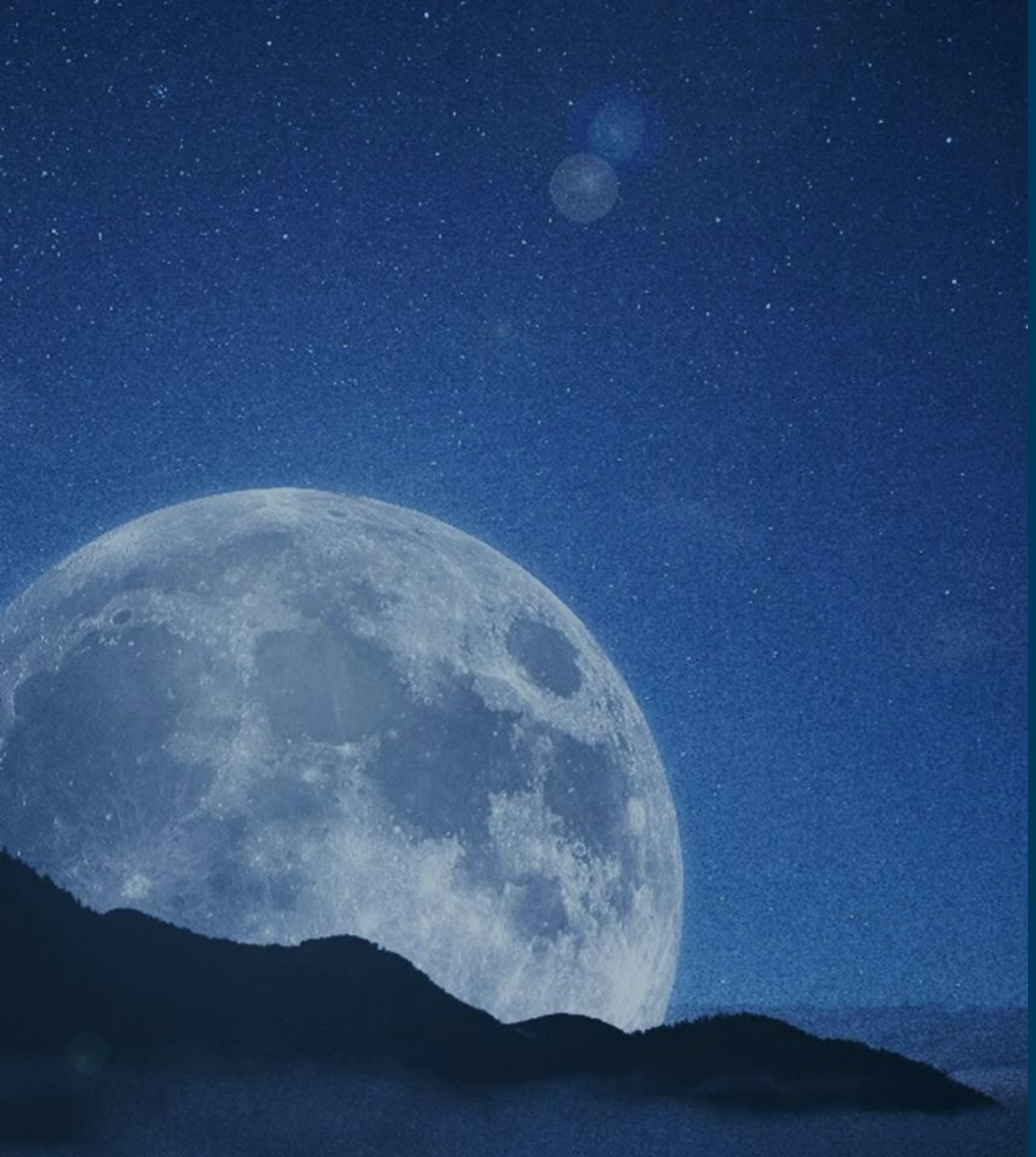
LUNARIS
Digital Solutions

DER CYBER RESILIENCE ACT ALS GRADMESSER MODERNER SOFTWAREENTWICKLUNG

Lars Roith



MIT WEM HABT IHR ES ZU TUN?

Lars Roith

Solution Consultant | CEO

Lars.Roith@lunaris.digital

https://lunaris.digital

LUNARIS
Digital Solutions

KEY TAKE AWAYS

Was ist der Cyber Resilience Act (CRA)?

Welche Anforderung stellt der CRA?

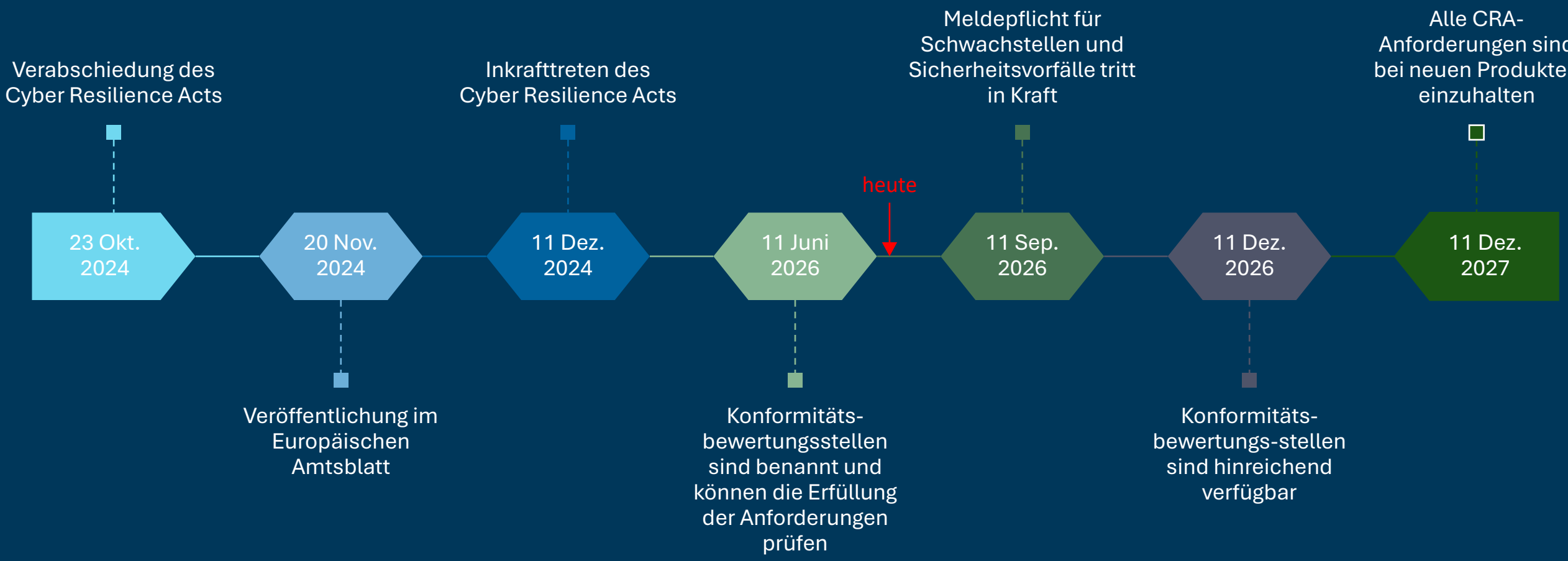
Wie adressieren moderne Entwicklungsprozesse diese Anforderungen?

Wie groß ist Eure Lücke?

WAS IST DER CYBER RESILIENCE ACT

Verordnung (EU) 2024/2847 des Europäischen Parlaments und des Rates vom 23. Oktober 2024 über **horizontale Cybersicherheitsanforderungen** für Produkte mit **digitalen Elementen**

DER ZEITPLAN DES CYBER RESILIENCE ACTS



LUNARIS
Digital Solutions

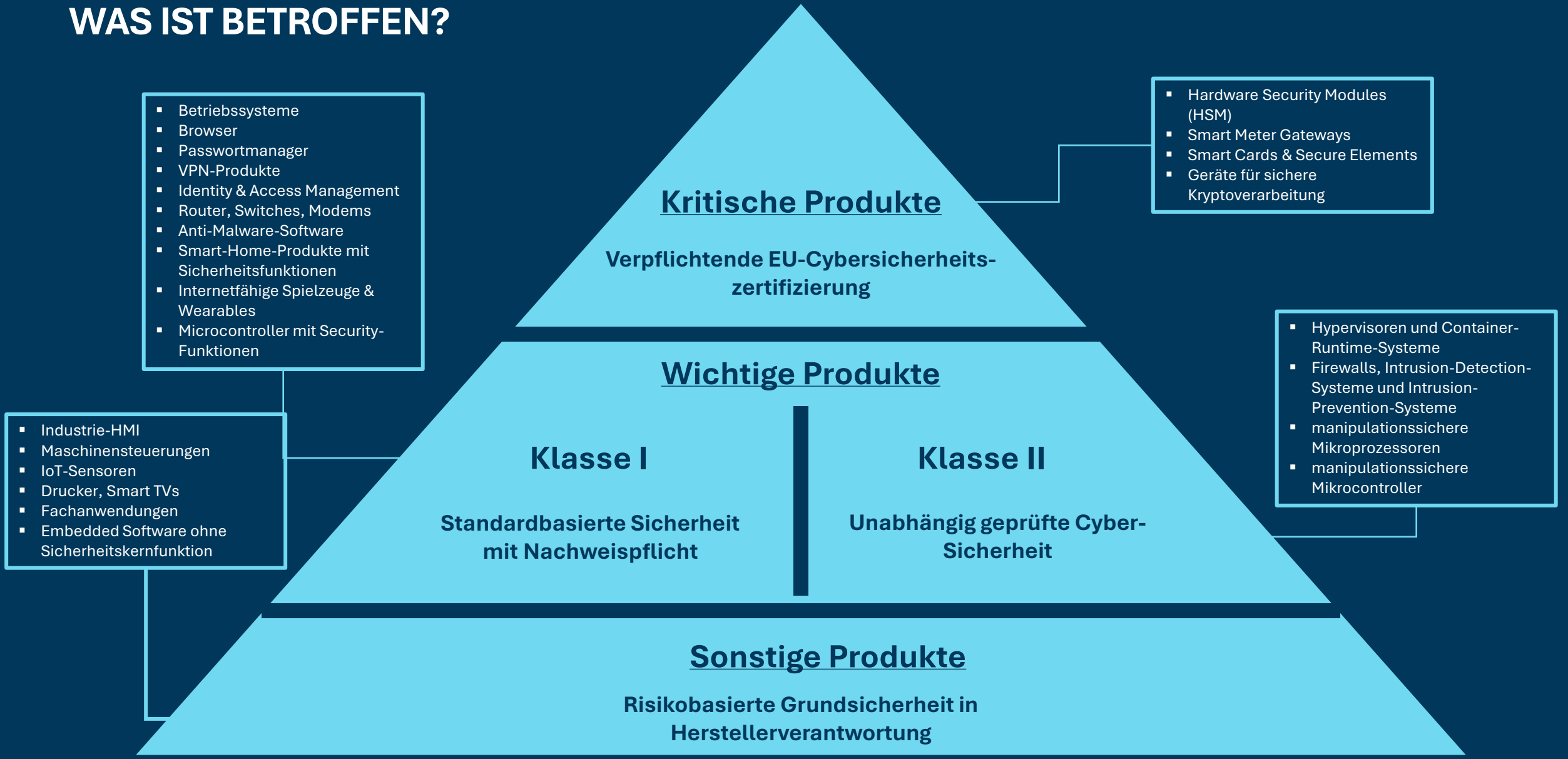
WER IST BETROFFEN?

Artikel 2 - Anwendungsbereich

Art. 2(1): Diese Verordnung gilt für **auf dem Markt bereitgestellte Produkte mit digitalen Elementen**, deren **bestimmungsgemäßer Zweck** oder **vernünftigerweise vorhersehbare Verwendung** eine **direkte oder indirekte logische oder physische Datenverbindung mit einem Gerät oder Netz** einschließt.

- Hersteller
- Importeure & Distributoren
- Systemintegratoren & OEMs

WAS IST BETROFFEN?



- Betriebssysteme
- Browser
- Passwortmanager
- VPN-Produkte
- Identity & Access Management
- Router, Switches, Modems
- Anti-Malware-Software
- Smart-Home-Produkte mit Sicherheitsfunktionen
- Internetfähige Spielzeuge & Wearables
- Microcontroller mit Security-Funktionen

- Industrie-HMI
- Maschinensteuerungen
- IoT-Sensoren
- Drucker, Smart TVs
- Fachanwendungen
- Embedded Software ohne Sicherheitskernfunktion

- Hardware Security Modules (HSM)
- Smart Meter Gateways
- Smart Cards & Secure Elements
- Geräte für sichere Kryptoverarbeitung

- Hypervisoren und Container-Runtime-Systeme
- Firewalls, Intrusion-Detection-Systeme und Intrusion-Prevention-Systeme
- manipulationssichere Mikroprozessoren
- manipulationssichere Mikrocontroller

LUNARIS
Digital Solutions

ANFORDERUNGEN IM ÜBERBLICK

Anhang I, Teil I

Produkte mit digitalen Elementen werden so **konzipiert, entwickelt und hergestellt**, dass sie **angesichts der Risiken** ein **angemessenes Cybersicherheitsniveau** gewährleisten.

- Cybersicherheit wird integraler Bestandteil von Design, Entwicklung und Produktion.
- Sicherheitsaufwand richtet sich nach realen, produktspezifischen Risiken
- Ziel ist ein nachweisbar angemessenes Sicherheitsniveau.

➢ *Der CRA fordert keine perfekte Sicherheit*
➢ *Der CRA duldet keine unbegründete Unsicherheit.*

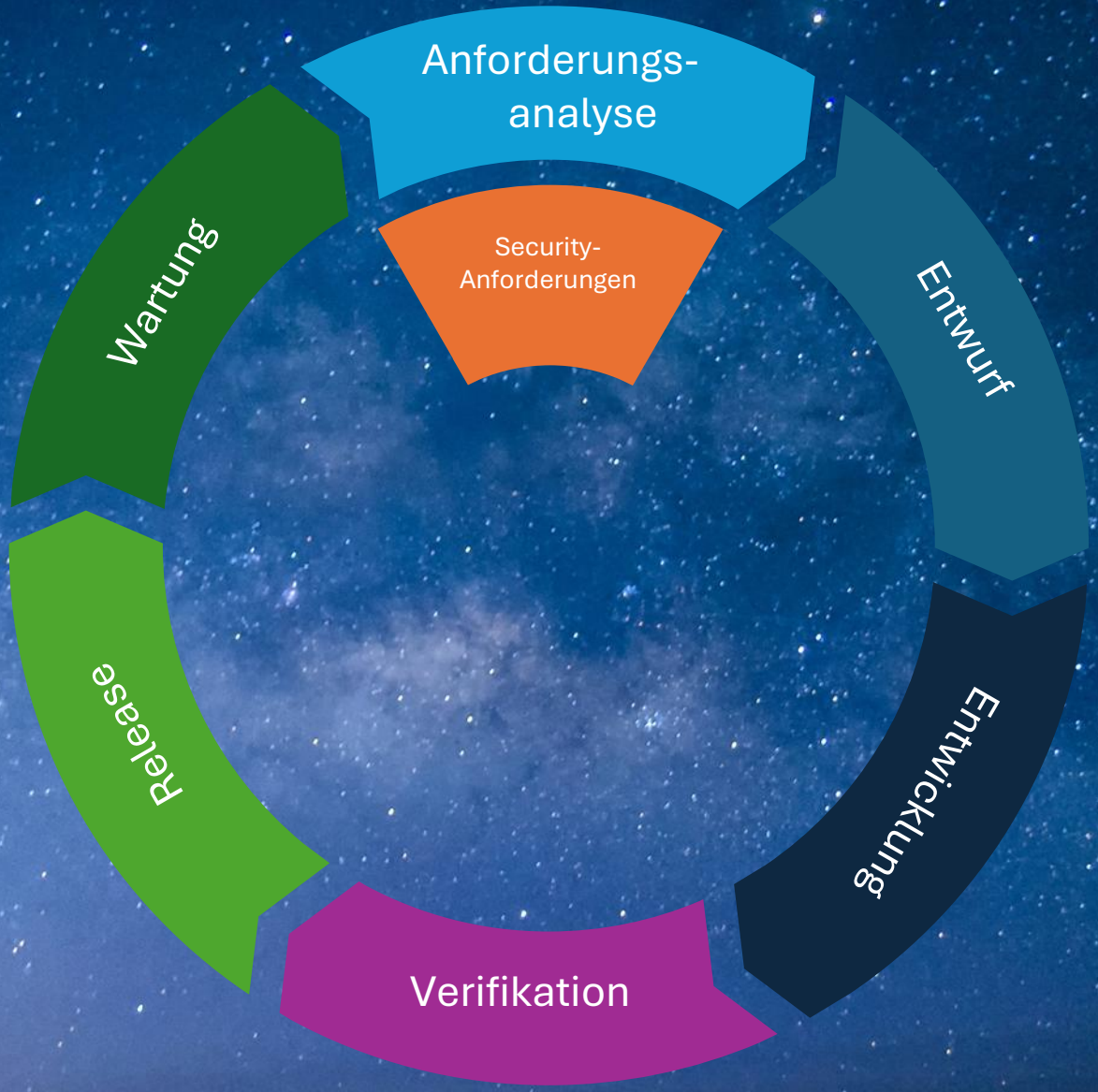
ANFORDERUNGEN IM ÜBERBLICK

Cybersecurityanforderungen an das Produkt

Anforderungen zur Schwachstellenbehandlung

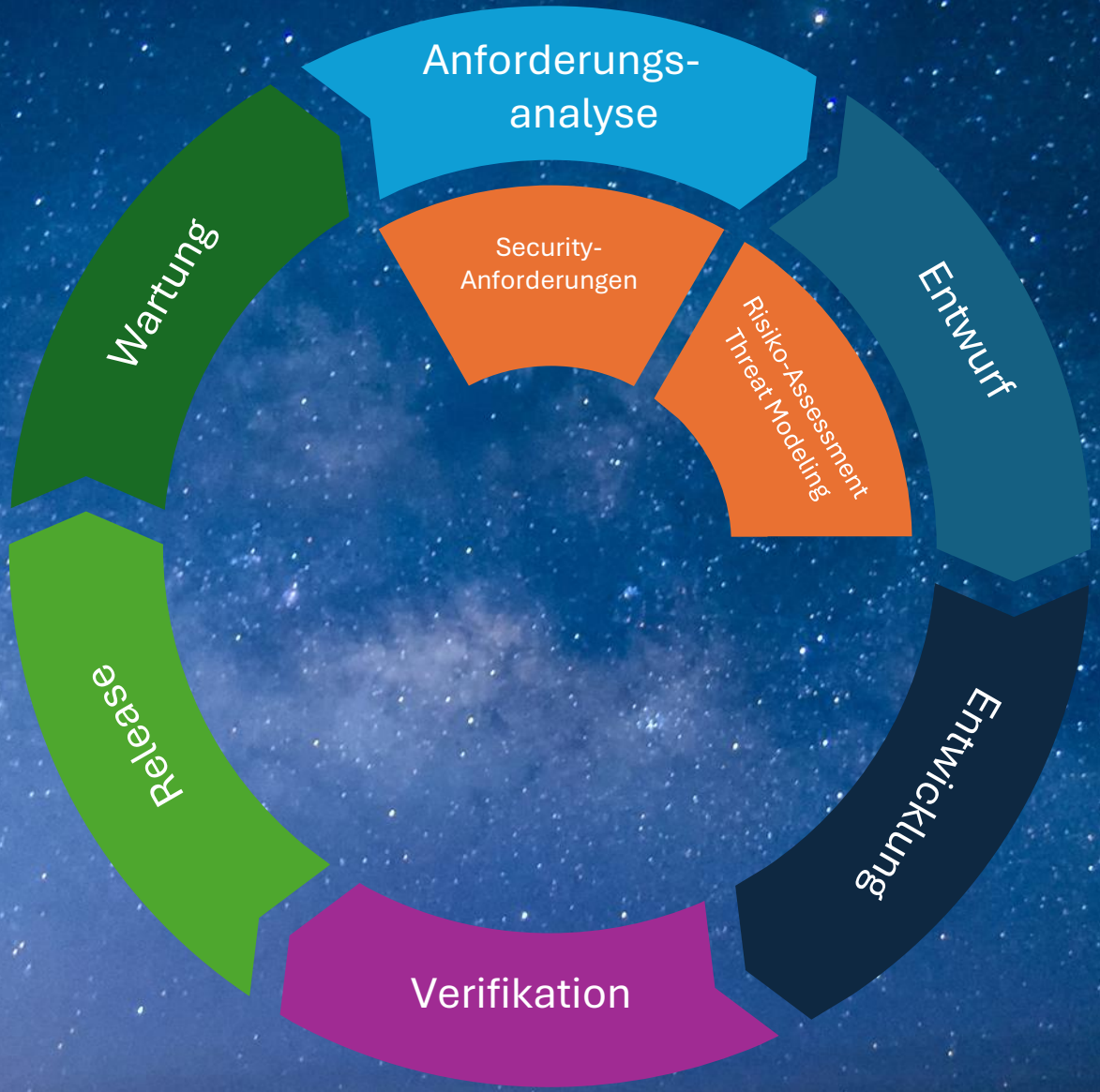
Anforderungen an Nutzerdokumentation

Anforderungen an technische Dokumentation



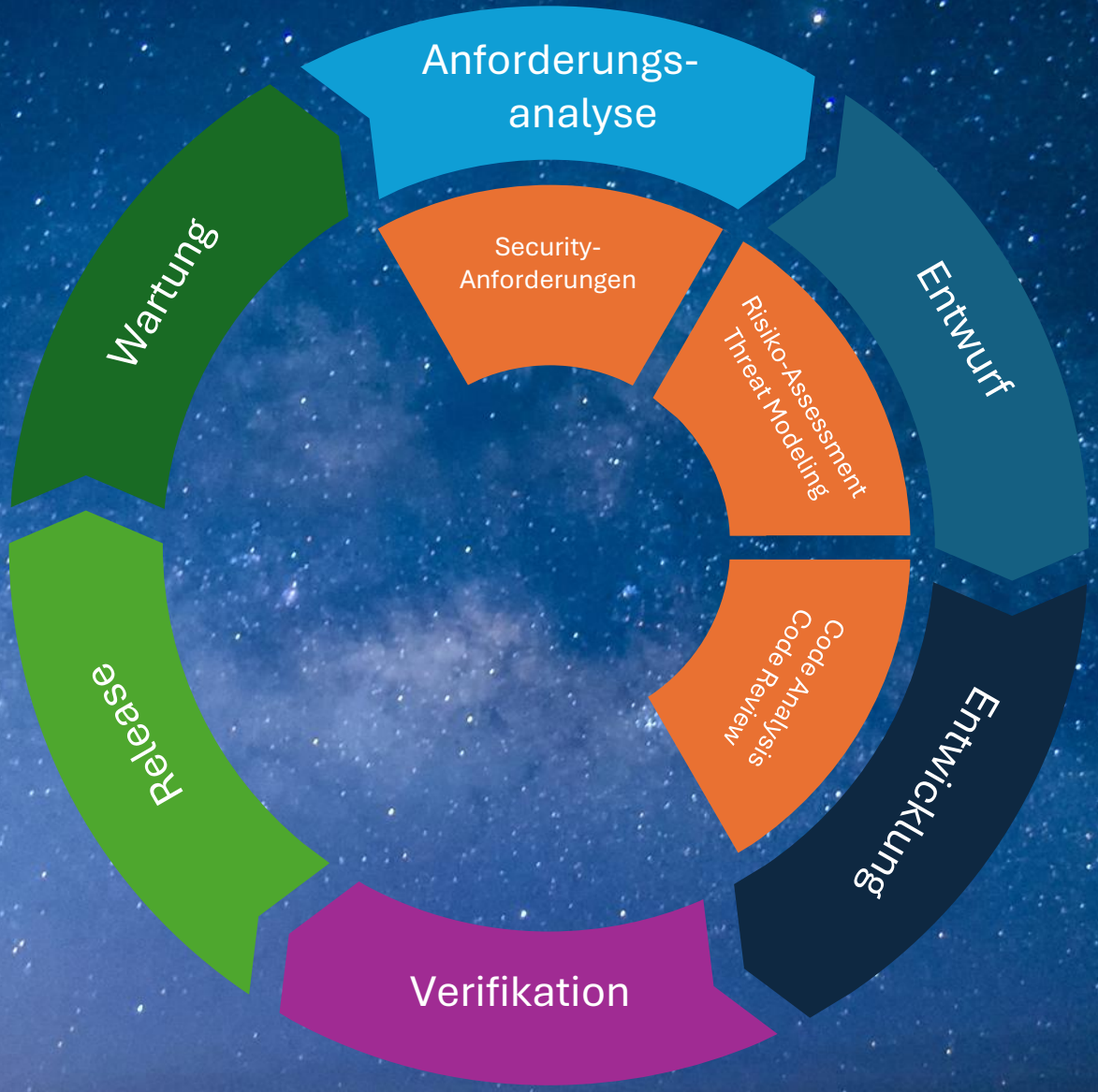
ANFORDERUNGSANALYSE

- Security-Anforderungen gehören früh in Backlog, DoR und Akzeptanzkriterien
- Der CRA macht etablierte NFR-Praktiken nachweispflichtig
- Secure-by-Default, Updatefähigkeit, Logging und Supportzeitraum müssen Anforderungen werden
- Späte Nachrüstung ist teuer und oft sogar architekturrelevant



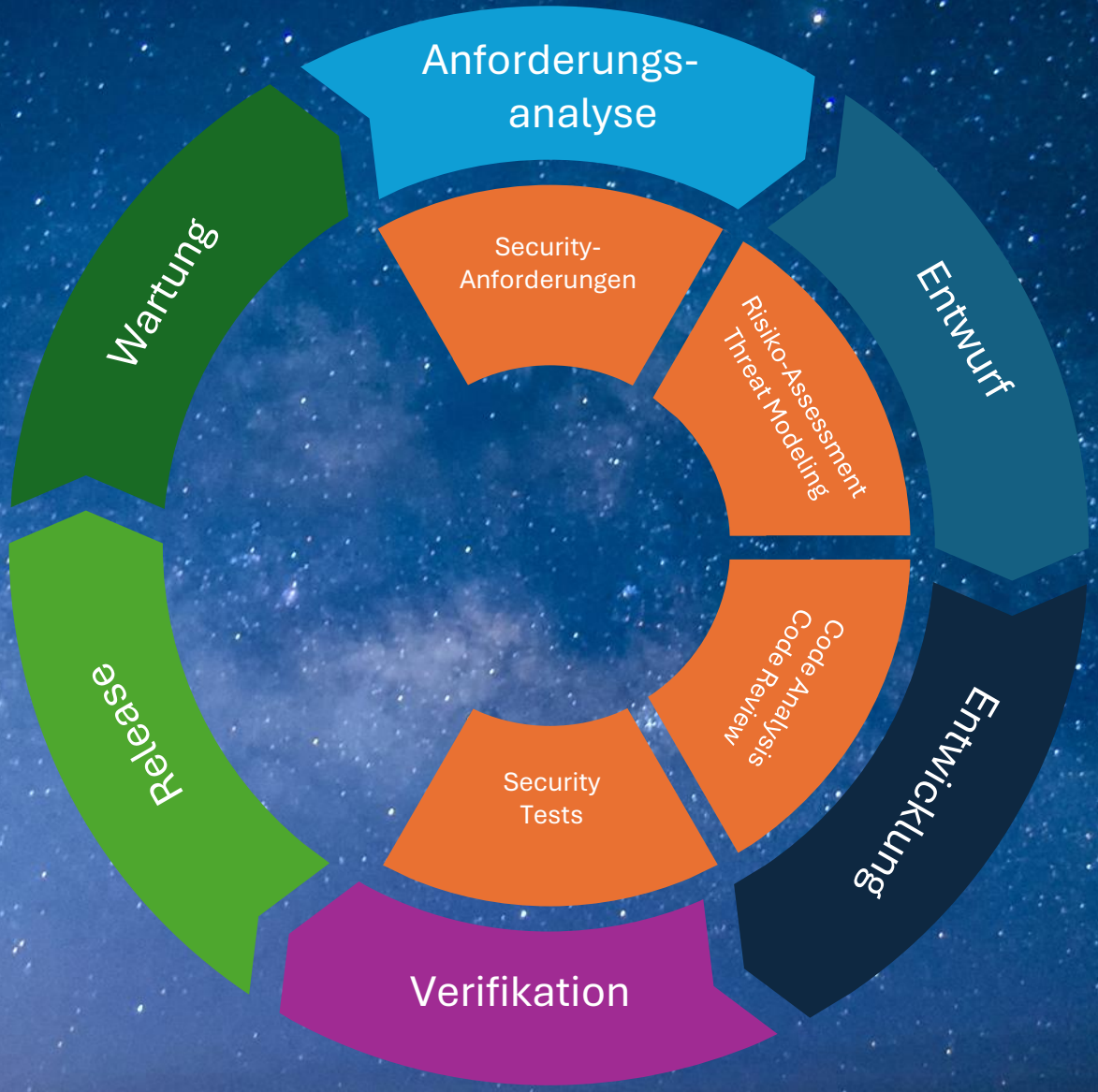
ENTWURF

- Threat Modeling macht Risiken architekturrelevant
- Etablierte Methoden: STRIDE, Attack Trees, TARA, PASTA
- Entscheidend sind konkrete Folgen für Schnittstellen, Rechte, Datenflüsse und Updatepfade
- Risiken müssen in Controls, Backlog-Items und Tests übersetzt werden



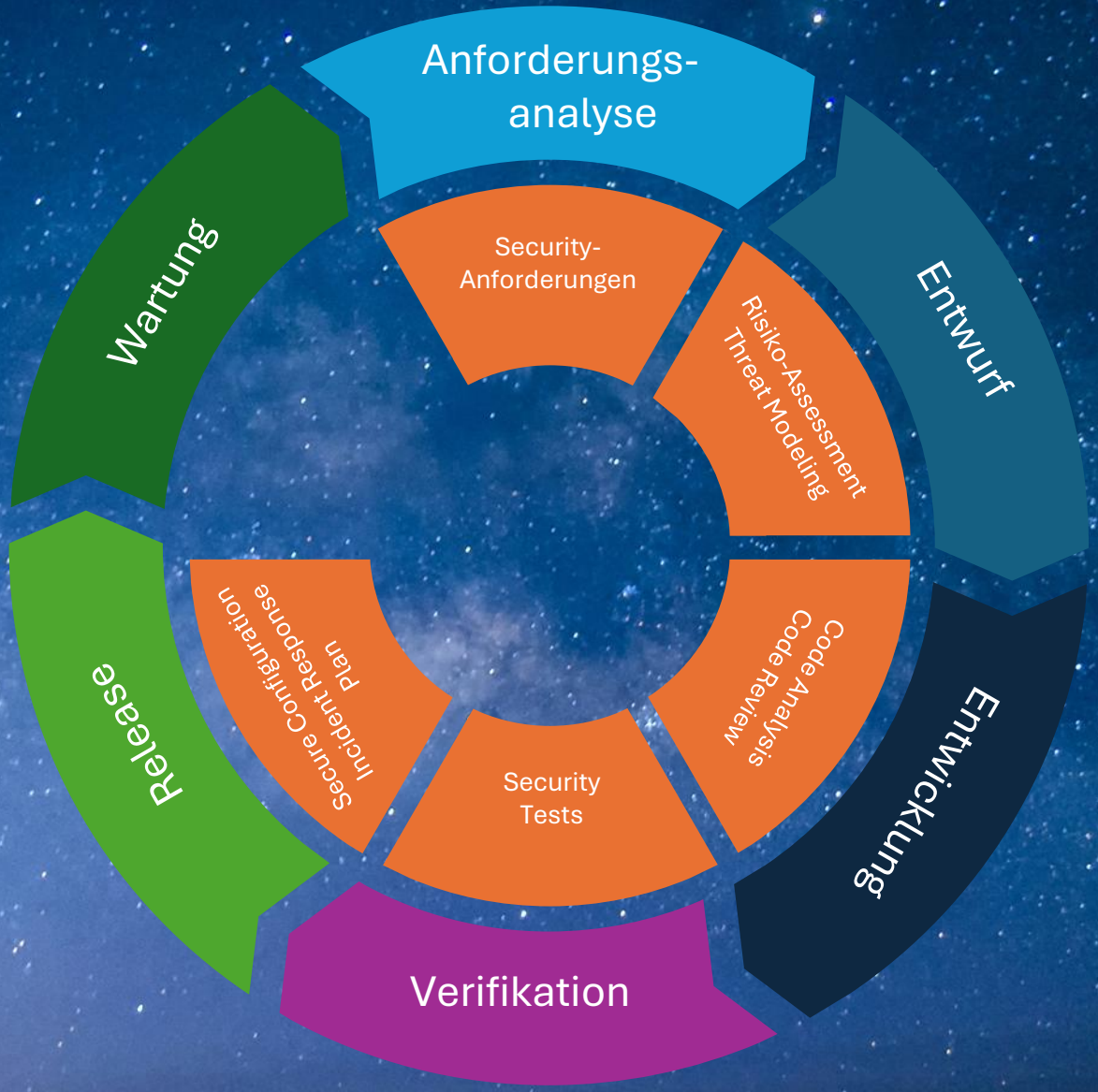
ENTWICKLUNG

- Security muss in IDE, Build, Pull Request und CI/CD sichtbar sein
- SAST, SCA, Secret Scanning, IaC-Checks und DAST sind Standardwerkzeuge
- Code Reviews brauchen Security-Fokus: Auth, Input Validation, Logging, Fehlerbehandlung
- Ohne automatisierte Checks bleibt Security abhängig von Aufmerksamkeit und Zufall



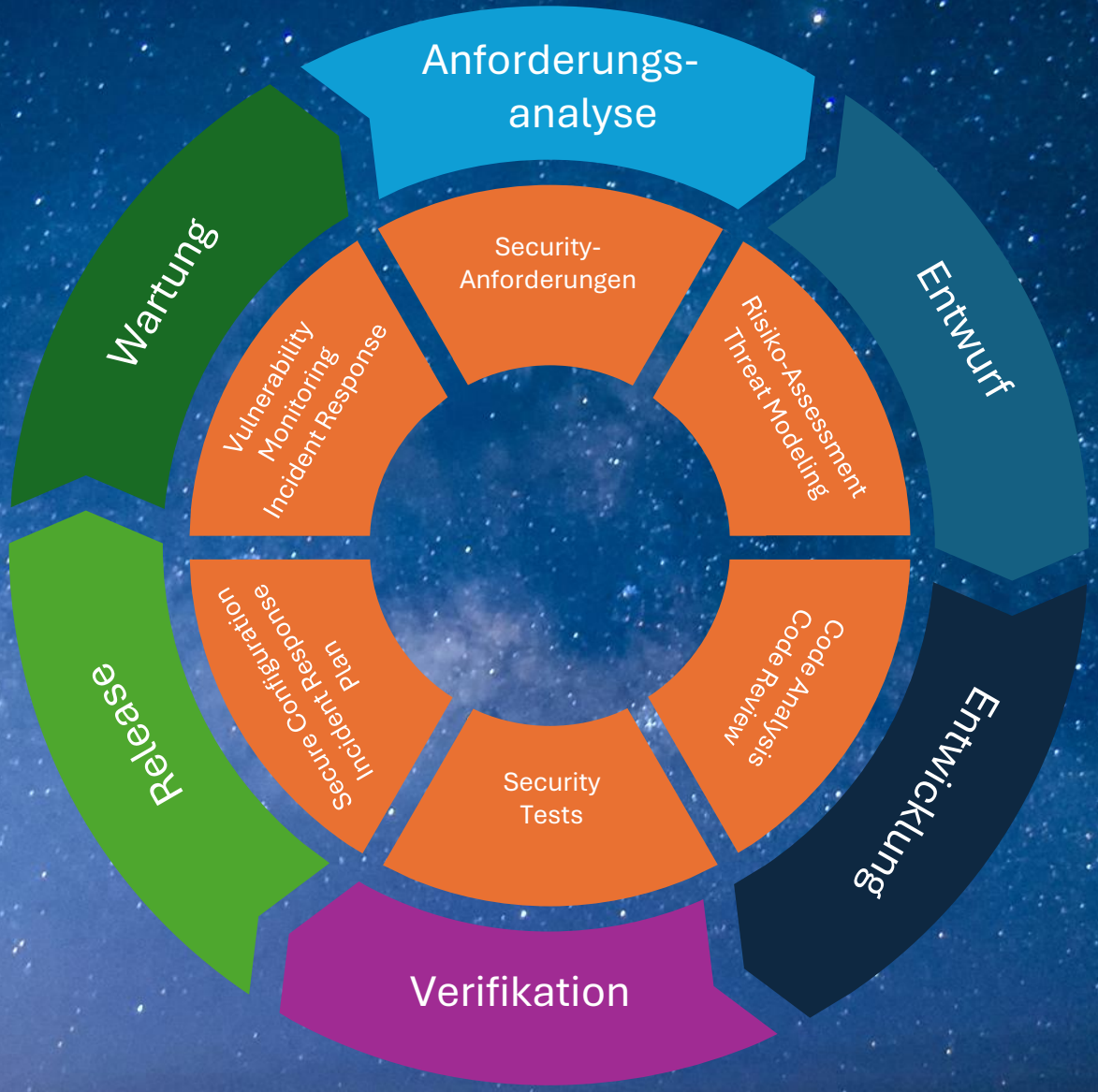
VERIFIKATION

- Security ergänzt die klassische Testpyramide: negative Tests, Fuzzing, Penetrationstests
- SBOM und SCA machen integrierte Komponenten und bekannte Schwachstellen sichtbar
- Tests müssen risikobasiert, wiederholbar und release-relevant sein
- Findings gehören zurück in Backlog, Risikoanalyse und Release-Gate



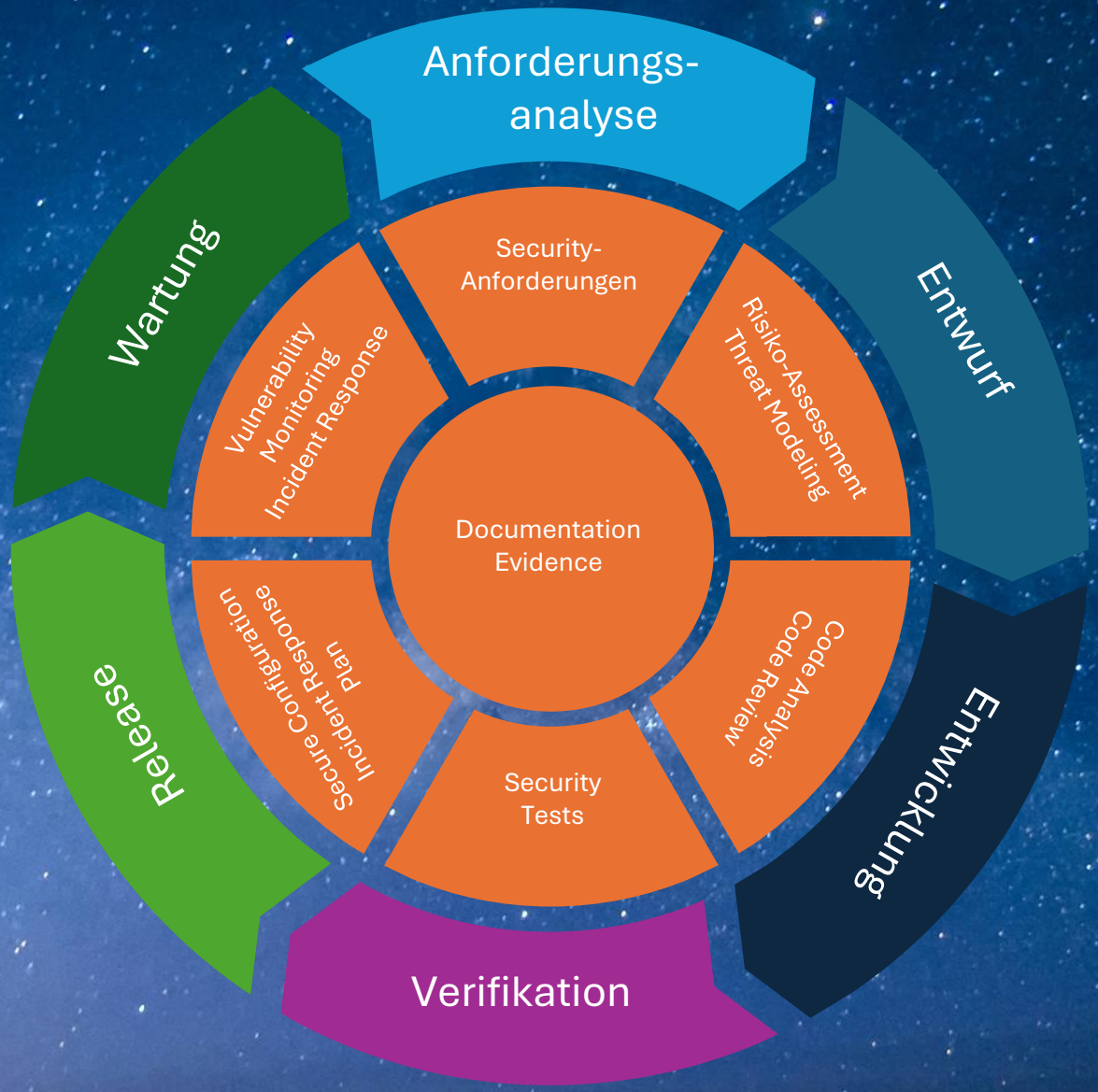
RELEASE

- Secure Configuration beinhaltet sichere Defaults, dokumentierte Abweichungen, reproduzierbare Hardening-Baseline
- Incident Response Plan: Verantwortlichkeiten, Meldewege, Patch-/Hotfix-Prozess und Kundenkommunikation stehen vor dem Release
- SBOM, bekannte Schwachstellen, Updatefähigkeit und Support-Ende werden Teil der Release-Freigabe
- Wer keinen Update- und Kommunikationskanal hat, kann Security-Vorfälle nicht beherrschbar behandeln



WARTUNG

- Produkte müssen über den Supportzeitraum aktiv überwacht werden
- SBOM wird zum Betriebsartefakt: Welche Version, welche Komponente, welcher Kunde ist betroffen?
- Vulnerability Monitoring braucht Triage: bekannt ≠ betroffen ≠ ausnutzbar ≠ patchpflichtig
- Incident Response braucht vorbereitete Abläufe: Bewertung, Eskalation, Meldung, Advisory, Hotfix
- Updatefähigkeit entscheidet über Reife: ohne Patchkanal wird jede Schwachstelle zum Krisenprojekt



DOKUMENTATION UND BEWEISFÜHRUNG

- CRA-Compliance braucht produktbezogene Nachweise über Vorgehen, Entscheidungen und Ergebnisse
- Technische Dokumentation entsteht über den gesamten Lifecycle
- Evidence umfasst Threat Models, SBOMs, Testberichte, Release-Freigaben, Update- und Incident-Prozesse
- Der Audit folgt der Beweiskette: Risiko erkannt, Maßnahme abgeleitet, umgesetzt, geprüft, freigegeben
- Interne Artefakte bleiben intern, gehören aber zur technischen Dokumentation und müssen vorlegbar sein

BEISPIEL 1: SCA IM CI REICHT NICHT

Schwachstellenmanagement endet nicht beim Scan

Typische Realität

Die Pipeline meldet CVEs, aber niemand besitzt die Entscheidung.

CRA-Druck

Bekannte ausnutzbare Schwachstellen müssen bewertet, priorisiert und behoben werden.

Moderne Umsetzung

SCA, Dependency-Checks, CVE-Monitoring, Exploit-Signale, Risiko-Triage und Fix-SLAs sind verbunden.

Evidence & Metrik

Wer hat wann entschieden, warum, mit welchem Fix und in welcher Version? MTTR, Triage-Time, offene kritische Findings pro Produkt.

BEISPIEL 2: SBOM IST KEIN SELBSTZWECK

Komponententransparenz muss zu Steuerungsfähigkeit führen

Typische Realität

Eine SBOM wird erzeugt, aber nicht genutzt.

CRA-Druck

Komponenten und Schwachstellen müssen über den Lebenszyklus beherrschbar sein.

Moderne Umsetzung

SBOM pro Release, Dependency Ownership, Policy-Gates und Herkunftsnachweise.

Evidence & Metrik

Build-Provenance, Attestations, signierte Artefakte, Trusted Repositories und gehärtete Runner; Komponente pro Produkt, Version und Kundenstand.

BEISPIEL 3: SECURITY-RELEASES DÜRFEN NICHT AUF DEN FEATURE-ZUG WARTEN

Updatefähigkeit ist eine Architektur- und Pipeline-Frage

Typische Realität

Kritische Fixes warten auf das nächste geplante Feature-Release.

CRA-Druck

Sicherheitsupdates müssen zeitnah, sicher und nachvollziehbar bereitgestellt werden.

Moderne Umsetzung

Getrennte Security-Release-Pfade, automatisierte Tests, Rollback-Strategie, signierte Artefakte.

Evidence & Metrik

Betroffene Versionen, Schweregrad, Impact, Mitigation, Upgrade-Pfad, Advisory, SBOM, Signatur und Rollout-Status.

BEISPIEL 4: SECURE BY DEFAULT STATT KUNDEN-HÄRTUNG

Unsichere Defaults sind keine Dokumentationsfrage

Typische Realität

Das Produkt ist sicher, wenn Kunden die Hardening-Anleitung lesen.

CRA-Druck

Sicherheit muss in der Standardkonfiguration angelegt sein.

Moderne Umsetzung

Default-Deny, Least Privilege, keine Standard-Credentials, sichere Recovery-Pfade und Security Header.

Evidence & Metrik

Default-Konfigurationen gegen Hardening-Baselines und Misconfiguration-Checks prüfen; Ausnahmen pro Version begründen.

BEISPIEL 5: ARCHITEKTUR ENTSCHEIDET ÜBER CRA-FÄHIGKEIT

Updatefähigkeit und Blast Radius werden im Entwurf festgelegt

Typische Realität

Ein Security-Fix zieht komplettes System, vollständigen Regressionstest und großen Kundeneingriff nach sich.

CRA-Druck

Sicherheitsmaßnahmen müssen zeitnah, zielgerichtet und mit beherrschbarem Risiko ausgeliefert werden.

Moderne Umsetzung

Klare Komponenten- und Trust-Grenzen, minimale Schnittstellen, entkoppelte Updatepfade, Rollback und Kompatibilitätsregeln.

Evidence & Metrik

Welche Architekturentscheidung reduziert welches Risiko? Betroffene Komponenten, Updateumfang, Testumfang, Rollback-Fähigkeit.

BEISPIEL 6: EOL IST KEIN KALENDERTERMIN

Supportzeiträume müssen technisch und organisatorisch tragfähig sein

Typische Realität

Produktversionen bleiben im Feld, obwohl das Team längst weitergezogen ist.

CRA-Druck

Schwachstellenhandling gilt über den definierten Supportzeitraum.

Moderne Umsetzung

Klare LTS-/EOL-Policy, unterstützte Versionen, Upgrade-Pfade und Kundenkommunikation.

Evidence & Metrik

Supportdauer ist Teil der Produktentscheidung; Supportende, Begründung, Nutzerinformation, letzte Security-Version und Migrationspfad.

DER SELBSTTEST

CRA SDLC Assessment


LUNARIS
Digital Solutions