausführungseinheit der CPU vorgenommen (Steigerung Taktfrequenz, Instruction Pipeline, Branch Prediction, Speculative Execution, Hyperthreading, usw.). Zwar sind alle diese Optimierungen Äquivalenzumformungen (und bleiben dem Anwendungsentwickler daher verborgen), jedoch nur hinsichtlich des Kontrollflusses einer Anwendung. Die Isolation verschiedener Anwendungen gegeneinander wurde bisher nur ungenügend unberücksichtigt, da der primäre Fokus der Hersteller die Ausführungsgeschwindigkeit war. Als Folge dieser Strategie sind in jüngster Vergangenheit zahlreiche Schwachstellen in CPUs verschiedener Hersteller entdeckt worden, die unerlaubten Zugriff auf Daten anderer Prozesse ermöglichen (z.B. Meltdown und Spectre[4], Foreshadow[5], ZombieLoad[6,7], L1D Eviction Sampling[8], Platypus[9], Take A Way[10]).

  Auch die Systemhersteller haben Maßnahmen ergriffen, um die Sicherheit ihrer Rechnersysteme zu erhöhen und spezielle Mikrochips entwickelt, die den Bootprozess überwachen, sodass keine unerwünschte Software ausgeführt wird und sensible Operationen in einer von der CPU getrennten Umgebung ausführen, sodass von auf der CPU ausgeführter Software keine geheimen Daten oder Schlüssel entwendet werden können. Aber auch die Systemhersteller haben die Architekturen Ihrer Produkte nur graduell angepasst, und auch die dort hinein integrierten Sicherheitschips enthalten Implementierungs- oder konzeptionelle Fehler, sodass auch sie Schwachstellen aufweisen (z.B. Plug'nPwn[11], CSME Boot ROM[12]).

  Im Bereich der Software hat mit der Einführung der Virtualisierung und der Container die größte und tiefgreifendste Veränderung der Architektur stattgefunden. Da, wie oben beschrieben, die CPU- und Rechnerarchitekturen für den Massenmarkt sich aber nicht grundlegend gewandelt haben und zwangsläufig mit Ihnen die in diesem Segment eingesetzten Betriebssysteme, konnten sich die mentalen Modelle, die der Anwendungsentwicklung zugrunde liegen, ebenfalls nicht wesentlich von denen der Mikro- und Mini-

Rechner der 1970er Jahre weiterentwickeln. Deshalb bietet die große Mehrheit der Virtualisierungsumgebungen heute Abstraktionen auf der Ebene eines solchen Betriebssystems an. Jedoch wird ein Großteil der von Virtualisierungs- und Containerplattformen angebotenen Funktionen von typischen, darauf betriebenen Anwendungen in der Praxis nie benutzt werden. Diese Funktionen werden lediglich vorgehalten, um eine möglichst breite Palette an Anwendungen ohne Änderung der Plattform unterstützen zu können. Diese (ungenutzte) Komplexität führt unausweichlich (vgl. oben) zum vermehrten Vorhandensein von Implementierungsfehlern und Schwachstellen in den Virtualisierungs- und Containerplattformen selbst, die es erlauben aus einem Container auszubrechen und auf Daten anderer Anwendungen zuzugreifen (z.B. Doki[13]).

Eine dauerhaft erfolgversprechende Lösung zur Überwindung der entstandenen Diskrepanzen zwischen Rechnerarchitekturen im Massenmarkt und tatsächlich auftretenden Einsatzszenarien besteht darin, eine neue, besser angepasste CPU- und Rechnerarchitektur zu entwerfen, unter Verwendung von, bzw. Inspiration durch, Mainframe-Konzepte:

- Minimalismus siegt (Komplexität gehört in die Anwendung, nicht in die Plattform)
- Virtualisierung ist das neue Normal
- dynamisches Ressourcenmanagement unter Berücksichtigung der Isolation von Anwendungen und Benutzern bereits in der CPU
- Austausch von Hard- und Software im laufenden Betrieb durch Klustereigenschaften (Uptime in Jahrzehnten messen)

Auch Mainframe-Systeme sind selbstverständlich nicht gegen erfolgreiche Angriffe immun; auch sie enthalten Implementierungsfehler und Schwachstellen.[14] Aber auf Grund Ihrer besser an die tatsächlichen Einsatzszenarien angepassten Architekturen weisen sie deutlich weniger ungenutzte Komplexität auf. Deshalb ist es grundsätzlich leichter ein solches besser angepasstes System sicher zu konfigurieren und zu betreiben.

- **Arbeitsspeicher-Enklaven und Verschlüsselung bis zur CPU als "Trusted Execution Environment"**
  Durch mehrere Hersteller von Prozessoren liegen seit Mitte der 2010er Jahre Implementierungen der Funktionalität einer Hardware-nah implementierten "Trusted Execution Environment" (Processor TEE) genannten gesicherten und vertrauenswürdigen Verarbeitungsumgebung vor, die unter dem Namen "Confidential Computing" vermarktet werden.

  In direkter Nachbarschaft zur CPU werden dabei Einheiten zur Ver- und Entschlüsselung mit auf den Prozessorchip gebracht, die jeden Datenverarbeitungsbefehl, der an die CPU gesendet wird, erst entschlüsseln, bevor er in der CPU (notabene unverschlüsselt) ausgeführt wird und jedes Datenverarbeitungsergebnis zuerst verschlüsseln, bevor es wieder über die Busse zum RAM zurückgeleitet wird. Datenverarbeitung mit verschlüsselten Daten (z.B. Homomorphe Verschlüsselung oder Multi-Party-Computation) ist nur für Lösungen von speziellen Problemen und extrem, hohem Rechenaufwand möglich und daher kein geeignetes Entwurfsmuster für Secure Platform.

  Die Teile des Arbeitsspeichers, die für diese Arbeitsweise vorgesehen sind, werden "Enklaven" (engl. enclaves) genannt. Für deren Initialisierung und Betrieb sind weitere Erweiterungen am Prozessorchip notwendig, wie etwa ein Register (Sealing Array), das die Schlüssel zum Ent- und Verschlüsseln bereithält.

  Ab dem Moment der Initialisierung, für den die Prozessorhersteller auch Verfahren zur Verifikation der Identität des Prozessors und der Attestierung des Zustandes des Servers implementieren, schützen die Prozessor-TEE insbesondere gegen Angriffe durch Schwachstellen in den Betriebssystemen und den Komponenten der Systemsoftware. Würden Teile der verschlüsselten Userland-Daten oder -Software in den Enklaven durch Schadsoftware oder Administratorbefehle verändert, so wäre keine Entschlüsselung der Befehle an der CPU mehr möglich. So wird eine hohe Isolation zwischen den Workloads in verschiedenen Enklaven erreicht.

  Eine sorgfältige Implementierung des Initialisierungsprozesses vorausgesetzt, bieten die Prozessor-TEE auch einen guten Schutz gegen unnötigen und/oder unbefugten Zugriff durch den Platform Provider, daher die Bezeichnung "vertrauliche Verarbeitung" bzw. "Confidential Computing". Der privilegierte Zugriff und die Erbeutung von privilegierten Zugangsdaten durch externe Angreifer sind praktisch nicht mehr möglich.

  Für eine sorgfältige Implementierung des Initialisierungsprozesses muss beispielsweise sichergestellt sein, dass der Schlüssel, mit dem die Daten, die sich im RAM, auf den Bussen oder im Storage befinden,

verschlüsselt werden, niemandem, insbesondere nicht dem Personal des Platform Providers bekannt werden können.

- **Perimetersensorik für Multi-Server-TEE und vorsorgliche Löschung als Schutz vor Manipulation**
  Eine von den Prozessorchips und ihren Herstellern weniger abhängige Möglichkeit Confidential Computing zu implementieren ist, eine mit Penetrationssensorik ausgerüstete Kapsel auf Gehäuse- oder Rack-Ebene, um einen oder mehrere Server zu bilden (Multi-Server TEE). In solchen Systemen wird entsprechend dem Standard für die höchste Schutzklasse bei Hardware Security Modulen, FIPS 140-2, beim Ansprechen der Penetrationssensorik eine vorsorgliche Löschung der in diesem Moment unverschlüsselten in der Kapsel vorliegenden Daten vorgenommen (Data Clean-up).

  Wie bei einer Prozessor-TEE werden die zu verarbeitenden Daten erst innerhalb der Multi-Server-TEE entschlüsselt und vor der dauerhaften Speicherung im verteilten Speichersystem oder dem Verlassen der TEE wieder verschlüsselt. Der sichere Betriebszustand, der so genannte "versiegelte Zustand" der TEE, wird erst nach einem Audit und nachfolgender Initialisierung der TEE erreicht.

  Die Versiegelung einer TEE basiert auf einem Audit durch unabhängige und auf die Technologie der High Performance Computing Backends spezialisierte Auditoren (Platform Auditors), welche die gesamte Secure Platform einschließlich der Hardware als auch der Software inspizieren und prüfen. Das Ergebnis der Prüfung eines Confidential Computing Moduls attestieren die Platform Auditoren durch Eingabe von für jeden Auditor spezifischen Teilgeheimnissen (die initial beispielsweise mit dem Shamir-Secret-Sharing-Verfahren erzeugt werden). Aus einem Quorum dieser Teilgeheimnisse wird durch eine Sealing Logic das Root-Geheimnis zur Verschlüsselung der Daten erzeugt. Da das Root-Geheimnis in einem volatilen Element gespeichert ist, wird es bei einem Verdacht einer logischen oder physischen Penetration durch den Mechanismus des Data Clean-up ebenfalls verworfen.

  Die Prozessor-TEE schützen insbesondere gegen Angriffe durch Schwachstellen in den Betriebssystemen und den Komponenten der Systemsoftware. Die Penetrationssensorik-basierten Multi-Server-TEE schützen besser vor Side-Channel-Angriffen. Den besten Schutz bieten die Kombinationen von Penetrationssensorik-basierten Muli-Server-TEE und Chip-basierten Prozessor-TEE.

  Da der versiegelte Zustand erst nach Eingabe mehrerer Teilgeheimnisse, welche dem notwendigen Quorum entsprechen müssen, erreicht werden kann und sich dafür ein Quorum an unabhängigen Auditoren über den Zustand eines Confidential Computing Moduls einig sein müssen, wird mit einem Multi-Server TEE Manipulationssicherheit für ein solches Modul erreicht[15]. Jeder durch die Penetrationssensorik registrierbare Manipulationsversuch führt zum Verlust des Root-Geheimnisses, das nur mit dem Quorum der Auditoren durch die Sealing Logic wiederhergestellt werden kann.

  Schließlich ist eine vertrauenswürdige Identifikationsschnittstelle eine Voraussetzung dafür, dass die Platform Auditors auch tatsächlich beurteilen können, ob sie ihr Teilgeheimnis der korrekten Schnittstelle anvertrauen, d.h. ob die Authentizität der Schnittstelle gewährleistet ist.

- **Zum Transport geeignete "Confidential Computing Module"**
  Es wurde mit formalen Arbeiten[16] gezeigt, dass ohne die Attestierung der Vertrauenswürdigkeit der Verarbeitungsumgebungen durch mehrere, unabhängige, menschliche Auditoren aus Fleisch und Blut keine Vertrauenskette für Datenverarbeitung aufgebaut werden kann.

  Da der vollständige Audit eines Confidential Computing Moduls sich jedoch auf mehrere Fachdisziplinen verteilt und ein hohes Maß an Fach- und Sachkunde sowie ein hohes Maß an Sorgfalt erfordert, ist eine dennoch möglichst kostengünstige und fehlerarme Implementierung des Auditprozesses erforderlich. Besonders günstig und weniger fehleranfällig sind diese Audits, wenn sie anstatt in der heterogenen Rechenzentrumslandschaft in einer zentralen Lokation mit industriellen Methoden organisiert werden. Allerdings müssen dann die fertigen Module zu ihrem Einsatzort transportiert werden können, ohne dass die Root-Geheimnisse im volatilen Element, welche das Ergebnis des Audits und der Versiegelung repräsentieren, verloren gehen. Dafür bedarf es einer Stützbatterie für die Komponenten der Penetrationssensorik und des volatilen Elements.

  Mit solchen, zum Transport geeigneten und standardisierten Confidential Computing Modulen, können die Betriebsabläufe von Platform Providern verbessert werden, indem in den Rechenzentren nicht einzelne Server, Netzelemente usw. individuell gewartet werden, sondern komplette Computing Module (IT-Packs) bei Bedarf ausgewechselt werden. Oft wird bei solchen Betriebsmodellen nicht bei einem fehlerhaften

Modul sofort ein Austausch initiiert, sondern gewartet, bis eine gewisse Quote von Modulen der Wartung bedarf, sodass ein ökonomischer Austausch mehrerer Module auf einmal erfolgen kann.

- **Verteilung der Workloads und der verschlüsselten Daten auf Knoten eines Computing-Netzes**
  Die zuvor erläuterten Entwurfsmuster dienen in erster Linie zur Sicherung der Vertraulichkeit und der Integrität der mit der Platform verarbeiteten Daten. Zur Sicherung der Verfügbarkeit der Plattform, des Schutzes gegen Datenverlust und als Schutz gegen Sabotage, dient die Verteilung der verschlüsselten Daten im Speichersystem als auch die Verteilung der Workloads auf verschiedene Confidential Computing Module, die als Knoten eines Computing-Netzes aufgefasst werden können.

  Eine hinreichende Vernetzung der Betriebsorte für die Confidential Computing Module mit hoher Bandbreite und geringer Latenz vorausgesetzt, finden sich nicht nur auf den primär genutzten, sondern auch auf sekundär genutzten Knoten solcher Netze die synchron replizierten Daten der Workloads, sodass bei Ausfall eines Confidential Computing Moduls, keine Daten verloren gehen. Zusätzlich können zu Knoten, zu denen die Übertragungsbandbreite und/oder die Latenz weniger vorteilhaft sind, asynchrone Replikationen vorgenommen werden.

  Ebenso können durch die Virtualisierungs- und Workload-Management-Software die Datenverarbeitungsaufgaben zu Zwecken der Lastverteilung und zur Sicherung der Verfügbarkeit bei Ausfall eines Knotens zu anderen, vom Ausfall nicht betroffener Knoten verschoben werden.

  Zum Schutz gegen Sabotage sind zusätzliche Algorithmen notwendig, die verborgen im Computing Netz zu anderen Knoten, deren Identitäten außerhalb des Computing-Netzes nicht bekannt sein können, die vorsorgliche Replikation der Daten und die Vorbereitung einer Verschiebung der Workloads vornehmen.

6.2 Referenzarchitektur für die technischen Empfehlungen

Die in Kapitel 6.1 aufgeführten Entwurfsmuster können zum Aufbau unterschiedlich gestalteter Secure Platforms einzeln oder gemeinsam herangezogen werden. Wenn nur ein Teil der Entwurfsmuster zum Einsatz kommt, müssen jedoch Kompromisse bezüglich der erzielbaren Sicherheit eingegangen werden. Die im Folgenden vorgestellte Referenzarchitektur zeigt eine umfassende Verwendung der benannten Entwurfsmuster. Der Leser kann optional auch nur Teile der Referenzarchitektur für seine Zwecke nutzen, jedoch sei empfohlen, die damit einhergehenden Sicherheitseinbußen zu berücksichtigen.

Zunächst wird das Umfeld einer Secure Platform in Abgrenzung zum in Kapitel 5 präsentierten Ist-Zustand des Umfeldes für Computing Plattformen erläutert. In
Abbildung 7 ist analog zu Abbildung 5 das Umfeld und die gesamte Wertschöpfungs- und Lieferkette einer Secure Platform mit den Angriffsvektoren auf die verarbeiteten Daten dargestellt.

Wo in Abbildung 5 noch eine große transparent rot eingefärbte Fläche das von Hyperscalern besetzte Bündel von Funktionen und Rollen markiert, sind in
Abbildung 7 die einzelnen Rollen und Dienste "disassembliert" einzeln dargestellt. Durch die grüne Färbung der Umrandungen dieser einzelnen Dienste ist außerdem deren zumindest teilweise quelloffene Bereitstellung angedeutet.

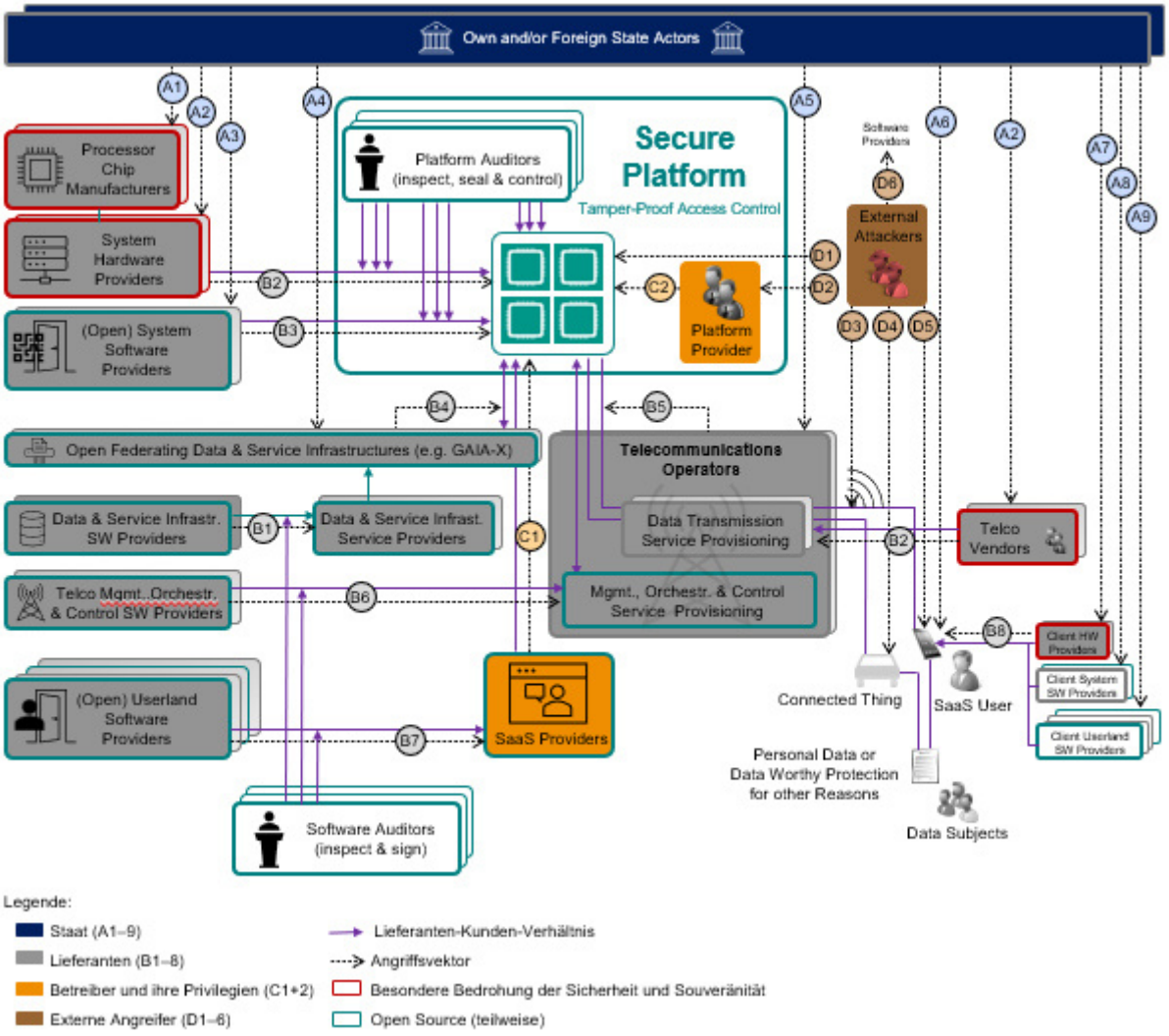

Abbildung 7: Umfeld der Referenzarchitektur: Sichere Plattform durch Transparenz, Disassemblierung (Zerlegung, um Austauschbarkeit von Teilsystemen zu schaffen), Inspektion und Versiegelung für souveräne und vertrauliche Datenverarbeitung (Sovereign Confidential Computing).

Eine derartige Zerlegung der oligopolen Bündel ist notwendig, um Offenheit und Transparenz für Wettbewerb und weniger einseitige Machtverteilung in den Märkten zu schaffen. Neben anderen Initiativen bemüht sich GAIA-X um eine solche Öffnung der Märkte durch eine föderierte Infrastruktur mit einem hierfür geeigneten Identitätsmanagement und der Förderung von Open-Source-Software-Entwicklungen für die dazu passende Daten- und Service-Infrastrukturdienste.

Die Computing Platforms, auf denen die Daten- und Service-Infrastrukturdienste ablaufen und auf denen die SaaS-Dienste der SaaS-Provider vermittels der Daten- und Service-Infrastrukturdienste ebenfalls ausgeführt werden sollen, müssen sich entsprechend der Anforderungen und Definitionen solcher Daten- und Dienste-Infrastrukturen mit den Attributen ihrer Identität als Secure Platform ausweisen. Dann werden sie ggf. von den SaaS-Providern oder beispielsweise auch von den Telekommunikationsnetzbetreibern, zur Deckung des erhöhten oder hohen Schutzbedarfs ihrer Anwendung ausgewählt.

In Abbildung 8 ist die Referenzarchitektur für solche Secure Platforms mit seinen Komponenten und deren Zusammenspiel gezeigt.

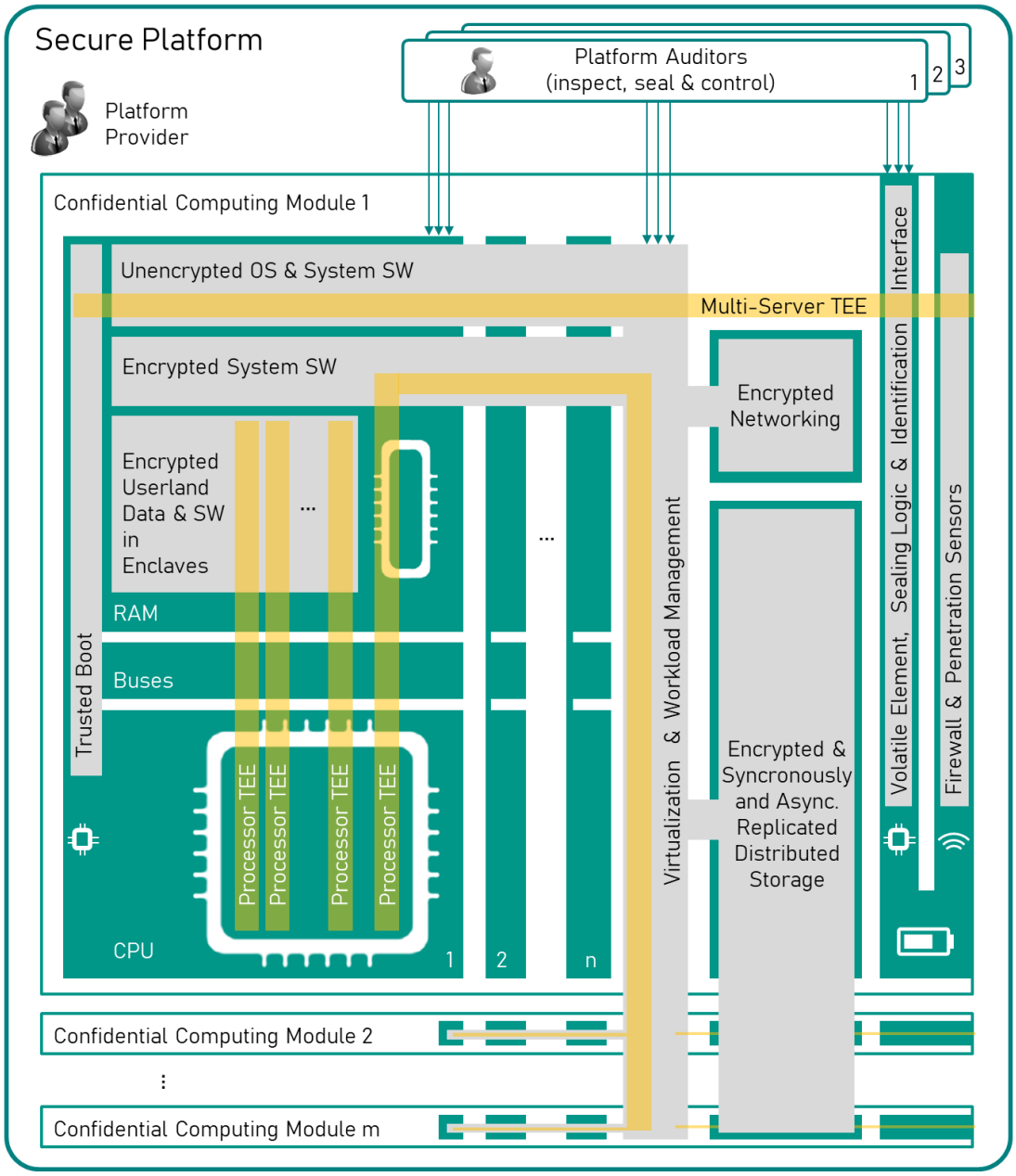

Abbildung 8: Referenzarchitektur einer "Secure Platform" für "Sovereign Confidential Computing"

Im Herrschaftsbereich eines Platform Providers ist ein "Confidential Computing Modul 1" detailliert mit einem das Modul umfassenden Rahmen und den darin enthaltenen Funktionsblöcken. Darunter iterieren beliebig weitere, ggf. identisch aufgebaute Confidential Computing Module der Anzahl 2 bis m.

Innerhalb des Confidential Computing Moduls 1 dominiert eine erste ggf. serverähnlich aufgebaute Recheneinheit 1 mit Hardware, beispielsweise bestehend aus CPU, Bussen, RAM und Trusted-Boot-Elementen (wo passend, mit einem Chip-Symbol versehen). Daneben sind weitere, ggf. identisch aufgebaute Recheneinheiten in Anzahl 2 bis n angedeutet.

Zusätzlich gehört zum Confidential Computing Modul 1 die Vernetzung der Komponenten mit Kabeln, Routern und Switches, die hier lediglich mit dem Block "Encrypted Networking" repräsentiert ist, sowie der zu Modul 1 gehörende Anteil des verteilten, mit den etablierten Mechanismen synchron und asynchron zu und von anderen Modulen replizierenden Speichersystems.

Außerdem gehören die am rechten Rand des Confidential Computing Moduls 1 gezeichneten Komponenten "Penetration Sensors", "Volatile Element", "Sealing Logic", "Identification Interface" und eine Stützbatterie für

diese Komponenten ebenfalls zum Confidential Computing Modul 1. Diese Komponenten sind am Rand des das Modul 1 umfassenden Rahmens angelehnt, um anzudeuten, dass die Penetrationssensorik direkt den Perimeter und ggf. die Anordnung der Komponenten innerhalb des Moduls physisch überwacht. Dies kann beispielsweise mit Radiotechnik im 50 bis 100GHz-Bereich erfolgen (Radio-Symbol), einer Frequenz welche die andere Elektronik nicht stört. Auch die Identifikationsschnittstelle grenzt direkt an die Außenhaut des Moduls, da sich die Platform Auditoren nach der Inspektion der Hard- und Software und dem Verschließen des Moduls davon überzeugen können müssen, dass die Schnittstelle, der sie ihre, den Audit testierende Teilgeheimnisse anvertrauen, auch wirklich zu der eben zuvor auditierten Hardware gehört. Das Betriebssystem, die Systemsoftware und die Software zur Virtualisierung und dem Workload-Management werden unabhängig von der Modulhardware auditiert, und bei Zufriedenheit der Auditoren signiert.

Zusammen mit dem Open Source Betriebssystem (OS) und der Open Source System Software (System SW) sowie der Open Source Virtualisierungs- und Workload-Management-Software, die jeweils mit den Routern der Vernetzung als auch mit den Kontrolleinheiten des Open Source Speichersystems interagieren müssen, bilden die Confidential Computing Module die Secure Platform.

Mit transparent gelber Farbe sind zum einen die Prozessor-TEE, in denen die Userland-Software ausgeführt wird, als auch die Prozessor-TEE für Teile der System-SW eingezeichnet. Durch die bei der Versiegelung der Prozessor-TEE in eine TEE eingeschriebenen Identitäten können mehrere TEE vertrauensvoll miteinander kommunizieren, was durch die gelben Flächen, die sich bis in andere Confidential Computing Module hinein erstrecken, angedeutet ist. Diese Prozessor-TEE isolieren die Workloads voneinander und die Workloads vom Betriebssystem kryptografisch.

Außerdem ist die quer vom Trusted Boot des Confidential Computing Moduls 1 bis zur Penetrationssensorik des Moduls 1 verlaufende Multi-Server-TEE-Funktionalität transparent gelb markiert. Diese ist dafür verantwortlich mit dem Mechanismus des "Data Clean-up" die Manipulationssicherheit des Modulaufbaus sicherzustellen.

6.3 Entwurfsempfehlungen

Die im vorliegenden Abschnitt vorgestellten Entwurfsempfehlungen haben zum Ziel, eine Rechnerplattform zu ermöglichen die den Anforderungen in IEC 62443-4-2[17] für alle drei, dort definierten Geräteprofile genügt und somit auch für den Einsatz in kritischen Infrastrukturen geeignet sein kann. Tabelle 2 zeigt eine Übersicht der Anforderungen der IEC 62443-4-2 an die verschiedenen Geräteprofile, sowie welche Anforderungen in IEC 62443-4-2 profilspezifisch, bzw. zwei oder mehr Profilen gemeinsam sind.

13 Embedded device requirements	**14 Host device requirements**	**15 Network device requirements**
		NDR 1.6 - Wireless access management
		NDR 1.13 - Access via untrusted networks
EDR 2.4 - Mobile code	HDR 2.4 - Mobile code	NDR 2.4 - Mobile code
EDR 2.13 - Use of physical diagnostic and test interfaces	HDR 2.13 - Use of physical diagnostic and test interfaces	NDR 2.13 - Use of physical diagnostic and test interfaces
EDR 3.2 - Protection from malicious code	HDR 3.2 - Protection from malicious code	NDR 3.2 - Protection from malicious code
EDR 3.10 - Support for updates	HDR 3.10 - Support for updates	NDR 3.10 - Support for updates
EDR 3.11 - Physical tamper resistance and detection	HDR 3.11 - Physical tamper resistance and detection	NDR 3.11 - Physical tamper resistance and detection
EDR 3.12 - Provisioning product supplier	HDR 3.12 - Provisioning product supplier	NDR 3.12 - Provisioning product supplier

13 Embedded device requirements	14 Host device requirements	15 Network device requirements
roots of trust	roots of trust	roots of trust
EDR 3.13 - Provisioning asset owner roots of trust	HDR 3.13 - Provisioning asset owner roots of trust	NDR 3.13 - Provisioning asset owner roots of trust
EDR 3.14 - Integrity of the boot process	HDR 3.14 - Integrity of the boot process	NDR 3.14 - Integrity of the boot process
		NDR 5.2 - Zone boundary protection
		NDR 5.3 - General purpose, person-to-person communication restrictions

Tabelle 2: Übersicht der Anforderungen für die in IEC 62443-4-2 definierten Geräteprofile

6.3.1 Entwurfsempfehlungen für die Hardwareplattform

Abbildung 9 zeigt ein vereinfachtes, funktionales Architekturmodell eines Rechnersystems, welches den Betrachtungen im vorliegenden Abschnitt zugrunde liegt. Die Pfeile in Abbildung 9 stellen den Fluss von Steuerungs- und Statusinformationen dar. Auf eine Darstellung der Programm- und Arbeitsdaten, welche beim Ablauf einer Anwendung auf der Plattform zum Einsatz kommen, wurde aus Gründen der Übersichtlichkeit verzichtet.

Beachtenswert ist hierbei, dass das funktionale Architekturmodell aus Abbildung 9 auf verschiedenen Ebenen in fraktaler, bzw. rekursiver Art und Weise wiederholt auftreten kann. So können einzelne Funktionen in ihrem Inneren wiederum ebenso aufgebaut sein.

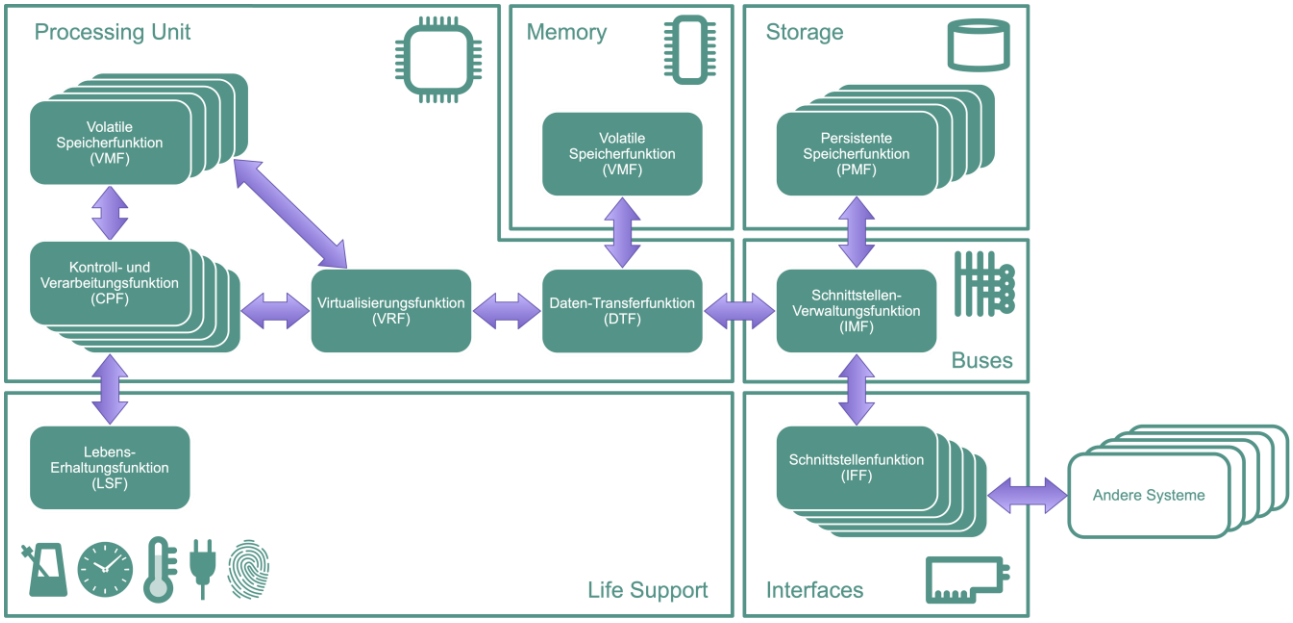

Abbildung 9: Vereinfachts, funktionales Architekturmodell eines Rechnersystems

Die in Abbildung 9 gezeigten, funktionalen Blöcke sind:

- Kontroll- und Verarbeitungsfunktion (CPF, engl.: Control and Processing Function): Ausführungseinheit, die die Anwendungssoftware und das Betriebssystem ausführt, und andere Funktionen steuert.
- Volatile Speicherfunktion (VMF, engl.: Volatile Memory Function): Flüchtiger, schnell zugreifbarer Speicher.
- Virtualisierungsfunktion (VRF, engl.: ViRtualization Function): Isoliert die verschiedenen Anwendungen untereinander, sowie gegen das Betriebssystem.
- Daten-Transferfunktion (DTF, engl.: Data Transfer Function): Transferiert Daten zwischen den verschiedenen Speicher- und Schnittstellenfunktionen.

- Persistente Speicherfunktion (PMF, engl.: Persistent Memory Function): Nichtflüchtiger, langsam zugreifbarer Speicher.
- Schnittstellen-Verwaltungsfunktion (IMF, engl.: Interface Management Function): Stellt eine uniforme Schnittstelle für über verschiedene, interne Bussysteme angebundene Schnittstellen-, und Speicherfunktionen bereit.
- Schnittstellenfunktion (IFF, engl.: InterFace Function): Stellt an einem internen Bussystem eine, in der Regel standardisierte, Übertragungs-Schnittstelle zur Verbindung mit anderen, externen Systemen bereit.
- Lebens-Erhaltungsfunktion (LSF, engl.: Life Support Function): Wie in vielen, komplexen, technischen Systemen sind zum Hochfahren und zum Betrieb des Hauptaggregats verschiedene Nebenaggregate erforderlich. Ebenso verhält es sich bei einem Rechnersystem.
  Im Einzelnen sind dies:
  - Arbeitstakt: Steuert, und synchronisiert alle Verarbeitungsvorgänge im gesamten Rechnersystem.
  - Uhrzeit und Kalenderdatum: Wird zur Synchronisation mit externen Systemen und für Protokollierungszwecke verwendet.
  - Thermische Steuerung: Kontrolliert die Zu- und Abfuhr von Wärme, um das System stets innerhalb der zulässigen Temperaturgrenzen betreiben zu können.
  - Energiesteuerung: Kontrolliert die Versorgung des Systems mit Betriebsenergie, so dass der Energieverbrauch der Systemauslastung angepasst und dadurch minimiert wird.
  - Identität: Bietet eine eineindeutige, nicht änderbare Identität eines jeden Rechnersystems, ähnlich einer Seriennummer.

6.3.1.1 Gehäuse, Leiterplatte und Verbindungstechnik

Im Markt haben sich präventive Maßnahmen zur Absicherung des Gehäuses gegen einen physischen Angriff (zum Beispiel festeres Material, Spezialschrauben) nur in wenigen Anwendungsfällen durchgesetzt. Neben den hohen Kosten und dem hohen Gewicht, ist es regelmäßig erforderlich, das Gehäuse zu Wartungs- oder Reparaturzwecken öffnen zu können. Die hierfür nötigen Verfahren und Werkzeuge müssen daher für das Wartungspersonal dokumentiert, beziehungsweise vorhanden sein. Da sich die Verbreitung solcher Informationen und Werkzeuge in der Praxis kaum verhindern lässt[18], erschweren präventive Maßnahmen allein einen physischen Angriff auf ein Gerät und sind damit in der Regel nicht ausreichend. Daher wird in der Referenzarchitektur in Abschnitt 5.2. die so genannte Multi-Server TEE als wichtiges Entwurfsmuster berücksichtigt.

Tabelle 3 führt ergänzend eine Reihe weiterer reaktiver Entwurfsmuster für die Gehäusekonstruktion auf. Reaktive Maßnahmen sind regelmäßig deutlich kostengünstiger als präventive. Mehrere reaktive Maßnahmen lassen sich daher leicht kombinieren, was zusätzlich den Vorteil bietet, dass ein physischer Angriff kombinatorisch mit der Anzahl der reaktiven Maßnahmen erschwert wird.

Entwurfsmuster	**Beschreibung**	**Klassifizierung nach IEC 62443-4-2**
Sicherheitsaufkleber	- Überprüfung nur vor Ort möglich. - Nur bei korrektem Anbringen wirksam. - Erfordert spalt- bzw. senkenfreien Stoß der Gehäuseteile an den Anbringungsstellen. - Materialalterung bedingt regelmäßigen Austausch.	EDR/HDR/NDR 3.11: **SL-C 2**
kanalisierte Gehäuseöffnungen	- Erschwert Einführen von Werkzeugen. - Kanalquerschnitt minimieren. - Wenn möglich Kanal abknickend auslegen.	EDR/HDR/NDR 3.11: **SL-C 2**
Abstandserkennung Gehäuseteile	- Abstrahlung aktiver Systeme kann u.U. von außen detektiert werden. - Untere Erkennungsschwelle bei Überlappung der Gehäuseteile berücksichtigen. - Gezielte Beeinflussung von außen ggf. möglich. - Benötigt Energieversorgung.	EDR/HDR/NDR 3.11: **SL-C 2**

Entwurfsmuster	Beschreibung	Klassifizierung nach IEC 62443-4-2
Bewegungsmelder Innenraum	- Abstrahlung aktiver Systeme kann u.U. von außen detektiert werden. - Fehlalarme durch eintretende Verschmutzung möglich. - Kanalisierte Gehäuseöffnungen können nicht überwacht werden. - Gezielte Beeinflussung von außen ggf. möglich. - Benötigt Energieversorgung.	EDR/HDR/NDR 3.11: **SL-C 2**
Helligkeitsmessung Innenraum	- Nur effektiv, wenn neben sichtbarem Spektrum auch UV, und Infrarot überwacht werden. - Kanalisierte Gehäuseöffnungen können nicht überwacht werden. - Gezielte Beeinflussung von außen ggf. möglich. - Benötigt Energieversorgung.	EDR/HDR/NDR 3.11: **SL-C 2**
Beschleunigungsmessung	- Untere Erkennungsschwelle berücksichtigen. - Bei ein- oder zweiachsiger Messung Erkennung von Vibrationen oder Stößen. - Ab dreiachsiger Messung zusätzlich Lageerkennung möglich. - Gezielte Beeinflussung von außen ggf. möglich. - Benötigt Energieversorgung.	EDR/HDR/NDR 3.11: **SL-C 2**
Alarmierungsschnittstelle	- Aufsichtspersonal kann sofort handeln. - Benötigt auslösendes Moment. - Alarmverbindung muss ebenfalls überwacht werden. - Benötigt Energieversorgung.	EDR/HDR/NDR 3.11: **SL-C 3, 4**

Tabelle 3: Entwurfsmuster für die Gehäusekonstruktion

Zur Vorbereitung eines Angriffs wird oft versucht die Konstruktion und Arbeitsweise eines Geräts zu analysieren (engl.: reverse engineering), um daraus Erkenntnisse über aussichtsreiche Angriffsmethoden und Angriffspunkte zu gewinnen. Hierzu kann zum Beispiel ein Gerät neu oder gebraucht erworben werden; auch werden regelmäßig Geräte von Elektroschrott-Sammelstellen entwendet, da solche Einrichtungen meist nicht stark überwacht werden. Tabelle 4 führt daher eine Reihe präventiver Entwurfsmuster für die Leiterplattenkonstruktion auf, die eine Analyse erschweren sollen.

Entwurfsmuster	Beschreibung	Klassifizierung nach IEC 62443-4-2
Multilayer PCB	- Äußerste Lagen dienen als Abschirmlagen, um Abstrahlung zu reduzieren. - Äußerste Lagen tragen Versorgungsspannungen, Masse. - Datenverkehr nur auf inneren Lagen, und verborgenen Vias. - Testpunkte auf einen gesonderten Streifen legen, der vor Einbau abgetrennt wird.	EDR/HDR/NDR 3.11: **SL-C 2**
Verbindungstechnik	- Wo immer verfügbar, BGA-Varianten verwenden (auch Steckverbinder). - Bei Steckverbindern muss die elektrische Verbindung schon getrennt sein, wenn sich die Steckergehäuse noch überlappen.	EDR/HDR/NDR 3.11: **SL-C 2**
Komponentenidentifizierung	- Bauteilbeschriftungen abschleifen, um Rückgriff auf Datenblätter zu erschweren.	EDR/HDR/NDR 3.11: **SL-C 2**
Lebenserhaltungssysteme	- Versorgungsspannungen, und Takte gegen schnelle Änderungen	EDR/HDR/NDR 3.11: **SL-C 2**

Entwurfsmuster	Beschreibung	Klassifizierung nach IEC 62443-4-2
	stabilisieren.	
Alarmierungsschnittstelle	- Bei plötzlichen Veränderungen von Versorgungsspannungen, Takte, oder Temperatur, Alarm auslösen und/oder abschalten.	EDR/HDR/NDR 3.11: **SL-C 3, 4**

Tabelle 4: Entwurfsmuster für die Leiterplattenkonstruktion

6.3.1.2 Rechner-Systemarchitektur

Etliche der Komponenten in einem Rechnersystem enthalten physische Schnittstellen, die dazu dienen, mit anderen Systemen zu kommunizieren. Diese können Fertigungs- und Testzwecken dienen (zum Beispiel UART, JTAG), oder für Anwendungen bestimmt sein (z.B. Bluetooth, NFC). Oft enthält eine Komponente mehrere solcher Schnittstellen (z.B. WiFi-Modul, das auch eine Bluetooth-Funktion enthält). Bei einem durchschnittlichen Rechnersystem kann die Anzahl der enthaltenen Schnittstellen leicht zweistellige Werte erreichen. Meist bleibt der überwiegende Teil davon jedoch ungenutzt und somit beim Systementwurf auch unbeachtet. Schnittstellen die Fertigungs- und Testzwecken dienen sind oft sogar ohne jede Softwarefunktion nutzbar. Kommunikationsschnittstellen für Anwendungen werden von modernen Betriebssystemen in der Regel im Betrieb erkannt, und in einen betriebsbereiten Zustand gebracht; so sind diese beim Entwurf unbeachteten Schnittstellen im Betrieb dennoch betriebsbereit. Unbeachtete Schnittstellen lassen sich daher für Angriffe nutzen; sie bieten sogar noch den Vorteil, dass der Angriff freie Ressourcen auf dem Zielsystem benutzt und daher dessen Verhalten bei der Ausführung legitimer Anwendungen zunächst nicht beeinflusst. Tabelle 5 führt daher eine Reihe präventiver Entwurfsmuster für die Behandlung von physischen Schnittstellen auf, die deren Verwendung für Angriffe erschweren sollen.

Entwurfsmuster	Beschreibung	Klassifizierung nach IEC 62443-4-2
Deaktivierung der Schnittstellen für Fertigung und Tests	- Auditierung aller Datenblätter, um sicher zu stellen, dass keine Schnittstellen unbeachtet bleiben. - Irreversible Deaktivierung (z.B. Fuse) am Ende der Produktion, und Dokumentation der Seriennummer.	EDR/HDR/NDR 2.13: **SL-C 2** EDR/HDR/NDR 3.11: **SL-C 2**
Kontrolle der Schnittstellen für Fertigung und Test durch Authentifizierung	- Auditierung aller Datenblätter, um sicher zu stellen, dass keine Schnittstellen unbeachtet bleiben. - Authentifizierung mit Passwort, welches auf einer eineindeutigen, nicht änderbaren Identität des Systems oder der Komponente basiert. - Protokollierung aller Authentifizierungen, und Authentifizierungsversuche.	EDR/HDR/NDR 2.13: **SL-C 3, 4** EDR/HDR/NDR 3.11: **SL-C 3, 4**
Deaktivierung unbenutzter Kommunikationsschnittstellen	- Auditierung aller Datenblätter, um sicher zu stellen, dass keine Schnittstellen unbeachtet bleiben. - Deaktivierung der unbenutzten Kommunikationsschnittstellen in der Konfiguration des Betriebssystems, und ggf. weiterer Software (z.B. BMC, Boot-Loader).	EDR/HDR/NDR 2.13: **SL-C 2** EDR/HDR/NDR 3.11: **SL-C 2** CR 7.7: **SL-C 1, 2, 3, 4**

Tabelle 5: Entwurfsmuster für physische Schnittstellen

Können die in Kapitel 6.3.1.1 beschriebenen Entwurfsmuster zur Erschwerung des Zugangs zu Datensignalen nicht vollständig umgesetzt werden, so könnten bei einem Angriff zum Beispiel die Daten, die zwischen der Processing Unit und den volatilen, beziehungsweise persistenten Speicherfunktionen ausgetauscht werden (vgl. Abbildung 9), durch Kontaktierung der entsprechenden Leiterbahnen mitgehört werden. Ein persistenter Speicher ließe sich sogar aus dem angegriffenen System entfernen und an ein anderes System anschließen,

um dort dessen Inhalte komfortabel auslesen zu können. Tabelle 6 führt daher eine Reihe präventiver Entwurfsmuster für die Benutzung von Speicherfunktionen auf, die deren Verwendung für Angriffe erschweren sollen.

Entwurfsmuster	Beschreibung	Klassifizierung nach IEC 62443-4-2
Ver-/Entschlüsseln von Daten in der DTF (vgl. Abbildung 9)	- Ver-/Entschlüsselungsfunktion, sowie volatiler und persistenter Schlüsselspeicher ist in der DTF vorhanden. - Neue Schlüssel für volatile Speicherfunktionen werden bei jedem Systemstart von der DTF erzeugt, und dort volatil gespeichert. - Schlüssel für persistente Speicherfunktionen werden von der DTF einmalig erzeugt, und dort persistent hinterlegt. - Kein Zugriff auf Schlüssel von außerhalb der DTF, sondern Referenzieren über Schlüssel-Kennungen. - Die CPF kann Erzeugen neuer Schlüssel bei der DTF anfordern, und diese dann für Kommunikation mit den IFF einsetzen. - Beim Systemstart muss sich die CPF bei der DTF unter Verwendung der eineindeutigen, nicht änderbaren Identität des Systems (vgl. LFS) authentifizieren.	EDR/HDR/NDR 3.11: **SL-C 2** EDR/HDR/NDR 3.14: **SL-C 1** (teilweise) CR 4.1: **SL-C 1, 2, 3, 4**

Tabelle 6: Entwurfsmuster für die Benutzung von Speicherfunktionen

Die bisher beschriebenen Entwurfsmuster hatten zum Ziel, das Auslesen von Informationen aus dem angegriffenen System zu erschweren, also mithin das Vertrauen in dessen Diskretion zu erhöhen. Die weiteren, ab hier im Folgenden beschriebenen Entwurfsmuster zielen hingegen darauf ab, das Verändern des Verhaltens des angegriffenen Systems zu erschweren, d.h. das Vertrauen in dessen Integrität zu erhöhen. Im Rahmen des vorliegenden Abschnitts zur Rechner-Systemarchitektur bedeutet dies den Startvorgang des Systems so abzusichern, dass vom Einschalten bis zur Betriebsbereitschaft des Betriebssystems kein Programmcode ausgeführt werden kann, der nicht vom Hersteller des Systems als vertrauenswürdig markiert ist. Tabelle 6 führt daher eine Reihe präventiver Entwurfsmuster für die Integrität des Systemstarts auf, die Angriffe auf den Startvorgang erschweren sollen.

Entwurfsmuster	Beschreibung	Klassifizierung nach IEC 62443-4-2
Secure Boot	- Stufenweiser Startvorgang entsprechend Abbildung 10. - Die Daten jeder Stufe liegt Geräte-individuell verschlüsselt im persistenten Speicher vor (vgl. z.B. Tabelle 6). - Authentifizierung und Autorisierung der (entschlüsselten) Daten der nächsten Stufe vor deren Ausführung anhand ihrer kryptographischen Signatur. - Die kryptographische Signatur wurde mit dem privaten, asymmetrischen Schlüssel des Herstellers des Systems erstellt. - Der öffentliche Schlüssel des Herstellers des Systems ist für die CPF unveränderbar auf dem System gespeichert.	EDR/HDR/NDR 3.2: **SL-C 1, 2, 3, 4** (teilweise) EDR/HDR/NDR 3.14: **SL-C 1, 2, 3, 4**

Tabelle 7: Entwurfsmuster für die Integrität des Systemstartvorgangs

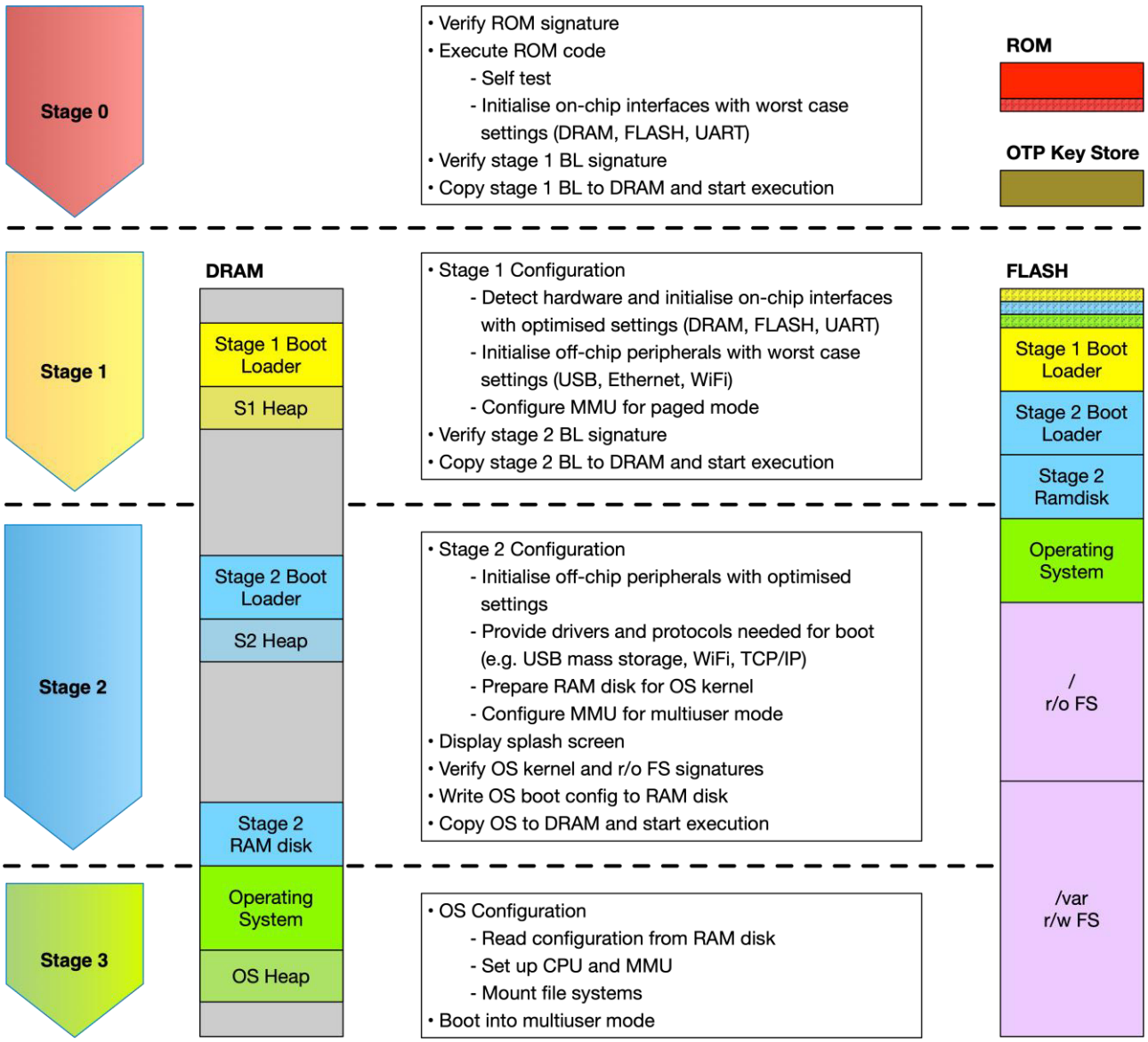

Abbildung 10: Konzeptionelles Prozessmodell eines abgesicherten Systemstarts

6.3.1.3 Prozessortechnik

Da die im Massenmarkt typischerweise eingesetzten Plattformen zur Software-Entwicklung den damit erstellten Anwendungen direkten Zugriff auf die Hardware der Prozessoreinheit ermöglichen ("bare metal ISA"), ist es für die Robustheit eines Systems gegenüber Angriffen von entscheidender Bedeutung, die Prozessoreinheit in einer Konfiguration zu betreiben, die Angriffe erschwert. Tabelle 8 führt daher eine Reihe präventiver Entwurfsmuster für den Einsatz von Prozessoreinheiten auf, die dies zu erreichen helfen.

Entwurfsmuster	Beschreibung	Klassifizierung nach IEC 62443-4-2
Address Space Layout Randomisation (ASLR)	- Bei Anforderung von neuem Speicher wird nicht der nächste, freie Block verwendet, sondern ein zufällig ausgewählter. - Führt dazu, dass Codeabschnitte und Daten bei jedem Programmstart anders im Speicher verteilt werden - Auslesen von im Programmcode, oder in Daten verankerter Geheimnisse wird dadurch erschwert.	EDR/HDR/NDR 3.2: **SL-C 1, 2, 3, 4** CR 3.3: **SL-C 1, 2, 3, 4** (teilweise)
Write xor Execute (W^E)	- Speicherblöcke, die Programmcode enthalten, können nicht verändert werden. - Gegebenenfalls leicht erhöhter Speicherbedarf.	EDR/HDR/NDR 3.2: **SL-C 1, 2, 3, 4** CR 3.3: **SL-C 1, 2, 3, 4** (teilweise)
Abschalten von Hyperthreading	- Bei manchen Produktfamilien von Prozessoreinheiten werden prozessornahe, volatile Speicher beim Hyperthreading nicht gelöscht. - Eine Anwendung kann so Daten einer anderen Anwendung aus diesen Speichern auslesen. - Falls für ein betroffenes Produkt keine Fehlerbehebung vom Hersteller vorliegt, sollte Hyperthreading nicht verwendet werden.	EDR/HDR/NDR 3.2: **SL-C 1, 2, 3, 4** CR 3.3: **SL-C 1, 2, 3, 4** (teilweise)

Tabelle 8: Entwurfsmuster für den Einsatz von Prozessoreinheiten

Aus demselben Grund sollten begleitend auch entsprechende Mechanismen in den Werkzeugen zur Software-Erstellung zum Einsatz kommen. Tabelle 9 führt daher eine Reihe präventiver Entwurfsmuster für Mechanismen auf, die auf niedriger Ebene zum Einsatz kommen, über welche die allermeisten Werkzeuge zur Software-Erstellung verfügen und die helfen, Angriffe zu erschweren.

Entwurfsmuster	Beschreibung	Klassifizierung nach IEC 62443-4-2
Position-Independent Executable (PIE)	- Ermöglicht die Programmausführung unabhängig von der Verteilung einzelner Codeabschnitte im Speicher. - Führt in der Praxis dazu, dass Codeabschnitte bei jedem Programmstart anders im Speicher verteilt werden. - Auslesen im Programmcode verankerter Geheimnisse wird dadurch erschwert. - Gegebenenfalls leicht verringerte Ausführungsgeschwindigkeit.	EDR/HDR/NDR 3.2: **SL-C 1, 2, 3, 4** CR 3.3: **SL-C 1, 2, 3, 4** (teilweise)
Stack Canaries	- Erlaubt das Erkennen von fehlerhaften Eingabedaten und ermöglicht Abwehr von Fehlverhalten der Anwendung. - Leicht erhöhter Speicherbedarf.	EDR/HDR/NDR 3.2: **SL-C 1, 2, 3, 4** CR 3.3: **SL-C 1, 2, 3, 4** (teilweise)

Entwurfsmuster	Beschreibung	Klassifizierung nach IEC 62443-4-2
Relocatable Read-only (RELRO)	- Bewirkt, dass alle dynamischen Abhängigkeiten einer Anwendung bereits bei deren Start geladen, und unveränderbar im Speicher abgelegt werden. - Erschwert Einschleusen von unerwünschtem Code erheblich. - Wird oft in zwei Varianten umgesetzt ("voll", und "partiell"), von denen nur die "volle" Variante signifikanten Schutz bietet. - Leicht verringerte Geschwindigkeit beim Anwendungsstart.	EDR/HDR/NDR 3.2: **SL-C 1, 2, 3, 4** CR 3.3: **SL-C 1, 2, 3, 4** (teilweise)

Tabelle 9: Entwurfsmuster für Mechanismen von Werkzeugen zur Software-Erstellung auf niedriger Ebene

Wie in Kapitel 6.1 bereits erwähnt, sind die Abschirmung verschiedener, auf einem System ausgeführter Anwendungen gegeneinander, sowie die Abschirmung des Betriebssystems gegenüber den Anwendungen für einen sicheren und zuverlässigen Betrieb entscheidend. Die heute den Markt für Server-, Desktop-, und Laptop-Systeme dominierenden Prozessorarchitekturen stellen nur beschränkte Werkzeuge hierfür zur Verfügung. Dies hat historische Gründe. Mit der Verfügbarkeit von Prozessoren in Form von integrierten Halbleiterschaltungen zu Beginn der 1970er Jahre stieg die Leistungsfähigkeit auch billigerer Systeme, daher begannen Bemühungen die für den Betrieb von Mehrplatzsystemen (die bis dahin nur auf Mainframe-Systemen verfügbar waren) erforderlichen Funktionen in Software nachzubilden. So sollten der Komfort und die Vorteile von Mehrplatzsystemen auch mit billigerer Hardware zugänglich gemacht werden. Mit der Aufnahme solcher, in Software umgesetzter Mehrplatz-Eigenschaften in das Unix Betriebssystem ab Ende der 1970er Jahre fand dieser Ansatz schnell zunehmende Verbreitung. Alle heute im Massenmarkt für Server-, Desktop-, und Laptop-Systeme verwendeten Betriebssysteme basieren daher immer noch auf diesem Entwurfsmuster: Ein Mehrplatzsystem mit minimalen Anforderungen an den Prozessor, dass die übrigen, benötigten Funktionen in Software umsetzt.

Diese Entwicklung war bedingt durch die technischen und wirtschaftlichen Randbedingungen, welche die Entwicklung komplexer, integrierter Halbleiterschaltungen während der zweiten Hälfte des zwanzigsten Jahrhunderts bestimmten: Die Entwurfs-, und Entwicklungswerkzeuge waren sehr teuer und unterstützten in ihrer Leistungsfähigkeit den Umgang mit Komplexität nur eingeschränkt. Die Entwicklung von Prozessoren in Form von integrierten Halbleiterschaltungen war daher sehr teuer und von sehr hohem Bedarf an höchstqualifiziertem Personal gekennzeichnet. Eine Verlagerung der Funktionen, die ab den 1970er Jahren - wie beschrieben - in die Software des Betriebssystems gewandert waren, zurück in die Hardware des Prozessors (wo diese in Mainframe-Systemen von Anfang an umgesetzt wurden; Typ-1 Hypervisor, vgl. Abbildung 11) war daher ausgeschlossen. In jüngerer Vergangenheit hat sich diese Entwicklung durch verstärkten Einsatz von Virtualisierung noch verstärkt. Die hierdurch zahlreich entstandenen Cloud-Angebote (z.B. virtuelle Server, oder Container) sind daher fast ausschließlich als Typ-2 Hypervisoren umgesetzt (vgl. Abbildung 11).

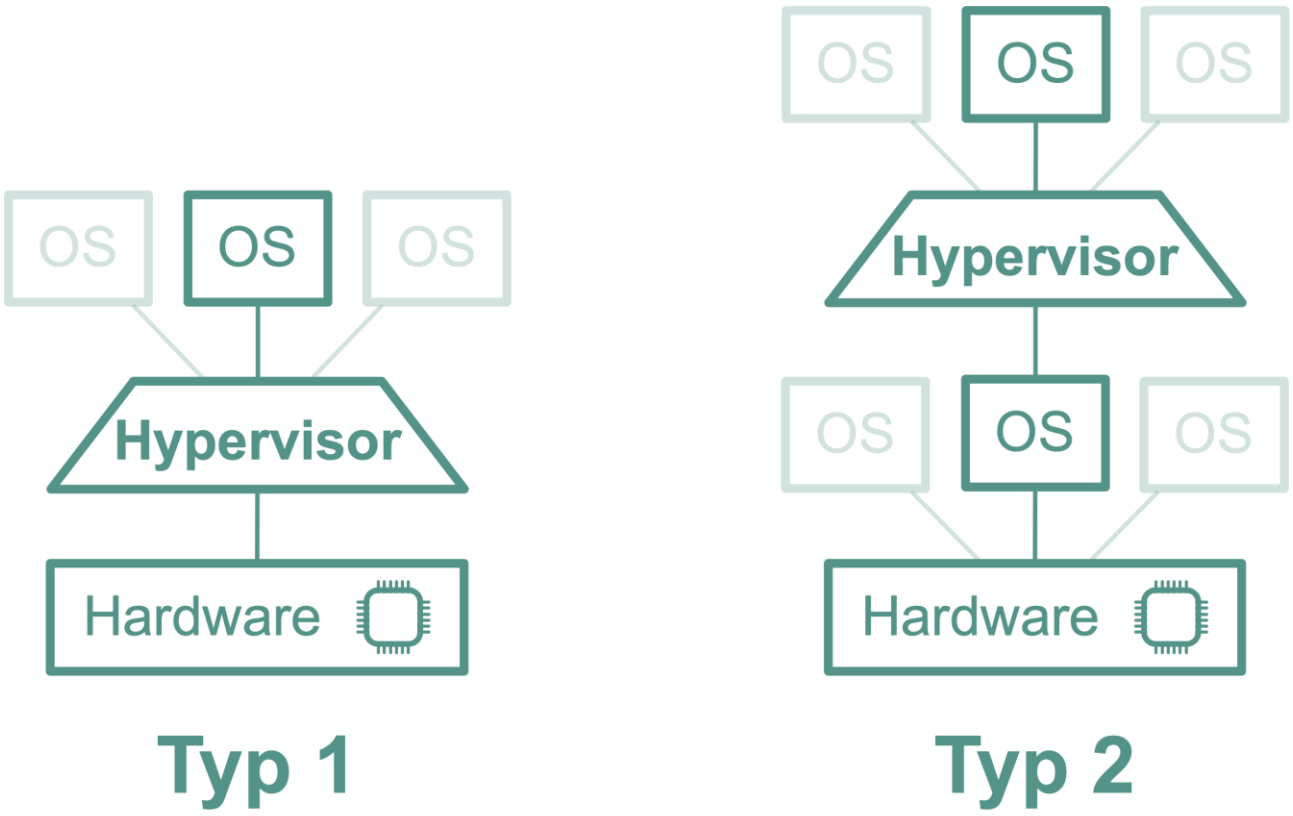

Abbildung 11: Klassifizierung von Hypervisor-Implementierungen

Damit in Kapitel 5.2 genannten Zieleigenschaften von zukünftigen Rechnersystemen regelmäßig intrinsisch erreicht werden können, sollten die nächsten Generationen von Prozessoreinheiten nicht mit dem Primärziel der Ausführbarkeit existierender Software-Systeme, sondern mit Blick auf die sichere Trennung der Anwendungen, und die daraus resultierende Resilienz des Gesamtsystems entworfen werden. Tabelle 10 führt daher eine Reihe präventiver Entwurfsmuster für künftige Generationen von Prozessoreinheiten auf, die es erlauben sollen, intrinsisch sichere, und resiliente Rechnersysteme zu erstellen.

Entwurfsmuster	Beschreibung	**Klassifizierung nach IEC 62443-4-2**
Dynamic Logical Partitions (DLPAR)	- unterteilt die vorhandenen Hardware-Ressourcen (Prozessoreinheiten, Speicher, Schnittstellen, Ein-/Ausgabe-Kapazitäten) in logische Partitionen, die völlig isoliert voneinander arbeiten. - Ressourcen können zur Kapazitätsanpassung während des Betriebs aus einer logischen Partition herausgenommen, und einer anderen hinzugefügt werden.	CR 7.1: **SL-C 1, 2, 3, 4** CR 7.2: **SL-C 1, 2, 3, 4**

Tabelle 10: Entwurfsmuster für künftige Generationen von Prozessoreinheiten

Ergänzende Entwurfsmuster für die Software-Architektur solcher künftigen Systeme werden in Kapitel 6.3.2.2 vorgestellt.

6.3.1.4 Netzwerktechnik

Da Netzwerk-Komponenten aus technischer Sicht ebenfalls Rechnersysteme sind, können alle im vorliegenden Dokument vorgestellten Entwurfsmuster auf sie ebenfalls angewendet werden.

Im vorliegenden Abschnitt soll auf ein zusätzliches Entwurfsmuster eingegangen werden, das in der Auslegung der Netzwerk-Architektur als "System von Systemen" Anwendung finden sollte. Da mit der Anzahl Funktionen eines Systems auch dessen Komplexität steigt, nimmt damit unausweichlich auch die Anzahl der wahrscheinlich in der Implementierung vorhandenen Fehler (und damit Schwachstellen) zu. Daher sind regelmäßig sowohl Netzwerk-Komponenten der Infrastruktur (Router, Switches, Firewalls, usw.), als auch Netzwerk-Schnittstellen mit Zusatzfunktionen (engl.: smart network interface controller, Abk. smartNIC) Ziele von Angriffen.

Das Entwurfsmuster, mit dem dieser Gefahr entgegengewirkt werden soll, besteht darin, die implementierten Funktionen jedes Geräts auf das unbedingt notwendige zu reduzieren, um damit dessen Komplexität, und somit auch die Wahrscheinlichkeit des Vorhandenseins von Schwachstellen, zu minimieren. Sind komplexere Funktionen unabdingbar, so sollten diese stets nur auf Systemen implementiert sein, die engmaschig überwacht und regelmäßig gewartet werden.

Bei der Beschaffung von Netzwerk-Komponenten sollte darauf geachtet werden, nicht "Leistungsfähigkeit auf Vorrat" zu beschaffen, d.h. nicht eine bessere Version eines Produktes zu wählen, weil man eine Zusatzfunktion oder höhere Kapazität vielleicht in der Zukunft nutzen möchte. Auch die Implementierungen unbenutzter oder abgeschalteter Funktionen enthalten Schwachstellen, die sich gegebenenfalls für einen Angriff nutzen lassen. Bei großer Unsicherheit über die voraussichtliche Entwicklung der Anforderungen hinsichtlich Funktionalität und Kapazität einer Netzwerk-Komponente, kann an Stelle von Kauf, die Wahl eines Miet- oder Mietkauf-Modells die Zeitspanne überbrücken helfen, bis genügend eigene Erfahrungen vorliegen.

6.3.2 Entwurfsempfehlungen für Betriebssystem und Virtualisierung

6.3.2.1 Angepasster Einsatz bestehender Systeme mit Typ-2 Virtualisierungsprozessortechnik

Wie in Kapitel 6.3.1.3 ausgeführt, setzt die überwiegende Mehrheit derzeit angebotener Cloud- und Container-Dienste Typ-2 Hypervisoren ein. Tabelle 11 führt daher eine Reihe präventiver Entwurfsmuster für den sicheren Einsatz von Typ-2 Hypervisoren auf.

Entwurfsmuster	**Beschreibung**	**Klassifizierung nach IEC 62443-4-2**
Unikernel-Container	- Die Anwendung wird auf einem leeren Container entwickelt. - Jegliche benötigte Funktion (Ein-/Ausgabe, Netzwerk, Multi-Threading, usw.) wird durch Code implementiert, der über die Entwicklungsumgebung eingebunden wird (z.B. sog. Unikernel). - Testabdeckung der Anwendung = Testabdeckung des Containers. - Es laufen keine nicht benötigten, und daher eventuell schlecht oder gar nicht konfigurierten, Dienste. - Gegebenenfalls erhöhter Entwicklungsaufwand.	EDR/HDR/NDR 3.2: **SL-C 1, 2, 3, 4** CR 6.2: **SL-C 2, 3, 4** (teilweise) CR 7.7: **SL-C 1, 2, 3, 4**

Entwurfsmuster	Beschreibung	Klassifizierung nach IEC 62443-4-2
Sandbox	- Beschränkte Umgebung, in welcher der Zugriff der Anwendung auf Ressourcen (Funktionen des Betriebssystem-Kerns, Netzwerk, persistente Speicher, usw.) vom Betriebssystem kontrolliert und begrenzt wird. - Welche Ressourcen zugänglich sein dürfen, wird von den EntwicklerInnen der Anwendung als Teil des Entwicklungsprozesses festgelegt. - Sandboxen verschiedener Anwendungen sind voneinander isoliert. - Gegebenenfalls erhöhter Entwicklungsaufwand.	EDR/HDR/NDR 3.2: **SL-C 1, 2, 3, 4** CR 4.1: **SL-C 1, 2, 3, 4**

Tabelle 11: Entwurfsmuster für den Einsatz von Typ-2 Hypervisoren

Die beiden Entwurfsmuster aus Tabelle 11 lassen sich vorteilhaft kombinieren, wenn die Container-Instanz selbst in einer Sandbox ausgeführt wird, und nur auf eine Teilmenge der Ressourcen des Systems zugreifen kann. In einer solchen Konstellation ist die Angreifbarkeit der Anwendung selbst schon auf ein Mindestmaß minimiert, da diese nur Code enthält, der auch tatsächlich benötigt wird, und von Tests abgedeckt wird. Sollte ein Angriff auf die Anwendung von außen (z.B. über eine Netzwerkverbindung) dennoch erfolgreich sein, stellt die Sandbox ein weiteres Hemmnis dar, in dessen Überwindung die Angreifer zusätzlichen Aufwand investieren muss.

6.3.2.2 Entwurf von Systemen mit Typ-1 Virtualisierung

Wie im zweiten Teil des Kapitel 6.3.1.3 ausgeführt, lässt die derzeit von den Herstellern verfolgte Strategie der kontinuierlichen Erhöhung der Komplexität der Systeme als Antwort auf Gefährdungen der Sicherheit und der Privatsphäre, nicht auf eine mittel- oder langfristig befriedigende oder tragfähige Entwicklung hoffen. Vielmehr gilt es den Trend umzukehren und stattdessen die Komplexität zu reduzieren. Eine hierfür prädestinierte Plattform stellen Typ-1 Hypervisoren dar, die unmittelbar auf das in Kapitel 6.3.1.3 vorgestellte Entwurfsmuster der Dynamic Logical Partitions (DLPAR) aufbauen. Tabelle 12 führt daher eine Reihe präventiver Entwurfsmuster für den sicheren und flexiblen Entwurf von Virtualisierungs-Systemen auf Basis von Typ-1 Hypervisoren auf.

Entwurfsmuster	Beschreibung	Klassifizierung nach IEC 62443-4-2
ein Typ-1 Hypervisor je DLPAR	- Da die Anzahl der Verfügbaren DLPAR durch die Prozessoreinheiten vorgegeben, und klein ist, wird in jeder DLPAR ein Typ-1 Hypervisor ausgeführt. - Jeder dieser Hypervisoren kann grundsätzlich beliebig viele Betriebssysteme ausführen (nur begrenzt durch vorhandene Ressourcen).	CR 7.1: **SL-C 1, 2, 3, 4** CR 7.2: **SL-C 1, 2, 3, 4** EDR/HDR/NDR 3.10: **SL-C 1**

Entwurfsmuster	Beschreibung	Klassifizierung nach IEC 62443-4-2
Cluster-Betriebssystem	- Das Cluster-Betriebssystem kann mehrere, verteilte Instanzen zu einer logischen Instanz zusammenfassen. - Verteilte Betriebssystem-Instanzen können im laufenden Betrieb weggenommen, oder hinzugefügt werden. - Verteilte Betriebssystem-Instanzen können auch auf verschiedenen Rechnersystemen, und auf Prozessoreinheiten unterschiedlicher Architekturen laufen. - Ganze Anwendungen, und einzelne Arbeitslasten einer Anwendung, können im laufenden Betrieb zwischen Betriebssystem-Instanzen verschoben werden. - Hardware und Software können so ohne Unterbrechung der Anwendung ausgetauscht, aktualisiert, oder deren Leistungsfähigkeit angepasst werden.	CR 7.1: **SL-C 1, 2, 3, 4** CR 7.2: **SL-C 1, 2, 3, 4** EDR/HDR/NDR 3.10: **SL-C 1**
Prozess-VM für Anwendungen	- Anwendungen werden in Objektcode für eine abstrakte Prozessoreinheit übersetzt (anstatt in Maschinencode für die tatsächlich vorhandene Prozessoreinheit). - Zum Ausführen einer Anwendung startet das Cluster-Betriebssystem eine neue Instanz einer virtuellen Maschine, die den Objektcode für die abstrakte Prozessoreinheit ausführen kann, und übergibt dieser den Objektcode zur Ausführung. - Jede Instanz der virtuellen Maschine führt nur eine einzige Anwendung aus, und beendet sich, wenn diese Anwendung beendet wird. - Anwendungen sind unabhängig von der tatsächlich vorhandenen Prozessoreinheit. - Anwendungen bleiben ohne erneutes Übersetzen auf allen zukünftigen Rechnersystemen unverändert ausführbar. - Existieren Werkzeuge, die in den Programmiersprachen C beziehungsweise C++ verfassten Code in Objektcode für die virtuelle Maschine übersetzen können, kann eine Vielzahl von Open Source, und anderer Software zugänglich gemacht werden. - Implementierungsfehler in der virtuellen Maschine (Software) lassen sich sehr viel leichter, und flexibler beheben als Implementierungsfehler in der Prozessoreinheit (Hardware).	

Tabelle 12: Entwurfsmuster für Virtualisierungssysteme auf Basis von Typ-1 Hypervisoren

7 Nächste Schritte

7.1 Referenzimplementierungen

Damit die Anbieter und die Entwickler der Software-Dienste sich in Zukunft leichter entscheiden können Compliance, Sicherheit und Datenschutz nicht nur formal, sondern auch materiell ernst zu nehmen, wird versucht, mit staatlicher Förderung den Aufbau einer offenen und föderierten Infrastruktur von Daten- und Infrastrukturdiensten, insbesondere mit dem Projekt GAIA-X voranzutreiben. Da noch ca. zwei Drittel der existierenden Rechen-Infrastruktur in vielen mittelgroßen und kleinen Rechenzentren verteilt ist, die nicht den Hyperscalern angehören, besteht bei einigen Akteuren die Hoffnung, diese physischen Infrastrukturen für die neuen offenen und föderiert integrierbaren Daten- und Infrastrukturdienste nutzen zu können.

Allerdings müssten dafür diese physischen Infrastrukturen auch sicherheitstechnisch so beschaffen sein, dass eine souveräne Nutzung möglich ist. Viele der bestehenden mittelgroßen und kleinen Rechenzentren erfüllen aber nicht mehr die Sicherheits- und Datenschutzanforderungen unter Berücksichtigung des aktuellen Standes der Technik. Diese Anforderungen werden aber seitens der Nutzer als gesetzlich Verantwortliche und damit mittelbar auch durch die Anbieter von Software-Dienste gestellt.

Daher sind die Entwicklung und die Bereitstellung von Secure Platforms eine ebenso notwendige Voraussetzung für eine souveräne Nutzung von Recheninfrastruktur wie die u.a. durch GAIA-X vorangetriebenen entbündelten und offen verfügbaren Daten- und Infrastrukturdienste.

Häufig ist bei den Akteuren der Community ein großer - vielleicht zu großer - Respekt vor dieser Aufgabe und vor dem technischen Vorsprung der Hyperscaler zu beobachten. Dennoch sind Anstrengungen in diesem Bereich unvermeidlich. Durch die Umsetzung einer oder mehrerer Referenzimplementierungen könnte die Machbarkeit solcher Secure Platforms bestätigt und so deren kommerzielle Entwicklung und Bereitstellung beschleunigt werden. Eine Referenzimplementierung wäre der erste Schritt für ohnehin notwendige Anstrengungen. Kollaborative und zusätzlich öffentlich geförderte Ansätze sind hierfür sicherlich erfolgsversprechender als Einzelinitiativen und daher wünschenswert.

7.2 Die dringlichsten Schritte und Forderungen des Bundesverbandes IT-Sicherheit TeleTrusT an die Bundesregierung

Neben der Förderung von universitärer und industrieller Forschung sollten auch "erste industrielle Deployments" mit staatlichen Mitteln gefördert werden. Dies betrifft sowohl die Hard- und Softwarekomponenten für komplette Secure Platforms bzw. Netze aus Confidential Computing Modulen als auch die europäischen Fähigkeiten in der Mikroelektronik, insbesondere CPU-Design und Herstellung. Eine Vernetzung solcher Förderung, also von industrieller Forschung, Entwicklung und Innovation mit Aktivitäten der Entwicklungszusammenarbeit könnte die Beschaffungssituation durch die Verteilung der Herstellung auf andere globale Machtblöcke als die USA und China weiter verbessern. Dadurch könnten mit den technischen Empfehlungen aus diesem Papier die Infrastruktur für Anwendungen mit einem hohen Bedarf an Sicherheit und Souveränität geschaffen werden. Die besonders hohe Sicherheit und Beschaffenheit für eine souveräne Nutzung sollte für höchste Anerkennung in den Fachkreisen durch qualifizierte und unabhängige sicherheitstechnische Untersuchungen bestätigt werden.

8 Exkurs - Sicherer Endandwenderarbeitsplatz

Sicherer Arbeitsplatz? Gibt es das? Die Antwort heißt derzeit: "nur bedingt". Trotz allem ist es richtig, einen möglichst sicheren Arbeitsplatz einzurichten. "Möglichst sicher" meint hier ausdrücklich, die Risiken von Fehlern zu minimieren. Mögliche Fehlerquellen reichen von CPU-Fehlern wie z.B. "Spectre" oder auch "Melt-Down" bis hin zu Tippfehlern. Fehler sind niemals gänzlich auszuschließen; jedoch sollten die Folgen möglichst klein gehalten werden.

Wie geht man vor? Es macht Sinn, für jeden Arbeitsplatz ein IT-Arbeitsplatzprofil zu erstellen. Eine Gruppierung erleichtert die Aufgabe und erlaubt letztlich auch die individuelle Anpassung dort, wo es erforderlich wird. Es sollten bei der Erstellung die üblichen aus der Betriebswirtschaft bekannten Ablauf- und Kontrollstrukturen berücksichtigt werden ergänzt um IT spezifische Abläufe. Hierbei sollten einige, hier folgende Grundprinzipien beachtet werden:

- zero trust
- deny - allow
- allow - deny
- need-to-know
- Segmentierung und Separierung
- Kerckhoff's Prinzip[19]
-

Sie bedeuten im Einzelnen:

"zero trust", wörtlich übersetzt mit "kein Vertrauen", heißt keinesfalls allem und jedem zu misstrauen. Es meint eine Herangehensweise, ausgehend von "keinem Vertrauen" hin zu kontrolliertem Arbeiten. Es soll durch diese Herangehensweise sichergestellt werden, mögliche Fehlerursachen frühzeitig zu erkennen und Risiken zu minimieren.

"deny - allow" ist in der Unix-Welt seit langem bekannt. Verbiete zunächst allen den Zugang, dann erlaube einzelnen den Zugang/Zugriff. Einfaches Beispiel "Intranet": Kein Zugriff aus dem Internet möglich (deny), nur Zugriff aus internen Bereichen (allow).

"allow - deny" geht den anderen Weg. Beispiel "Intranet": Erlaube allen aus den internen Netzen den Zugriff, sperre jedoch den Zugang zu bestimmten Seiten.

"need-to-know" meint den Umgang mit Daten. Jeder Mitarbeiter benötigt einiges Wissen, um seine Aufgaben erledigen zu können. Zu diesem Wissen gehören auch vertrauliche Informationen.

"Segmentierung und Separierung" meint die Trennung von Arbeitsbereichen. Separierung trennt die verschiedenen Arbeitsfelder voneinander, Segmentierung betrifft die zugrunde liegende Netzwerkarchitektur.

Kerckhoff's Prinzip wird in der Kryptographie benutzt, trifft aber nicht nur dort zu. Auguste Kerckhoff hat den Satz entwickelt, nachdem ein Geheimnis nicht verraten werden kann, wenn zur Verschlüsselung eine sichere Methode angewendet wird. Zur sicheren Methode gehört ein geheimer Schlüssel. Die Methode darf bekannt werden, der Schlüssel jedoch nicht. Analog gilt z.B., dass die Besitzer von Zugangsschlüsseln nicht bekannt sein sollten. Es schadet hingegen nicht, wenn man weiß, dass ein Schloss nur dann geöffnet werden kann, wenn z.B. 3 von 5 Schlüsselwärtern ihre Schlüssel in der richtigen Reihenfolge benutzen.

Ein IT-Arbeitsplatzprofil sollte gewisse Anforderungen erfüllen. So u.a. eine Tätigkeitsbeschreibung für den Mitarbeiter (Gruppe). Sodann folgen Ausführungen zu den IT-spezifischen Themen. Für jeden Abschnitt sollten die Einzelheiten formuliert werden:

- Zugangs- und Zugriffskontrolle
- Identifikation und Authentisierung
- Kryptographische Unterstützung
- Informationsflusskontrolle
- Sicherheitsprotokollierung und Nachweisführung
- Interner Schutz der Benutzerdaten
- Selbstschutz der Sicherheitsfunktionen und ihrer Daten
- Sicherheitsmanagement

Beispiel "Identifikation und Authentisierung":

Der Mitarbeiter meldet sich mit Namen (Namenskürzel oder Emailadresse) und Passwort an.

Zusätzlich wird vom System das Vorhandensein eines weiteren Sicherheits-Merkmals (Token) geprüft, der sogenannten Zwei-Faktor Authentifizierung. Entweder in Form von Hardwaretoken, wie z.B. Smartcards, U2F-Tokens oder als Software-Token das von einer separaten Authenticator-Applikation z.B. auf dem Handy, aber grundsätzlich über einen separaten Übertragungskanal angezeigt oder zur Generierung angestoßen wird.

Bei erfolgreicher Eingabe der Kenndaten weist das System Mitarbeitern ihrer Rolle zu. In dieser Rolle sind die Zugriffsberechtigungen auf Speichermedien, Netzwerke etc. hinterlegt. Der Mitarbeiter startet die benötigten Programme selbst. Eine weitere Identifizierung ist nicht erforderlich (Single-Sign On).

Einen besonderen Punkt stellt die "Informationsflusskontrolle" dar. Hier sollte dargelegt werden, wie der Informationsaustausch zwischen internen Abteilungen sowie mit externen Stellen abläuft. Hier kommt das Prinzip "need-to-know" zum Tragen. Durch festgelegte Routinen zum Informationsaustausch wird ein versehentliches Offenlegen von vertraulichen Informationen deutlich eingeschränkt.

Die vorgenannten Anforderungen können durch weitere Anforderungen ergänzt werden.

Es stehen auf dem Markt reichlich technische Hilfsmittel zur Umsetzung der genannten Anforderungen zur Verfügung. Angefangen von gehärteten Betriebssystemen bis hin zu Systemen mit BSI Prüfsiegel zum Einsatz in Behörden obliegt es dem Betreiber, eine geeignete Lösung auszuwählen. Beispiele für Anforderungsprofile finden sich auf den Internetseiten des BSI.[20]

9 Referenzen

[1] Bundesverband IT-Sicherheit e.V. (TeleTrusT) (2020). Handreichung "Security by Design" - Leitfaden für Entscheider (https://www.teletrust.de/publikationen/broschueren/security-by-design/?tx_reintdownloadmanager_reintdlm%5Bdownloaduid%5D=10072&cHash=308282f64534cf4e565f85cb4fb86f46)

[2] Edsger W. Dijkstra: "The humble programmer" (https://dl.acm.org/doi/10.1145/1283920.1283927)

[3] Bruce Schneier: "Complexity the Worst Enemy of Security" (https://www.schneier.com/news/archives/2012/12/complexity_the_worst.html)

[4] Meltdown and Spectre - Vulnerabilities in modern computers leak passwords and sensitive data. (https://meltdownattack.com)

[5] Foreshadow - Breaking the Virtual Memory Abstraction with Transient Out-of-Order Execution (https://foreshadowattack.eu)

[6] ZombieLoad Attack - Return of the Leaking Dead (https://zombieloadattack.com)

[7] Side Channel Vulnerabilities: Microarchitectural Data Sampling and Transactional Asynchronous Abort (https://www.intel.com/content/www/us/en/architecture-and-technology/mds.html)

[8] L1D Eviction Sampling / CVE-2020-0549 (https://software.intel.com/security-software-guidance/advisory-guidance/l1d-eviction-sampling)

[9] Platypus - With Great Power Comes Great Leakage (https://platypusattack.com)

[10] Moritz Lipp, Vedad Hažić, Michael Schwarz, Arthur Perais, Clémentine Maurice, Daniel Gruss (2020). Take A Way: Exploring the Security Implications of AMD's Cache Way Predictors. ASIA CCS '20: Proceedings of the 15th ACM Asia Conference on Computer and Communications Security, October 2020, S. 813-825. (https://doi.org/10.1145/3320269.3384746) [11] Plug'nPwn - Connect to Jailbreak (https://blog.t8012.dev/plug-n-pwn/)

[12] Intel x86 Root of Trust: loss of trust (https://blog.ptsecurity.com/2020/03/intelx86-root-of-trust-loss-of-trust.html#more)

[13] Watch Your Containers: Doki Infecting Docker Servers in the Cloud (https://www.intezer.com/blog/cloud-security/watch-your-containers-doki-infecting-docker-servers-in-the-cloud/)

[14] The Mainframe Is Seeing a Resurgence. Is Security Keeping Pace? (https://www.darkreading.com/vulnerabilities---threats/the-mainframe-is-seeing-a-resurgence-is-security-keeping-pace/a/d-id/1335476)

[15] Jäger, Rieken, et.al. (2020). Manipulationssichere Cloud-Infrastrukturen. ISBN 978-3-658-31848-2. Springer Nature.

[16] L. Abdullah, F. Freiling, J. Quintero, and Z. Benenson. (2018). Sealed Computation: Abstract Requirements for Mechanisms to Support Trustworthy Cloud Computing. 2nd International Workshop on SECurity and Privacy Requirements Engineering- SECPRE 2018 in conjunction with ESORISCS2018.

[17] IEC 62443-4-2: "Security for industrial automation and control systems - Part 4-2: Technical security requirements for IACS components".

[18] Saflok Hotel Safe Override (2018) (https://www.youtube.com/watch?v=De0D7otNxME)

[19] Auguste Kerckhoff (1883). La cryptographie militaire. Journal des sciences militaires, Vol. IX, S. 5-38 (Jan. 1883), und S. 161-191 (Feb. 1883). (https://www.petitcolas.net/kerckhoffs/index.html)

[20] Bundesamt für Sicherheit in der Informationstechnik BSI (2021). VS-Anforderungsprofile (https://www.bsi.bund.de/DE/Themen/Oeffentliche-Verwaltung/Zulassung/VS-Anforderungsprofile/vs-anforderungsprofile_node.html)

Bundesverband IT-Sicherheit e.V. (TeleTrusT)

Der Bundesverband IT-Sicherheit e.V. (TeleTrusT) ist ein Kompetenznetzwerk, das in- und ausländische Mitglieder aus Industrie, Verwaltung, Beratung und Wissenschaft sowie thematisch verwandte Partnerorganisationen umfasst. Durch die breit gefächerte Mitgliederschaft und die Partnerorganisationen verkörpert TeleTrusT den größten Kompetenzverbund für IT-Sicherheit in Deutschland und Europa. TeleTrusT bietet Foren für Experten, organisiert Veranstaltungen bzw. Veranstaltungsbeteiligungen und äußert sich zu aktuellen Fragen der IT-Sicherheit. TeleTrusT ist Träger der "TeleTrusT European Bridge CA" (EBCA; PKI-Vertrauensverbund), der Expertenzertifikate "TeleTrusT Information Security Professional" (T.I.S.P.) und "TeleTrusT Professional for Secure Software Engineering" (T.P.S.S.E.) sowie des Vertrauenszeichens "IT Security made in Germany". TeleTrusT ist Mitglied des European Telecommunications Standards Institute (ETSI). Hauptsitz des Verbandes ist Berlin.

Kontakt:

Bundesverband IT-Sicherheit e.V. (TeleTrusT)
Dr. Holger Mühlbauer
Geschäftsführer
Chausseestraße 17
10115 Berlin
Telefon: +49 30 4005 4306
E-Mail: holger.muehlbauer@teletrust.de
https://www.teletrust.de

TeleTrusT
Pioneers in IT security.