Praxisleitfaden – Autonome KI-Agenten

Governance, Architektur & Regulierung am Beispiel OpenClaw

Inhalt

1. Executive Summary

KI-Agenten sind keine Chatbots. Sie sind digitale Handlungseinheiten.

OpenClaw steht exemplarisch für eine neue Systemklasse: KI-Agenten, die nicht nur Inhalte analysieren, sondern – abhängig von ihrer Konfiguration – eigenständig Entscheidungen treffen und operative Aktionen in Unternehmenssystemen auslösen können.

Der entscheidende Unterschied zu ChatGPT: ChatGPT antwortet. Ein KI-Agent kann handeln.

Damit verschiebt sich das Risikoprofil fundamental.

Was strategisch relevant ist

KI-Agenten verbinden:

- Informationsverarbeitung
- Entscheidungslogik
- Systemzugriff

Sobald ein System Zugriff auf interne Ressourcen erhält, entstehen neue Risikodimensionen:

- Manipulation der Entscheidungslogik
- Überprivilegierte Systemrechte
- Unkontrollierte Datenaggregation
- Automatisierte Fehlhandlungen
- Haftungs- und Reputationsrisiken

Das Risiko liegt nicht im Modell – sondern in der Integration.

Die Management-Kernfrage:

Nicht: „Ist das technisch leistungsfähig?“ Sondern: „Ist die Autonomie strukturell kontrolliert?“

Verantwortung und Regulierung

Wer Open-Source-KI-Agenten produktiv einsetzt, trägt die vollständige Verantwortung für Architektur, Zugriffskontrolle, Schwachstellenmanagement und regulatorische Einordnung (z. B. NIS2, AI Act, ISO 27001).

Sicherheit und Governance werden nicht mitgeliefert – sie müssen aktiv gestaltet, dokumentiert und dauerhaft gesteuert werden.

Fazit

KI-Agenten können Produktivität signifikant steigern. Aber:

- Autonomie braucht Architektur. Innovation braucht Governance.
- Unternehmen, die KI-Agenten kontrolliert skalieren, schaffen strategischen Vorsprung.
- Unternehmen, die sie unstrukturiert integrieren, erzeugen systemisches Risiko.

2. Einordnung: Was KI-Agenten von klassischer Software unterscheidet

Der Einsatz von KI-Agenten wie OpenClaw stellt keinen gewöhnlichen Softwareeinsatz dar. Während klassische Anwendungen deterministisch programmiert sind und klar definierte Funktionen ausführen, agieren KI-Agenten auf Grundlage probabilistischer Modelle, dynamischer Kontextverarbeitung und autonomer Entscheidungslogik.

Diese Unterschiede haben unmittelbare Auswirkungen auf Architektur, Governance und Sicherheitsbewertung.

Autonomie und Handlungsfähigkeit

Ein wesentliches Unterscheidungsmerkmal von KI-Agenten gegenüber klassischer Anwendungssoftware ist ihre operative Autonomie. Abhängig von Architektur, Konfiguration und Berechtigungsmodell sind sie in der Lage, eigenständig Aktionen auszuführen, ohne dass jeder einzelne Schritt explizit vordefiniert ist.

Zu den typischen Handlungsformen gehören insbesondere:

- Aufrufe externer APIs
- Verarbeitung, Transformation und Weitergabe von Daten
- Ausführung von Befehlen oder automatisierten Workflows
- Interaktion mit angebundenen Systemkomponenten und Fachanwendungen

Die Auswahl und Ausführung dieser Aktionen erfolgt nicht ausschließlich auf Grundlage deterministischer Entscheidungslogik. Vielmehr interpretiert ein zugrunde liegendes Sprachmodell die vorliegenden Inhalte, bewertet mögliche Handlungsoptionen probabilistisch und leitet daraus einen nächsten operativen Schritt ab.

Damit entsteht eine neue Form technischer Dynamik: Das System reagiert nicht nur regelbasiert auf klar definierte Eingaben, sondern kontextsensitiv auf semantisch interpretierte Informationen. Die konkrete Systemreaktion ist daher das Ergebnis einer modellbasierten Entscheidungsfindung innerhalb eines definierten Handlungsrahmens.

Aus Sicherheits- und Governance-Perspektive bedeutet dies, dass nicht nur implementierte Funktionen, sondern auch Entscheidungsräume, Berechtigungen und Integrationen Gegenstand der Risikobetrachtung sein müssen. Die operative Autonomie eines KI-Agenten ist funktional gewünscht, erfordert jedoch eine entsprechend kontrollierte und nachvollziehbare Systemarchitektur.

Kontextverarbeitung statt statischer Eingaben

Klassische Software verarbeitet in der Regel strukturierte Eingaben auf Basis klar definierter Parameter und formaler Schnittstellen. Eingabewerte unterliegen festen Schemata, Validierungsregeln und eindeutigen Typdefinitionen. Die Verarbeitung erfolgt innerhalb eines vorab spezifizierten, deterministischen Rahmens.

KI-Agenten hingegen analysieren überwiegend unstrukturierte Inhalte, etwa E-Mails, Dokumente, Webinhalte oder Benutzeranweisungen in natürlicher Sprache. Diese Informationen werden nicht lediglich syntaktisch ausgewertet, sondern semantisch interpretiert. Das zugrunde liegende Sprachmodell extrahiert Bedeutung, Kontext und potenzielle Handlungsoptionen und integriert diese in den weiteren Entscheidungsprozess.

Aus sicherheitstechnischer Sicht entsteht hieraus eine signifikant erweiterte Angriffsfläche. Unstrukturierte Inhalte können – bewusst oder unbeabsichtigt – manipulative, irreführende oder regelumgehende Anweisungen enthalten. Da das System auf Kontextinterpretation ausgelegt ist, besteht das Risiko, dass solche Inhalte in die Handlungslogik des Agenten einfließen.

Die Fähigkeit zur Kontextverarbeitung ist funktional gewünscht und stellt einen wesentlichen Mehrwert von KI-Agenten dar. Gleichzeitig erfordert sie zusätzliche architektonische und organisatorische Schutzmechanismen, um zwischen legitimen Informationen und potenziell schädlichen Handlungsimpulsen zu unterscheiden. Entsprechende Kontrollmechanismen sind daher integraler Bestandteil einer sicheren Systemintegration.

Dynamische Tool-Integration

KI-Agenten verfügen typischerweise über die Fähigkeit, externe Funktionen („Tools“) oder Integrationen einzubinden und zur Aufgabenerfüllung zu nutzen. Dabei kann es sich beispielsweise um Datenbankschnittstellen, Dateisystemzugriffe, Cloud-Dienste oder interne Fachanwendungen handeln. Der Agent greift somit nicht nur auf eigene Verarbeitungslogik zurück, sondern interagiert aktiv mit operativen Systemressourcen.

Diese dynamische Tool-Integration führt zu einer erheblichen Erweiterung der Systemgrenze. Der sicherheitsrelevante Betrachtungsrahmen beschränkt sich nicht auf den Agenten oder das zugrunde liegende Sprachmodell, sondern umfasst sämtliche angebundenen Komponenten, Schnittstellen und Datenquellen.

Aus sicherheitstechnischer Perspektive entsteht dadurch eine verteilte Vertrauensarchitektur. Jede Integration erweitert die potenzielle Angriffsfläche und beeinflusst das Gesamtrisiko. Maßgeblich sind insbesondere das Berechtigungsmodell der angebundenen Systeme, die Absicherung der Schnittstellen sowie die Kontrolle der durch den Agenten initiierten Aktionen.

Eine isolierte Bewertung des Agentencodes ist daher nicht ausreichend. Sicherheitsrelevant ist vielmehr das Zusammenspiel von Modelllogik, Tool-Konfiguration und den operativen Rechten innerhalb der angebundenen Systemlandschaft.

Nicht-deterministisches Verhalten

Im Gegensatz zu klassischer Software, die bei identischer Eingabe reproduzierbar dasselbe Ergebnis erzeugt, können Sprachmodelle unterschiedliche Ausgaben generieren. Die Entscheidungsfindung erfolgt auf Grundlage probabilistischer Modelllogik, bei der Wahrscheinlichkeitsverteilungen und Kontextgewichtungen eine zentrale Rolle spielen.

Dieses nicht-deterministische Verhalten hat unmittelbare Auswirkungen auf Qualitätssicherung und Sicherheitsbewertung. Klassische Testverfahren, die auf klar

definierten Eingabe-Ausgabe-Erwartungen beruhen, stoßen an Grenzen. Ebenso ist eine formale Verifikation im traditionellen Sinne nur eingeschränkt möglich. Auch die vollständige Vorhersagbarkeit des Systemverhaltens kann nicht gewährleistet werden, selbst wenn Rahmenbedingungen und Konfiguration unverändert bleiben.

Für die Sicherheitsarchitektur bedeutet dies eine methodische Verschiebung: Die Absicherung darf sich nicht ausschließlich auf funktionale Tests stützen. Stattdessen sind strukturierte Bedrohungsmodellierung, klare architektonische Kontrollmechanismen, restriktive Berechtigungsmodelle sowie belastbare Governance- und Monitoring-Strukturen erforderlich. Ziel ist es, Entscheidungsräume technisch zu begrenzen und Risiken systematisch zu kontrollieren, auch wenn das konkrete Einzelverhalten des Modells nicht deterministisch ist.

Erweiterte Systemverantwortung

Beim Einsatz von Open-Source-Agenten wie OpenClaw liegt die operative und sicherheitstechnische Gesamtverantwortung vollständig beim Betreiber. Anders als bei vollständig gemanagten SaaS-Diensten existiert kein externer Anbieter, der zentrale Sicherheitsmechanismen, Konfigurationsstandards oder regulatorische Einordnungen verbindlich vorgibt oder überwacht.

Das einsetzende Unternehmen ist daher selbst verantwortlich für die sichere Konfiguration der Anwendung, die kontrollierte Anbindung externer und interner Integrationen, die Definition und Durchsetzung eines restriktiven Berechtigungsmodells sowie die Implementierung von Logging-, Monitoring- und Incident-Response-Strukturen. Ebenso obliegt dem Betreiber die regulatorische Bewertung im jeweiligen Anwendungskontext, einschließlich der Einordnung in bestehende Compliance- und Risikomanagementprozesse.

Der Einsatz eines KI-Agenten ist vor diesem Hintergrund nicht als isolierte Einführung eines Tools zu verstehen. Vielmehr handelt es sich um eine sicherheitskritische Systemintegration, die tief in bestehende IT- und Datenstrukturen eingreifen kann. Entsprechend sind Architekturentscheidungen, Betriebsmodelle und Governance-Strukturen mit derselben Sorgfalt zu behandeln wie bei anderen geschäftskritischen Systemkomponenten.

Konsequenz für Unternehmen

Der Einsatz von KI-Agenten führt nicht lediglich zu einer Optimierung bestehender Prozesse, sondern verändert das Risikoprofil der gesamten IT-Landschaft. Diese Systeme verbinden Sprachmodelle mit Automatisierungslogik, operativen Systemzugriffen, externen Informationsquellen und unternehmensinternen Datenbeständen. Dadurch entstehen neue Abhängigkeiten, neue Entscheidungsräume und neue Angriffspunkte.

KI-Agenten wirken als Integrations- und Orchestrierungsschicht zwischen bislang getrennten Systemdomänen. Sie verarbeiten Informationen, leiten daraus Handlungen ab und greifen – abhängig von ihrer Konfiguration – aktiv in operative Abläufe ein. Die damit verbundene Dynamik kann Effizienzgewinne ermöglichen, erhöht jedoch zugleich die Komplexität der Sicherheitsarchitektur.

Die Bewertung und Absicherung solcher Systeme erfordert daher ein interdisziplinäres Vorgehen. Notwendig sind ein belastbares Architekturverständnis, eine strukturierte

Bedrohungsmodellierung, eine klare regulatorische Einordnung sowie etablierte Governance-Strukturen. Sicherheitsmaßnahmen müssen nicht nur technische Schwachstellen adressieren, sondern auch Entscheidungslogik, Berechtigungsmodelle und Integrationsketten berücksichtigen.

Vor diesem Hintergrund dient OpenClaw in diesem Leitfaden als praxisnahes Referenzsystem für eine gesamte Systemklasse: autonome KI-Agenten in Unternehmensumgebungen. Die dargestellten Prinzipien sind nicht auf eine spezifische Implementierung beschränkt, sondern auf vergleichbare agentenbasierte Architekturen übertragbar.

3. Technisches Funktionsprinzip von OpenClaw

OpenClaw steht exemplarisch für eine Klasse moderner KI-Agenten, die Large Language Models (LLMs) mit Automatisierungs- und Integrationsfunktionen kombinieren. Das System folgt keinem statischen Ablaufplan, sondern arbeitet in einem iterativen Entscheidungs- und Ausführungszyklus. Für die Sicherheitsbewertung ist es erforderlich, dieses Funktionsprinzip strukturell zu verstehen.

Der Agenten-Loop

Im Kern arbeitet ein KI-Agent in einem wiederkehrenden Entscheidungs- und Ausführungszyklus, der als „Agenten-Loop“ bezeichnet wird. Dieser Zyklus beginnt mit der Aufnahme einer Eingabe, beispielsweise in Form einer Nutzeranweisung, eines Dokuments oder externer Webinhalte. Anschließend erfolgt die semantische Analyse des verfügbaren Kontexts durch das zugrunde liegende Sprachmodell.

Auf Basis dieser Analyse leitet das System eine mögliche nächste Handlung ab. Diese kann in einer textlichen Antwort bestehen oder – sofern entsprechende Integrationen konfiguriert sind – im Aufruf eines Tools oder einer externen Schnittstelle. Das Ergebnis einer solchen Aktion wird wiederum verarbeitet, in den Kontextspeicher integriert und bildet die Grundlage für die nächste Entscheidungsiteration.

Der Prozess wiederholt sich so lange, bis entweder eine definierte Abbruchbedingung erreicht ist oder ein vorab festgelegter Zielzustand eintritt. Der Agent agiert somit nicht in einem linearen Ablauf, sondern in einer sequenziellen, kontextsensitiven Schleife mit fortlaufender Zustandsaktualisierung.

Agenten-Loop:

1. Aufnahme einer Eingabe (z. B. Nutzeranweisung, Dokument, Webinhalt)
2. Kontextanalyse durch das Sprachmodell
3. Ableitung einer nächsten Handlung
4. Optionaler Aufruf eines Tools oder einer Integration
5. Verarbeitung des Ergebnisses
6. Aktualisierung des Kontexts
7. Entscheidung über den nächsten Schritt

Dieser iterative Prozess endet entweder durch eine explizite Abbruchbedingung oder durch Erreichen eines definierten Zielzustands.

Sicherheitsrelevant ist hierbei, dass die einzelnen Entscheidungsschritte nicht ausschließlich regelbasiert deterministisch erfolgen. Sie werden vielmehr durch die probabilistische Modelllogik des Sprachmodells beeinflusst. Damit entsteht ein adaptiver Entscheidungsprozess, dessen konkrete Ausprägung im Einzelfall variieren kann. Entsprechend müssen Sicherheitsmechanismen nicht nur einzelne Aktionen, sondern den gesamten iterativen Entscheidungsraum des Agenten adressieren.

Tool-Calls und Systeminteraktion

OpenClaw ist in der Lage, externe Funktionen („Tools“) aufzurufen und in seine Entscheidungsprozesse einzubinden. Hierzu zählen unter anderem API-Zugriffe, Dateisystemoperationen, Datenbankabfragen, Web-Anfragen, Systembefehle sowie die Interaktion mit internen Unternehmensanwendungen. Der Agent beschränkt sich damit nicht auf reine Informationsverarbeitung, sondern kann aktiv auf operative Systemressourcen zugreifen.

Aus sicherheitstechnischer Perspektive verschiebt sich dadurch die maßgebliche Systemgrenze. Diese verläuft nicht ausschließlich entlang der Agentenapplikation oder des zugrunde liegenden Sprachmodells, sondern entlang der angebundenen Systeme und der jeweils eingeräumten Berechtigungen. Der effektive Sicherheitsumfang ergibt sich aus der Kombination von Modelllogik, Tool-Schnittstellen und Zugriffskontrollen.

Ein Tool-Call stellt technisch eine strukturierte Schnittstelle zwischen dem Sprachmodell und ausführbaren Systemfunktionen dar. Das Modell generiert dabei einen strukturierten Aufruf, der von der Laufzeitumgebung interpretiert und ausgeführt wird. Die Konfiguration dieser Schnittstellen – einschließlich Parametervalidierung, Berechtigungsprüfung, Logging und Fehlerbehandlung – ist daher ein zentraler Bestandteil der Sicherheitsarchitektur.

Unzureichend definierte oder überprivilegierte Tool-Schnittstellen können dazu führen, dass modellbasierte Fehlentscheidungen unmittelbare operative Auswirkungen entfalten. Entsprechend ist die Gestaltung und Absicherung der Tool-Integration als kritische Kontrollinstanz innerhalb der Gesamtarchitektur zu behandeln.

Kontextspeicher und Zustandsverwaltung

KI-Agenten operieren auf Basis eines dynamischen Kontextspeichers, der den aktuellen Arbeitszustand des Systems abbildet. Dieser Kontext kann bisherige Dialoginhalte, Zwischenergebnisse, eingebundene Dokumente sowie Statusinformationen zu laufenden Aufgaben enthalten. Er bildet die Grundlage für konsistente Entscheidungsprozesse innerhalb des Agenten-Loops.

Abhängig von Architektur und Deployment-Modell kann der Kontext ausschließlich temporär im Arbeitsspeicher gehalten oder persistent gespeichert werden. Persistente Speicherung ermöglicht längere Aufgabenketten und Nachvollziehbarkeit, erhöht jedoch zugleich die Anforderungen an Datenschutz, Zugriffskontrolle und Integritätssicherung.

Aus sicherheitstechnischer Sicht sind mehrere Aspekte besonders relevant. Hierzu zählen insbesondere die mögliche Speicherung sensibler Informationen im Kontext, die Zugriffsmöglichkeiten auf diese Daten, die Protokollierung und Nachvollziehbarkeit von Zustandsänderungen sowie die saubere Trennung unterschiedlicher Nutzer- oder Mandantenkontexte. Ohne klare Isolation besteht das Risiko, dass Informationen unzulässig aggregiert oder zwischen Sitzungen übertragen werden.

Eine unzureichend kontrollierte Kontextverwaltung kann zu Datenabfluss, ungewollter Informationsverknüpfung oder regulatorisch relevanten Datenschutzverletzungen führen. Entsprechend ist der Kontextspeicher nicht als rein technische Hilfsstruktur zu betrachten, sondern als sicherheitskritische Datenhaltungskomponente innerhalb der Gesamtarchitektur.

Integrationen und Erweiterungen

OpenClaw kann durch zusätzliche Integrationen und Erweiterungen funktional ausgebaut werden. Dies umfasst insbesondere die Anbindung von Cloud-Diensten, Kommunikationsschnittstellen, internen Fachsystemen sowie externen Datenquellen. Der Agent wird damit zu einer zentralen Orchestrierungskomponente innerhalb einer heterogenen Systemlandschaft.

Jede zusätzliche Integration erweitert die Systemgrenze und verändert die Vertrauensarchitektur. Anstelle einer klar abgegrenzten Anwendung entsteht eine verteilte Struktur, in der mehrere technische und organisatorische Vertrauenszonen miteinander verbunden sind. Datenflüsse verlaufen nicht mehr ausschließlich innerhalb einer Anwendung, sondern über System- und gegebenenfalls Organisationsgrenzen hinweg.

Aus sicherheitstechnischer Perspektive ist daher nicht nur die einzelne Integration isoliert zu bewerten, sondern die gesamte Integrationskette. Maßgeblich sind dabei insbesondere Authentisierungs- und Autorisierungsmechanismen, Datenübertragungswege, Protokollierung, Update- und Abhängigkeitsmanagement sowie die Absicherung externer Schnittstellen.

Eine fragmentierte Betrachtung einzelner Komponenten greift zu kurz. Die Sicherheitsbewertung muss systemisch erfolgen und alle verbundenen Dienste, Abhängigkeiten und Vertrauensbeziehungen in die Risikoanalyse einbeziehen.

Deployment-Modelle

Der Betrieb eines KI-Agenten kann in unterschiedlichen technischen Szenarien erfolgen. Möglich sind unter anderem eine lokale Installation auf Einzel- oder Serversystemen, eine containerisierte Bereitstellung, der Betrieb innerhalb virtualisierter Umgebungen, die Integration in bestehende Cloud-Infrastrukturen sowie hybride Modelle, die mehrere Ansätze kombinieren. Die Wahl des Deployment-Modells ist keine rein infrastrukturelle Entscheidung, sondern hat unmittelbare Auswirkungen auf Sicherheits- und Governance-Aspekte.

Insbesondere beeinflusst das gewählte Betriebsmodell die Netzwerkexposition des Systems, die Ausgestaltung von Zugriffskontrollen, die technische Mandantentrennung sowie die Umsetzbarkeit von Logging- und Monitoring-Konzepten. Auch die regulatorische Einordnung kann je nach Betriebsform variieren, etwa im Hinblick auf Datenlokation, Auftragsverarbeitung oder branchenspezifische Compliance-Anforderungen.

Während Entwicklungs- und Testumgebungen häufig mit reduzierten Schutzmechanismen betrieben werden, ist für produktive Einsätze – insbesondere in regulierten oder kritischen Infrastrukturen – eine isolierte und kontrollierte Betriebsumgebung essenziell. Hierzu zählen klar segmentierte Netzwerke, restriktive Berechtigungsmodelle, abgesicherte Schnittstellen sowie eine belastbare Überwachungs- und Incident-Response-Struktur.

Das Deployment-Modell bildet somit einen integralen Bestandteil der Sicherheitsarchitektur und ist frühzeitig in die Gesamtplanung einzubeziehen.

Zusammenfassung der architekturellen Implikationen

Aus technischer Perspektive ist OpenClaw nicht als isoliertes Softwareprodukt zu verstehen. Vielmehr fungiert das System als Integrations- und Orchestrierungsschicht zwischen dem zugrunde liegenden Sprachmodell, unternehmensinternen Datenbeständen, operativen Systemressourcen sowie externen Diensten und Schnittstellen. Der Agent verbindet diese Komponenten in einem dynamischen Entscheidungs- und Ausführungsprozess.

Die Sicherheitsbewertung darf sich daher nicht auf eine isolierte Analyse des Quellcodes oder einzelner Funktionen beschränken. Erforderlich ist eine mehrschichtige Betrachtung, die das Verhalten des Modells, die Ausgestaltung der Tool- und Integrationslogik, das zugrunde liegende Berechtigungsmodell, die Architektur der Kontext- und Datenhaltung sowie die konkrete Betriebsumgebung einbezieht.

Erst das Zusammenspiel dieser Ebenen bestimmt das tatsächliche Risikoprofil des Systems. Sicherheitsmaßnahmen müssen entsprechend ganzheitlich konzipiert werden und sowohl technische als auch organisatorische Kontrollmechanismen umfassen.

Dieses architekturelle Gesamtverständnis bildet die Grundlage für das im folgenden Kapitel entwickelte Bedrohungsmodell.

4. Bedrohungsmodell für KI-Agenten

Die Sicherheitsbewertung von KI-Agenten erfordert ein erweitertes Bedrohungsmodell, das über klassische Applikationssicherheit hinausgeht. Traditionelle Angriffsmuster wie SQL-Injection oder Cross-Site-Scripting bleiben grundsätzlich relevant, adressieren jedoch nur einen Teil des Risikos. KI-Agenten agieren zusätzlich kontextgetrieben, probabilistisch und – abhängig von ihrer Konfiguration – operativ handlungsfähig. Dadurch entstehen neuartige Angriffsflächen, die in herkömmlichen Sicherheitsmodellen nicht vollständig abgebildet sind.

Im Unterschied zu rein deterministischer Software beeinflussen semantische Interpretation, dynamische Kontextverarbeitung und autonome Tool-Interaktion das Systemverhalten. Risiken ergeben sich somit nicht nur aus fehlerhafter Implementierung, sondern auch aus der Interaktion zwischen Modelllogik, Eingabedaten, Integrationen und Berechtigungsstrukturen.

Ein angemessenes Bedrohungsmodell für Systeme wie OpenClaw muss daher sowohl klassische IT-Sicherheitsaspekte als auch modell- und architekturspezifische Risiken berücksichtigen. Im Folgenden werden die zentralen Bedrohungskategorien strukturiert dargestellt und in ihren systemischen Zusammenhängen eingeordnet.

Prompt Injection

Prompt Injection bezeichnet die gezielte Manipulation des Verhaltens eines KI-Agenten durch eingebettete Anweisungen innerhalb von Eingabedaten. Anders als bei klassischen Code-Injection-Angriffen erfolgt die Beeinflussung nicht auf syntaktischer Ebene, sondern durch semantische Steuerung der Modellinterpretation.

Da KI-Agenten unstrukturierte Inhalte wie Webseiten, E-Mails oder Dokumente analysieren und in ihren Kontext integrieren, können darin versteckte oder explizite Handlungsanweisungen enthalten sein. Beispiele hierfür sind Aufforderungen, vorherige Anweisungen zu ignorieren, gespeicherte Informationen weiterzugeben oder bestimmte Tools mit vorgegebenen Parametern aufzurufen. Solche Anweisungen werden vom Modell nicht als „Code“, sondern als Bestandteil des zu interpretierenden Kontexts verarbeitet.

Das zugrunde liegende Sprachmodell unterscheidet nicht inhärent zwischen legitimen Nutzereingaben und manipulativen Inhalten Dritter. Ohne zusätzliche Kontrollmechanismen kann eine eingebettete Anweisung daher in den Entscheidungsprozess einfließen und operative Aktionen auslösen.

Prompt Injection stellt somit keine klassische technische Schwachstelle im Sinne fehlerhaften Quellcodes dar, sondern eine systemimmanente Manipulationsmöglichkeit auf Ebene der Entscheidungslogik. Die Abwehr erfordert architektonische Schutzmaßnahmen, klare Trennung von Instruktions- und Inhaltskontext sowie restriktive Tool- und Berechtigungsmodelle.

Risikofolgen:

- Unerwünschte Tool-Calls
- Umgehung von Kontrolllogik
- Datenabfluss

- Veränderung von Systemzuständen

Indirect Prompt Injection

Indirect Prompt Injection bezeichnet die Manipulation des Agentenverhaltens über externe Inhalte, die automatisiert in den Verarbeitungskontext des Systems einfließen. Im Unterschied zur direkten Prompt Injection erfolgt die Beeinflussung nicht durch unmittelbare Interaktion mit dem Agenten, sondern mittelbar über eine Datenquelle, die vom System regulär verarbeitet wird.

KI-Agenten beziehen Informationen aus unterschiedlichen Drittquellen, etwa aus Webseiten, E-Mails, Dokumenten, Ticketsystemen oder API-Antworten. In solchen Inhalten können versteckte oder semantisch manipulativ formulierte Anweisungen enthalten sein, beispielsweise die Aufforderung, interne Sicherheitsrichtlinien zu ignorieren, Dateien herunterzuladen und auszuführen oder gespeicherte Zugangsdaten weiterzugeben. Diese Inhalte werden vom Modell als Bestandteil des Kontexts interpretiert und können – ohne zusätzliche Schutzmechanismen – Einfluss auf die Entscheidungslogik nehmen.

Der Angreifer interagiert hierbei nicht direkt mit dem Zielsystem, sondern kompromittiert oder präpariert eine Informationsquelle, die später automatisiert verarbeitet wird. Dadurch erweitert sich die Angriffsfläche auf sämtliche externen Datenquellen, die in den Agenten-Loop eingebunden sind.

Da das Sprachmodell nicht inhärent zwischen legitimen Kontextinformationen und versteckten Handlungsanweisungen unterscheidet, entsteht ein strukturelles Risiko. Die Absicherung erfordert daher eine klare Trennung zwischen Informationskontext und steuernden Instruktionen sowie zusätzliche Kontrollinstanzen vor der Ausführung operativer Aktionen.

Risikofolgen:

• Auslösung unbeabsichtigter Tool-Calls
• Verarbeitung und Weitergabe sensibler Informationen
• Umgehung interner Sicherheitsregeln
• Automatisierte Fehlhandlungen in angebundenen Systemen

Tool-Manipulation

KI-Agenten interagieren mit externen Tools und operativen Systemfunktionen. Jede freigegebene Funktion erweitert dabei die Angriffsfläche, da sie dem Agenten zusätzliche Handlungsmöglichkeiten eröffnet. Das Risiko entsteht nicht primär durch das Tool selbst, sondern durch die Kombination aus Modellentscheidung, Parametrisierung und Berechtigungsumfang.

Besondere Gefährdungen ergeben sich durch überprivilegierte Tool-Berechtigungen, fehlende oder unzureichende Eingabevalidierung vor der Ausführung eines Tool-Calls, unklar definierte oder zu weit gefasste Parametergrenzen sowie durch die Möglichkeit, mehrere Tools sequenziell zu kombinieren („Tool-Chaining“). In solchen Konstellationen kann eine einzelne Fehlentscheidung oder manipulierte Eingabe eine Kette operativer Aktionen auslösen.

Ein Agent kann – abhängig von seiner Konfiguration – formal legitime Tools in einer Weise einsetzen, die funktional unerwünscht oder sicherheitskritisch ist. Wird das Modell beispielsweise durch manipulierte Inhalte zu einer bestimmten Handlung veranlasst, kann es vorhandene Integrationen missbräuchlich nutzen, ohne dass ein klassischer Softwarefehler vorliegt.

Die wirksame Begrenzung dieses Risikos erfordert ein strikt definiertes Berechtigungsmodell, eine technische Validierung von Parametern vor der Ausführung sowie Kontrollmechanismen, die kritische Aktionen absichern oder genehmigungspflichtig machen. Tool-Schnittstellen sind somit als sicherheitskritische Kontrollpunkte innerhalb der Gesamtarchitektur zu behandeln.

Risikofolgen:

- Veränderung von Daten
- Zugriff auf sensible Ressourcen
- Automatisierte Fehlaktionen in Drittsystemen

Privilege Escalation

KI-Agenten agieren typischerweise nicht im Kontext eines einzelnen Endnutzers, sondern verwenden Service-Accounts, API-Tokens oder technische Identitäten zur Interaktion mit angebundenen Systemen. Der effektive Handlungsrahmen des Agenten wird daher maßgeblich durch die diesen Identitäten zugewiesenen Berechtigungen bestimmt.

Sind diese Berechtigungen zu weit gefasst, entsteht ein erhebliches Eskalationsrisiko. Ursachen hierfür sind häufig die Nutzung von administrativen oder „Superuser“-Accounts, eine fehlende Mandantentrennung, gemeinsam genutzte Token für mehrere Prozesse oder eine unzureichende Segmentierung der Betriebsumgebung. In solchen Konstellationen verfügt der Agent über weitergehende Rechte als für den konkreten Anwendungsfall erforderlich.

Wird der Agent manipuliert oder kompromittiert, entfaltet sich seine Wirkung wie die eines privilegierten automatisierten Benutzers. Dies kann den Zugriff auf sensible Datenbereiche, die Veränderung sicherheitsrelevanter Konfigurationen oder die Umgehung interner Kontrollmechanismen ermöglichen.

Die wirksame Begrenzung dieses Risikos erfordert die konsequente Umsetzung des Least-Privilege-Prinzips, eine saubere Trennung von Mandanten und Funktionsbereichen sowie die Verwendung eindeutig zugeordneter und zweckgebundener Zugangsdaten. Technische Identitäten sind als eigenständige Risikofaktoren zu behandeln und in bestehende Identity- und Access-Management-Prozesse zu integrieren.

Risikofolgen:

- Zugriff auf geschützte Datenbereiche
- Veränderung kritischer Konfigurationen
- Umgehung interner Kontrollmechanismen

Datenexfiltration

KI-Agenten verarbeiten und aggregieren Informationen aus unterschiedlichen internen und externen Quellen. Durch diese Zusammenführung entsteht ein kumulatives Datenrisiko, das über die isolierte Betrachtung einzelner Datenbestände hinausgeht. Informationen, die für sich genommen unkritisch erscheinen, können in aggregierter Form einen sensitiven oder regulatorisch relevanten Kontext ergeben.

Eine Exfiltration sensibler Daten kann auf unterschiedlichen Wegen erfolgen. Hierzu zählen insbesondere die direkte Weitergabe an externe APIs, die Generierung von Antworten mit vertraulichen Inhalten, die Speicherung sensibler Informationen in ungeschützten Logs oder deren Weiterverarbeitung durch angebundene Integrationen. In allen Fällen ist nicht zwingend ein klassischer Sicherheitsbruch erforderlich; bereits eine modellbasierte Fehlentscheidung oder unzureichende Zugriffskontrolle kann zu einer ungewollten Offenlegung führen.

Besonders kritisch ist die Fähigkeit des Agenten, Informationen kontextübergreifend zu verknüpfen. Durch die Aggregation mehrerer Datenquellen können implizite Zusammenhänge offengelegt oder geschützte Informationen rekonstruiert werden, obwohl die einzelnen Quellen isoliert betrachtet keine hohe Schutzbedürftigkeit aufweisen.

Die Absicherung gegen Datenexfiltration erfordert daher eine Kombination aus restriktivem Berechtigungsmanagement, kontrollierten Integrationen, sensibler Datenklassifikation sowie Monitoring- und Logging-Mechanismen, die ungewöhnliche Datenflüsse erkennen und nachvollziehbar machen.

Supply-Chain-Risiken (Skills, Plugins, Integrationen)

Erweiterbare Agentensysteme ermöglichen die Integration zusätzlicher Module, sogenannter „Skills“, Plugins oder externer Integrationen. Diese Komponenten können aus Drittquellen stammen, funktional nur eingeschränkt geprüft sein oder eigene technische Abhängigkeiten mitbringen. Mit jeder Erweiterung vergrößert sich nicht nur der Funktionsumfang, sondern auch die potenzielle Angriffsfläche.

Das Risiko entspricht in seiner Struktur klassischen Supply-Chain-Szenarien aus der Softwareentwicklung. Ungeprüfte oder kompromittierte Komponenten können Schadfunktionen enthalten, unsichere Update-Mechanismen nutzen oder indirekt weitere Abhängigkeiten einführen, die ihrerseits Schwachstellen aufweisen. Da solche Module häufig privilegierten Zugriff auf Agentenfunktionen oder Systemressourcen erhalten, können Sicherheitsdefizite unmittelbare Auswirkungen auf das Gesamtsystem haben.

Ohne eine strukturierte Überprüfung von Herkunft, Integrität, Versionsstand und Abhängigkeitsstruktur entsteht ein systemisches Risiko. Die Einbindung externer Erweiterungen sollte daher denselben Prüf- und Freigabeprozessen unterliegen wie andere geschäftskritische Softwarekomponenten, einschließlich Code-Review, Abhängigkeitsanalyse, Versionskontrolle und kontinuierlichem Schwachstellenmonitoring.

Risikofaktoren:

- Fehlende Code-Reviews

- Unklare Herkunft
- Unsichere Update-Mechanismen
- Ungeprüfte externe APIs

Fehlkonfiguration & Exponierte Instanzen

Ein erheblicher Anteil realer Sicherheitsvorfälle ist nicht auf hochkomplexe Angriffe zurückzuführen, sondern auf Fehlkonfigurationen oder unzureichend abgesicherte Betriebsumgebungen. Dieses Muster gilt in besonderem Maße für KI-Agenten, da sie häufig mit weitreichenden Systemrechten und Integrationen betrieben werden.

Typische Risikokonstellationen umfassen öffentlich erreichbare Instanzen ohne angemessene Zugriffsbeschränkung, fehlende oder unzureichende Authentisierungsmechanismen, mangelhafte Netzwerksegmentierung, unkontrollierten ausgehenden Internetzugang (Egress) sowie die unverschlüsselte Speicherung sensibler Daten. Solche Konfigurationsdefizite schaffen Angriffsflächen, die unabhängig von der Modelllogik bestehen und unmittelbar ausnutzbar sein können.

Die sicherheitstechnische Tragweite erhöht sich, wenn der Agent über privilegierte Service-Accounts, Datenbankzugriffe oder operative Systemrechte verfügt. In diesem Fall wirkt eine Fehlkonfiguration nicht isoliert, sondern potenziert die Auswirkungen möglicher Manipulationen oder unbefugter Zugriffe.

Die Absicherung beginnt daher nicht erst bei der Modell- oder Prompt-Ebene, sondern bei einer strukturierten Härtung der Betriebsumgebung. Sichere Standardkonfigurationen, restriktive Netzwerkrichtlinien, konsequente Authentisierung sowie Verschlüsselung sensibler Daten sind grundlegende Voraussetzungen für einen belastbaren Betrieb.

Systemische Besonderheit von KI-Agenten

Die zuvor beschriebenen Risiken sind nicht isoliert zu betrachten. Charakteristisch für KI-Agenten ist vielmehr die Möglichkeit, dass sich einzelne Schwachstellen entlang der Integrations- und Entscheidungslogik gegenseitig verstärken. Sicherheitsvorfälle entstehen häufig nicht durch einen einzelnen Fehler, sondern durch die Verkettung mehrerer Faktoren.

So kann etwa eine Prompt Injection eine missbräuchliche Tool-Nutzung auslösen, die in der Folge zu einer Datenexfiltration führt. Ebenso kann eine Fehlkonfiguration in Verbindung mit überprivilegierten Service-Accounts eine Privilege Escalation begünstigen und schließlich in einer umfassenden Systemkompromittierung münden. Auch Supply-Chain-Schwachstellen in Erweiterungen oder Integrationen können zu einem unkontrollierten Zugriff auf angebundene Systeme führen.

Diese Kaskadeneffekte sind Ausdruck der integrativen Architektur von KI-Agenten. Da Sprachmodell, Tool-Integration, Kontextspeicher, Berechtigungsmodell und Betriebsumgebung eng miteinander verzahnt sind, wirken sich Schwächen in einer Ebene potenziell auf das Gesamtsystem aus.

Die Sicherheitsbewertung muss daher integrativ, architekturbezogen und risikoorientiert erfolgen. Zudem ist eine regulatorisch anschlussfähige Dokumentation und Einordnung

erforderlich, um die Anforderungen bestehender Compliance- und Risikomanagementrahmenwerke nachvollziehbar abzubilden. Nur eine ganzheitliche Betrachtung ermöglicht eine belastbare Bewertung der tatsächlichen Systemrisiken.

Methodische Einordnung

Das dargestellte Bedrohungsmodell für KI-Agenten lässt sich in etablierte sicherheitstechnische Frameworks einordnen. Eine solche Zuordnung ist nicht zwingend erforderlich, erleichtert jedoch die Integration in bestehende Informationssicherheits- und Risikomanagementprozesse.

So können die beschriebenen Risikokategorien beispielsweise entlang des STRIDE-Modells strukturiert werden, insbesondere in den Dimensionen Information Disclosure, Elevation of Privilege, Tampering und Repudiation. Prompt-Injection- und Tool-Manipulationsszenarien lassen sich zudem als spezifische Ausprägungen von Manipulation und unautorisiertem Zugriff interpretieren.

Ebenso ist eine Abbildung auf die MITRE-ATT&CK-Matrix möglich. Relevante Taktiken betreffen unter anderem Credential Access, Execution, Persistence und Exfiltration. Insbesondere bei Agenten mit Systemzugriffen und Integrationen lassen sich typische Angriffsketten modellieren, die bekannten Angriffstechniken ähneln, jedoch durch die modellbasierte Entscheidungslogik eine zusätzliche semantische Komponente erhalten.

Die methodische Einordnung in solche Frameworks unterstützt die Anschlussfähigkeit an bestehende ISMS-Strukturen, erleichtert die Dokumentation gegenüber internen Revisionen oder externen Prüfern und ermöglicht eine systematische Integration in unternehmensweite Risikoregister. KI-Agentenspezifische Risiken werden dadurch nicht isoliert betrachtet, sondern in die etablierte Sicherheitsgovernance eingebettet.

5. Regulatorische Einordnung

Der produktive Einsatz von KI-Agenten wie OpenClaw erfolgt nicht im regulatorischen Vakuum.
Abhängig von Branche, Einsatzkontext und Integrationsgrad können unterschiedliche gesetzliche und normative Anforderungen relevant werden.

KI-Agenten sind technisch neuartig, unterliegen jedoch bestehenden Sicherheits- und Governance-Anforderungen. Entscheidend ist nicht das verwendete Tool, sondern dessen Einbettung in die Unternehmensarchitektur.

Der produktive Einsatz von KI-Agenten wie OpenClaw erfolgt nicht im regulatorischen Vakuum. Abhängig von Branche, Kritikalität der verarbeiteten Daten, Integrationsgrad sowie konkretem Anwendungsfall können unterschiedliche gesetzliche und normative Anforderungen einschlägig sein. Die technologische Neuartigkeit des Systems führt nicht zu einer regulatorischen Sonderstellung, sondern ist in bestehende Sicherheits- und Governance-Rahmenwerke einzuordnen.

Maßgeblich ist dabei nicht primär das eingesetzte Tool, sondern dessen funktionale Einbettung in die Unternehmensarchitektur. Ein KI-Agent, der ausschließlich unterstützende Textanalysen durchführt, ist regulatorisch anders zu bewerten als ein System, das automatisiert auf Produktivdaten zugreift, Geschäftsprozesse steuert oder in sicherheitskritische Infrastrukturen integriert ist.

Bestehende Anforderungen aus Informationssicherheits-, Datenschutz- und IT-Sicherheitsgesetzen sowie branchenspezifischen Regelwerken bleiben grundsätzlich anwendbar. Hierzu zählen insbesondere Vorgaben zum Risikomanagement, zur Zugriffskontrolle, zur Protokollierung, zur Datensicherheit sowie zur Nachvollziehbarkeit automatisierter Entscheidungen. In regulierten Sektoren können zusätzliche Anforderungen an Dokumentation, Prüfpfade und technische Absicherung bestehen.

Für Unternehmen bedeutet dies, dass KI-Agenten nicht isoliert als Innovationsprojekt betrachtet werden dürfen. Ihre Einführung ist vielmehr als Bestandteil der bestehenden IT-Governance zu behandeln. Eine frühzeitige regulatorische Einordnung, dokumentierte Risikoanalyse und klare Verantwortlichkeitszuweisung sind zentrale Voraussetzungen für einen belastbaren und rechtskonformen Betrieb.

ISO/IEC 27001 und BSI IT-Grundschutz

Im Rahmen eines Informationssicherheitsmanagementsystems (ISMS) nach ISO/IEC 27001 oder auf Basis des BSI IT-Grundschutzes sind KI-Agenten als informationsverarbeitende Systeme zu klassifizieren. Sie unterliegen damit denselben grundlegenden Anforderungen wie andere geschäftskritische Anwendungen, unabhängig von ihrer technologischen Besonderheit.

Zentrale Anforderungen betreffen insbesondere die Schutzbedarfsfeststellung der verarbeiteten Informationen, die strukturierte Risikobewertung und -behandlung, die Umsetzung angemessener Zugriffskontrollen sowie die Protokollierung und Überwachung sicherheitsrelevanter Ereignisse. Darüber hinaus sind das Management von Änderungen – etwa bei Modellversionen, Integrationen oder Berechtigungen – sowie das Lieferanten- und

Drittparteienmanagement relevant, insbesondere bei der Nutzung externer Modelle, Cloud-Dienste oder Erweiterungen.

Da KI-Agenten häufig mehrere Systemkomponenten integrieren und auf unterschiedliche Datenquellen zugreifen, ist ihre Einordnung nicht isoliert vorzunehmen. Vielmehr sind sie in die bestehende Asset-Struktur, in Risikoanalysen und in Kontrollmaßnahmen des ISMS einzubetten. Dies umfasst auch die Dokumentation von Verantwortlichkeiten, Schnittstellen und Abhängigkeiten.

Der produktive Einsatz eines KI-Agenten erfordert daher eine systematische Integration in das bestehende Risikomanagement. Nur durch eine konsistente Einbindung in etablierte Sicherheitsprozesse lässt sich ein nachvollziehbarer und prüffähiger Betrieb gewährleisten.

NIS2-Richtlinie

Die NIS2-Richtlinie verpflichtet betroffene Unternehmen zur Umsetzung angemessener technischer und organisatorischer Maßnahmen zur Beherrschung von Cyberrisiken. Ziel ist es, die Resilienz kritischer und wesentlicher Einrichtungen gegenüber IT-Sicherheitsvorfällen systematisch zu stärken.

Werden KI-Agenten wie OpenClaw in geschäftskritischen Prozessen eingesetzt, sind sie als Bestandteil der eingesetzten IKT-Systeme zu bewerten. Damit unterliegen sie den Anforderungen an ein strukturiertes Risikomanagement, das sowohl technische als auch organisatorische Schutzmaßnahmen umfasst.

Im Kontext von KI-Agenten sind insbesondere folgende Aspekte relevant: die Durchführung und Dokumentation von Risikomanagementmaßnahmen für IKT-Systeme, die Fähigkeit zur frühzeitigen Erkennung und Meldung von Sicherheitsvorfällen, die Absicherung der Lieferkette – etwa bei externen Modellen, Integrationen oder Erweiterungen – sowie wirksame Zugriffskontrollen und Maßnahmen zur Aufrechterhaltung des Geschäftsbetriebs (Business Continuity).

KI-Agenten, die in operative Abläufe eingebunden sind oder auf sensible Daten zugreifen, beeinflussen unmittelbar das Gesamtrisiko der IKT-Landschaft. Sie sind daher in bestehende Risikoanalysen, Incident-Response-Prozesse und Meldeverfahren einzubeziehen. Eine isolierte Betrachtung als reines Innovationsprojekt wäre im Anwendungsbereich der NIS2-Richtlinie nicht ausreichend.

Cyber Resilience Act (CRA)

Der Cyber Resilience Act (CRA) adressiert primär Hersteller, Importeure und Händler digitaler Produkte mit vernetzten oder softwarebasierten Komponenten. Ziel ist die Einführung einheitlicher Sicherheitsanforderungen über den gesamten Produktlebenszyklus hinweg.

Organisationen, die KI-Agenten in eigene Produkte integrieren oder als Bestandteil einer marktfähigen Lösung bereitstellen, können mittelbar oder unmittelbar in den Anwendungsbereich des CRA fallen. Maßgeblich ist nicht die interne Nutzung eines Systems, sondern dessen Einbettung in ein Produkt, das am Markt bereitgestellt wird.

Relevante Anforderungen betreffen insbesondere die Etablierung eines sicheren Entwicklungsprozesses (Secure Development Lifecycle), die Dokumentation sicherheitsrelevanter Eigenschaften und Risiken, ein strukturiertes Schwachstellenmanagement einschließlich koordinierter Offenlegung von Sicherheitslücken sowie die Sicherstellung von Update- und Patchfähigkeit über den definierten Unterstützungszeitraum hinweg.

Wird ein KI-Agent als integraler Bestandteil eines Produkts eingesetzt – etwa zur Steuerung von Funktionen, zur Verarbeitung sicherheitsrelevanter Daten oder zur Interaktion mit externen Systemen –, kann er Teil der produktbezogenen Sicherheitsbewertung werden. In diesem Fall sind nicht nur betriebliche Schutzmaßnahmen, sondern auch produktspezifische Konformitäts- und Dokumentationspflichten zu berücksichtigen.

Unternehmen sollten daher frühzeitig prüfen, ob der Einsatz eines KI-Agenten ausschließlich als internes IT-System erfolgt oder ob eine Einbindung in ein marktbereitgestelltes Produkt vorliegt. Diese Differenzierung ist entscheidend für die regulatorische Einordnung im Kontext des Cyber Resilience Act.

AI Act

Der AI Act etabliert einen risikobasierten Regulierungsrahmen für KI-Systeme innerhalb der Europäischen Union. Die regulatorischen Anforderungen richten sich nicht pauschal nach der eingesetzten Technologie, sondern nach dem konkreten Anwendungsrisiko und dem Einsatzkontext.

Für KI-Agenten sind insbesondere mehrere Aspekte relevant. Hierzu zählen die Risikoklassifizierung des eingesetzten Systems, die Implementierung geeigneter Governance- und Risikomanagementstrukturen, die Erfüllung von Dokumentations- und Transparenzpflichten sowie Anforderungen an Überwachung, Nachvollziehbarkeit und menschliche Aufsicht. Maßgeblich ist dabei, ob und in welchem Umfang der Agent autonome Entscheidungen trifft oder Prozesse mit rechtlicher oder faktischer Wirkung beeinflusst.

Ob ein KI-Agent als Hochrisiko-KI-System einzustufen ist, hängt vom konkreten Einsatzszenario ab. Anwendungen im Umfeld kritischer Infrastrukturen, bei Personalentscheidungen, im Finanzsektor oder in anderen regulierten Bereichen können unter die Hochrisiko-Kategorie fallen. In solchen Fällen gelten erweiterte Anforderungen an Risikomanagement, Datenqualität, technische Dokumentation, Logging und Marktüberwachung.

Auch bei der Nutzung eines Open-Source-Systems verbleibt die regulatorische Verantwortung nicht beim ursprünglichen Entwickler, sondern beim Betreiber, Anbieter oder Integrator, der das System in Verkehr bringt oder produktiv einsetzt. Entscheidend ist somit nicht die Herkunft der Software, sondern die konkrete Implementierung und Zweckbestimmung im jeweiligen Organisationskontext.

Telekommunikationsgesetz (TKG)

Unternehmen im Telekommunikationssektor unterliegen besonderen gesetzlichen Sicherheitsanforderungen gemäß dem Telekommunikationsgesetz (TKG). Ziel dieser

Vorgaben ist es, die Integrität, Verfügbarkeit und Vertraulichkeit von Netzen und Diensten sicherzustellen und die Resilienz der Telekommunikationsinfrastruktur zu gewährleisten.

Zentrale Anforderungen betreffen insbesondere die Umsetzung des Stands der Technik. Technische und organisatorische Maßnahmen müssen dem jeweils aktuellen Sicherheitsniveau entsprechen und geeignet sein, identifizierte Risiken wirksam zu minimieren. Darüber hinaus sind strukturierte Sicherheitskonzepte zu erstellen, regelmäßig zu überprüfen und fortzuschreiben. Änderungen in der Systemarchitektur, etwa durch die Integration neuer Technologien wie KI-Agenten, sind dabei systematisch zu berücksichtigen.

Ein weiterer Schwerpunkt liegt auf der Nachweisführung gegenüber zuständigen Aufsichtsbehörden, insbesondere der Bundesnetzagentur. Unternehmen müssen die Einhaltung der gesetzlichen Anforderungen dokumentieren und prüffähig belegen können. Dies umfasst auch Risikoanalysen, Schutzmaßnahmen und gegebenenfalls Sicherheitsvorfälle.

Besonders schützenswerte Betriebsprozesse – etwa der Netzbetrieb, Steuerungs- und Managementsysteme oder Authentifizierungsmechanismen – unterliegen erhöhten Anforderungen an Absicherung und Überwachung. Wird ein KI-Agent in solchen betriebsrelevanten oder sicherheitskritischen Abläufen eingesetzt, ist er als integraler Bestandteil der sicherheitsrelevanten Systemlandschaft zu betrachten. Entsprechend ist er in bestehende Sicherheitskonzepte, Risikoanalysen und Kontrollmechanismen einzubeziehen und hinsichtlich seiner spezifischen Risiken gesondert zu bewerten.

Kritische Infrastrukturen (KRITIS)

Unternehmen, die als Betreiber Kritischer Infrastrukturen (KRITIS) eingestuft sind, unterliegen erhöhten gesetzlichen Anforderungen an die IT- und Informationssicherheit. Ziel ist der Schutz von Systemen und Prozessen, deren Ausfall erhebliche Versorgungsengpässe oder Gefährdungen für die öffentliche Sicherheit verursachen könnte.

Zentral ist die Verpflichtung zur Umsetzung angemessener organisatorischer und technischer Maßnahmen nach dem Stand der Technik. Diese Maßnahmen müssen geeignet sein, Störungen der Verfügbarkeit, Integrität, Authentizität und Vertraulichkeit der eingesetzten Systeme zu vermeiden oder deren Auswirkungen zu begrenzen. Die Anforderungen erstrecken sich dabei auf die gesamte sicherheitsrelevante Systemlandschaft.

Darüber hinaus sind Sicherheitskonzepte systematisch zu erstellen, regelmäßig zu überprüfen und fortzuschreiben. Änderungen in der Architektur – etwa durch die Einführung eines KI-Agenten – sind risikobasiert zu bewerten und in bestehende Schutzkonzepte zu integrieren. Ergänzend bestehen Nachweis- und Meldepflichten gegenüber dem Bundesamt für Sicherheit in der Informationstechnik (BSI). Sicherheitsvorfälle sind fristgerecht zu melden, und die Wirksamkeit der implementierten Maßnahmen ist regelmäßig prüffähig zu dokumentieren.

Kommt ein KI-Agent in sicherheitsrelevanten oder betriebssteuernden Prozessen zum Einsatz, ist er als Bestandteil der KRITIS-relevanten Infrastruktur zu behandeln. Dies gilt insbesondere dann, wenn er operative Entscheidungen vorbereitet, Systemzugriffe ausführt oder sensible Daten verarbeitet. Entsprechend sind umfassende Risikoanalysen, technische

und organisatorische Kontrollmechanismen, kontinuierliches Monitoring sowie klar definierte Verantwortlichkeiten erforderlich. Die Integration muss nachvollziehbar in das bestehende Sicherheits- und Compliance-Framework eingebettet werden.

Konsequenz für Betreiber

Die regulatorische Einordnung eines KI-Agenten richtet sich nicht nach seiner Bezeichnung oder technologischen Einordnung, sondern nach seiner konkreten Funktion im Unternehmen. Entscheidend ist, welche Rolle das System innerhalb der IT- und Prozesslandschaft übernimmt und welche Auswirkungen ein Fehlverhalten oder Ausfall haben könnte.

Maßgebliche Bewertungskriterien sind insbesondere der Umfang der Systemzugriffe, die Art und Sensitivität der verarbeiteten Daten, der Integrationsgrad in bestehende Systeme sowie der Einfluss auf geschäftskritische oder sicherheitsrelevante Prozesse. Je stärker ein KI-Agent in operative Abläufe eingebunden ist, desto höher sind die Anforderungen an Risikomanagement, Kontrolle und Dokumentation.

KI-Agenten sind daher als reguläre – gegebenenfalls sicherheitskritische – IT-Systeme zu behandeln. Ihre Einführung sollte auf Basis einer strukturierten Risikoanalyse erfolgen, nachvollziehbar dokumentiert und in bestehende Governance-, Sicherheits- und Compliance-Strukturen integriert werden. Technologische Innovation entbindet nicht von regulatorischer Sorgfaltspflicht.

Dieser Leitfaden stellt keine Rechtsberatung dar. Er dient der strukturierten Einordnung sicherheitsrelevanter Aspekte beim Einsatz von KI-Agenten im Unternehmenskontext und soll Unternehmen bei einer risikoorientierten Bewertung unterstützen.

6. Secure-by-Design-Architektur

Der sichere Einsatz von KI-Agenten setzt eine Architektur voraus, die Risiken nicht erst im laufenden Betrieb adressiert, sondern bereits auf konzeptioneller Ebene systematisch reduziert. Sicherheitsmechanismen dürfen nicht als nachgelagerte Ergänzung verstanden werden, sondern müssen integraler Bestandteil von Design, Implementierung und Integration sein.

Secure-by-Design bedeutet in diesem Kontext, die besonderen Eigenschaften von KI-Agenten – insbesondere Kontextverarbeitung, Tool-Integration, probabilistische Entscheidungslogik und autonome Handlungsfähigkeit – von Beginn an in die Sicherheitsarchitektur einzubeziehen. Ziel ist es, Entscheidungsräume technisch zu begrenzen, Integrationen kontrollierbar zu gestalten und Auswirkungen potenzieller Fehlentscheidungen zu isolieren.

Eine belastbare Secure-by-Design-Architektur umfasst dabei sowohl technische als auch organisatorische Maßnahmen. Hierzu zählen unter anderem eine klare Segmentierung der Betriebsumgebung, ein restriktives Berechtigungsmodell nach dem Least-Privilege-Prinzip, abgesicherte Schnittstellen zu externen Systemen, kontrollierte Datenflüsse sowie umfassende Logging- und Monitoring-Mechanismen.

Die nachfolgenden Maßnahmen bilden ein strukturiertes Referenzmodell für die sichere Architektur von KI-Agenten im Unternehmenskontext. Sie dienen als Orientierung für Planung, Implementierung und Auditierung und können je nach Risikoprofil und regulatorischem Umfeld angepasst werden.

Isolierung und Betriebsumgebung

KI-Agenten sollten nicht als unmittelbarer Bestandteil produktiver Kernsysteme betrieben werden, sondern in klar abgegrenzten und kontrollierten Umgebungen. Aufgrund ihrer Integrationsfähigkeit und Entscheidungsdynamik ist eine technische und organisatorische Entkopplung von geschäftskritischen Systemen essenziell.

Eine geeignete Betriebsarchitektur umfasst in der Regel eine containerisierte oder virtualisierte Bereitstellung, um Laufzeitumgebungen voneinander zu isolieren und reproduzierbar zu gestalten. Ergänzend ist eine konsequente Netzwerksegmentierung mit klar definierten und dokumentierten Kommunikationspfaden erforderlich. Datenflüsse und Schnittstellen sollten explizit freigegeben und restriktiv konfiguriert werden.

Zudem ist eine saubere Trennung von Entwicklungs-, Test- und Produktivumgebungen sicherzustellen. Änderungen an Modellkonfiguration, Integrationen oder Berechtigungen dürfen nicht unkontrolliert in produktive Systeme übernommen werden. Direkte Systemzugriffe des Agenten sollten auf das zwingend erforderliche Maß reduziert und, wo möglich, über kontrollierte Schnittstellen abstrahiert werden.

Die konsequente Isolierung der Betriebsumgebung dient der Schadensbegrenzung. Sie reduziert die Auswirkungen potenzieller Fehlentscheidungen, Manipulationen oder Kompromittierungen und bildet eine zentrale Grundlage für eine belastbare Secure-by-Design-Architektur.

Identity & Access Management

KI-Agenten agieren typischerweise über technische Identitäten wie Service-Accounts oder API-Tokens. Diese Identitäten bestimmen den effektiven Handlungsspielraum des Systems und sind daher als sicherheitskritische Komponenten zu behandeln. Eine unzureichende Kontrolle kann dazu führen, dass Fehlentscheidungen oder Manipulationen unmittelbare operative Auswirkungen entfalten.

Zentral ist die konsequente Anwendung des Least-Privilege-Prinzips. Der Agent darf ausschließlich die Berechtigungen erhalten, die für den definierten Anwendungsfall zwingend erforderlich sind. Darüber hinaus ist eine saubere Mandantentrennung sicherzustellen, insbesondere wenn mehrere Organisationseinheiten oder Datenbereiche betroffen sind.

Empfohlen wird die Verwendung separater technischer Identitäten für unterschiedliche Funktionsbereiche oder Integrationen. Dadurch lassen sich Rechte granular steuern und im Falle eines Sicherheitsvorfalls gezielt einschränken oder entziehen. Tokens sollten zeitlich begrenzt und regelmäßig rotiert werden, um das Risiko missbräuchlicher Nutzung zu reduzieren.

Ergänzend ist eine regelmäßige Überprüfung und Rezertifizierung vergebener Berechtigungen erforderlich. Änderungen im Einsatzkontext oder in der Systemarchitektur müssen sich unmittelbar im Berechtigungsmodell widerspiegeln. Überprivilegierte Accounts stellen eines der größten Risiken im Agentenbetrieb dar und sind konsequent zu vermeiden.

Secret- und Token-Handling

API-Schlüssel, Zugangsdaten, Zertifikate und sonstige Geheimnisse sind als besonders schützenswerte Informationen zu behandeln. Sie dürfen weder im Klartext in Konfigurationsdateien noch innerhalb von Prompts oder statischen Systemparametern gespeichert werden. Eine unsachgemäße Ablage kann dazu führen, dass sensible Zugangsdaten unbeabsichtigt im Kontext des Sprachmodells verarbeitet oder über Logs und Antworten offengelegt werden.

Empfohlen wird der Einsatz zentraler Secret-Management-Systeme, die eine sichere Speicherung, Zugriffskontrolle und Protokollierung ermöglichen. Der Zugriff auf Geheimnisse sollte strikt rollenbasiert erfolgen und auf die jeweils erforderlichen Funktionen begrenzt sein. Ergänzend sind regelmäßige Rotationsmechanismen für Tokens und Schlüssel vorzusehen, um das Risiko langfristig kompromittierter Zugangsdaten zu reduzieren.

Zugriffe auf Secrets sollten nachvollziehbar protokolliert und in bestehende Monitoring- und Audit-Prozesse integriert werden. Dabei ist sicherzustellen, dass das Sprachmodell selbst keinen direkten Zugriff auf Klartext-Geheimnisse erhält. Der Zugriff auf sensible Informationen sollte ausschließlich über kontrollierte Laufzeitmechanismen erfolgen, bei denen das Modell lediglich strukturierte, funktionsgebundene Parameter übergibt, ohne Kenntnis der zugrunde liegenden Zugangsdaten zu erhalten.

Ein strukturiertes Secret- und Token-Handling ist damit ein zentraler Baustein der Secure-by-Design-Architektur und wesentlich für die Begrenzung operativer Risiken im Agentenbetrieb.

Kontrolle von Tool-Calls

Die Entscheidung zur Ausführung eines Tools darf nicht ausschließlich dem Sprachmodell überlassen werden. Auch wenn das Modell eine Handlung vorschlägt, muss die tatsächliche Ausführung durch eine externe Kontrollinstanz abgesichert werden. Dadurch wird verhindert, dass modellbasierte Fehlinterpretationen oder manipulierte Eingaben unmittelbar operative Auswirkungen entfalten.

Zentral ist ein Whitelisting-Ansatz, bei dem ausschließlich explizit freigegebene Tools verfügbar sind. Darüber hinaus müssen Parameter vor der Ausführung technisch validiert werden, um unerwartete oder sicherheitskritische Eingabekombinationen zu verhindern. Ergänzend empfiehlt sich eine policy-basierte Genehmigungslogik, die definierte Regeln – etwa hinsichtlich Datenumfang, Zielsystem oder Berechtigungsstufe – automatisiert überprüft.

Weitere Schutzmechanismen umfassen Rate-Limiting zur Begrenzung automatisierter Aufrufketten sowie die Einbindung eines Human-in-the-Loop bei besonders sensiblen oder irreversiblen Aktionen. In solchen Fällen sollte eine manuelle Freigabe erfolgen, bevor der Tool-Call ausgeführt wird.

Durch die Implementierung einer Kontrollschicht außerhalb des Sprachmodells wird die Vorhersagbarkeit erhöht und das Missbrauchspotenzial deutlich reduziert. Tool-Calls sind somit nicht als rein funktionale Schnittstellen, sondern als sicherheitskritische Kontrollpunkte innerhalb der Gesamtarchitektur zu behandeln.

Egress-Kontrolle und Netzwerkrestriktion

Ein unkontrollierter Internetzugang stellt im Betrieb von KI-Agenten ein erhebliches Risiko dar. Da diese Systeme potenziell eigenständig externe Anfragen initiieren können, erhöht eine uneingeschränkte Netzwerkanbindung die Gefahr von Datenexfiltration, unerwünschten Integrationen oder Supply-Chain-Angriffen über kompromittierte Drittressourcen.

Eine wirksame Absicherung erfordert die konsequente Beschränkung ausgehender Verbindungen auf definierte und dokumentierte Ziele. Technisch kann dies durch Netzwerksegmentierung, Firewall-Regeln und DNS- beziehungsweise Domain-Allowlisting umgesetzt werden. Der Agent sollte ausschließlich mit explizit freigegebenen Endpunkten kommunizieren dürfen.

Ergänzend empfiehlt sich die Integration eines zentralen Proxys oder Secure Gateways, über den sämtliche ausgehenden Verbindungen geführt und kontrolliert werden. Dadurch lassen sich Verbindungsziele, Datenvolumen und Protokolle zentral überwachen. Ein kontinuierliches Monitoring externer API-Zugriffe unterstützt zudem die frühzeitige Erkennung ungewöhnlicher Kommunikationsmuster.

KI-Agenten sollten somit nicht als frei agierende Internet-Clients betrieben werden, sondern innerhalb klar definierter Netzwerkrestriktionen. Eine kontrollierte Egress-Strategie ist ein wesentlicher Bestandteil der Secure-by-Design-Architektur und trägt maßgeblich zur Begrenzung systemischer Risiken bei.

Logging, Monitoring und Nachvollziehbarkeit

KI-Agenten erzeugen komplexe, mehrstufige Entscheidungsprozesse, deren Ablauf für Sicherheits-, Compliance- und Audit-Zwecke nachvollziehbar dokumentiert werden muss. Aufgrund der nicht-deterministischen Modelllogik ist eine transparente Protokollierung zentral, um Handlungen im Nachhinein analysieren und bewerten zu können.

Erforderlich ist eine strukturierte Protokollierung relevanter Eingaben, Systementscheidungen und Modellantworten. Ebenso müssen sämtliche Tool-Calls einschließlich Parameter, Zielsystem und Rückgabewert revisionssicher erfasst werden. Änderungen am Kontext – etwa durch Integration externer Inhalte oder Zwischenergebnisse – sollten ebenfalls dokumentiert werden, um Entscheidungswege rekonstruieren zu können.

Darüber hinaus ist ein kontinuierliches Monitoring ungewöhnlicher oder sicherheitsrelevanter Aktivitäten notwendig. Hierzu zählen beispielsweise atypische Zugriffsmuster, erhöhte Tool-Nutzung, ungewöhnliche Datenvolumina oder abweichende Kommunikationsziele. Die Einbindung in bestehende SIEM-Systeme ermöglicht eine zentrale Korrelation mit anderen sicherheitsrelevanten Ereignissen innerhalb der IT-Landschaft.

Nachvollziehbarkeit ist nicht nur aus sicherheitstechnischer Sicht essenziell, sondern auch regulatorisch von Bedeutung. Anforderungen aus Informationssicherheitsstandards, branchenspezifischen Regelwerken oder dem AI Act setzen eine dokumentierte Überwachungs- und Kontrollstruktur voraus. Logging und Monitoring sind daher integrale Bestandteile einer belastbaren Secure-by-Design-Architektur.

Zwei-Agenten-Modell (Sandbox & Production)

Eine bewährte architektonische Praxis ist die klare Trennung zwischen einem explorativen beziehungsweise experimentellen Agentenbetrieb und einem produktiv eingesetzten, restriktiv konfigurierten Agenten. Dieses Zwei-Agenten-Modell dient der strukturierten Risikobegrenzung und verhindert, dass experimentelle Funktionen unmittelbar operative Auswirkungen entfalten.

Der Sandbox-Agent wird für Testzwecke, zur Evaluierung neuer Skills oder Integrationen sowie für experimentelle Entwicklungsarbeiten eingesetzt. In dieser Umgebung können neue Modelle, Konfigurationen oder Workflows erprobt werden, ohne dass produktive Datenbestände oder geschäftskritische Systeme betroffen sind. Die Sandbox sollte dabei technisch isoliert und klar von produktiven Ressourcen getrennt sein.

Der Produktiv-Agent hingegen ist auf einen definierten Anwendungszweck beschränkt. Er verfügt ausschließlich über minimale, zweckgebundene Berechtigungen, arbeitet in einer segmentierten und kontrollierten Umgebung und unterliegt einer strikten Protokollierung und Überwachung. Änderungen an Konfiguration oder Integrationen sollten nur über definierte Freigabeprozesse erfolgen.

Die konsequente Trennung zwischen Entwicklungs- und Produktivbetrieb reduziert das Risiko unkontrollierter Erweiterungen und unbeabsichtigter Funktionserweiterungen. Sie unterstützt zudem eine strukturierte Governance, indem Innovation und Stabilität organisatorisch und technisch voneinander abgegrenzt werden.

Consensus Mechanism

Ein Consensus Mechanism beschreibt eine architektonische Sicherheitsmaßnahme, bei der mehrere unabhängige Agenten oder Modellinstanzen zu einem gemeinsamen Ergebnis gelangen müssen, bevor eine sicherheitsrelevante Aktion ausgeführt wird. Anstelle einer singulären Modellentscheidung wird eine Mehrinstanzen-Validierung implementiert.

In der Praxis kann dies beispielsweise bedeuten, dass mehrere Agenten denselben Sachverhalt unabhängig bewerten und eine Aktion – etwa ein Tool-Call, eine Datenfreigabe oder eine Systemänderung – nur dann ausgeführt wird, wenn ein definiertes Entscheidungsquorum erreicht wird (z. B. Mehrheitsentscheidung oder einstimmige Zustimmung).

Ein solcher Mechanismus kann unterschiedliche Ausprägungen haben:

- Mehrheitsbasierte Entscheidungslogik („Voting“)
- Trennung von Entscheidungs- und Kontrollagent (z. B. Executor und Validator)
- Kombination unterschiedlicher Modelltypen zur Reduzierung systematischer Fehlbewertungen
- Eskalation an einen Human-in-the-Loop bei divergierenden Bewertungen

Ziel ist die Reduktion einzelner Fehlentscheidungen, insbesondere in Szenarien mit erhöhtem Risiko, etwa bei sensiblen Datenzugriffen oder irreversiblen Systemaktionen. Durch die Verteilung der Entscheidungslogik auf mehrere Instanzen wird die Wahrscheinlichkeit reduziert, dass eine einzelne manipulierte oder fehlerhafte Bewertung unmittelbar operative Auswirkungen entfaltet.

Ein Consensus Mechanism ersetzt jedoch keine grundlegende Sicherheitsarchitektur. Er wirkt ergänzend zu Berechtigungsmodellen, Tool-Restriktionen und Governance-Prozessen. Zudem erhöht er Komplexität, Ressourcenbedarf und potenzielle Latenz. Die Implementierung sollte daher risikobasiert erfolgen und insbesondere bei sicherheitskritischen oder regulatorisch sensiblen Anwendungsfällen in Betracht gezogen werden.

Cross-Validation / Cross-Monitoring

Cross-Validation beziehungsweise Cross-Monitoring beschreibt eine architektonische Kontrollmaßnahme, bei der Agenten die Ergebnisse oder Entscheidungen anderer Agenten überprüfen. Ziel ist es, Abweichungen, Inkonsistenzen oder potenziell sicherheitskritische Handlungen frühzeitig zu erkennen und gegebenenfalls zu eskalieren.

Im Unterschied zu einem reinen Mehrheitsmechanismus steht hier nicht die kollektive Entscheidungsfindung im Vordergrund, sondern die gegenseitige Überwachung. Ein primärer Agent kann beispielsweise eine Handlung vorbereiten oder eine Bewertung vornehmen, während ein zweiter, unabhängiger Agent diese Entscheidung anhand definierter Kriterien überprüft. Bei signifikanten Abweichungen oder Verstößen gegen Policies wird ein Alarm ausgelöst oder die Aktion blockiert.

Typische Anwendungsformen umfassen:

- Validierung von Tool-Calls durch einen separaten Kontrollagenten

- Überprüfung von Antworten auf potenziell sensible Dateninhalte
- Policy-Checks vor der Ausführung operativer Aktionen
- Erkennung ungewöhnlicher Entscheidungs- oder Zugriffsmuster

Ein wesentliches Ziel besteht darin, systematische Fehlentscheidungen, Prompt-Injection-Effekte oder unerwartete Kontextmanipulationen zu erkennen. Durch die Trennung von ausführender und überwachender Instanz wird die Wahrscheinlichkeit reduziert, dass ein einzelner kompromittierter oder fehlgeleiteter Agent unkontrolliert handelt.

Cross-Validation ersetzt jedoch keine strukturellen Sicherheitsmaßnahmen wie Berechtigungsbegrenzung oder Netzwerkrestriktionen. Vielmehr ergänzt sie diese um eine zusätzliche Kontrollschicht auf Entscheidungs- und Handlungsebene. Insbesondere in sicherheitskritischen oder regulatorisch sensiblen Einsatzszenarien kann ein solches Vier-Augen-Prinzip auf Systemebene einen erheblichen Beitrag zur Risikominimierung leisten.

Checks and Balances (Prinzip der gegenseitigen Kontrolle)

Das Prinzip der „Checks and Balances“ ist aus der Politikwissenschaft bekannt und beschreibt die gegenseitige Kontrolle unterschiedlicher Machtinstanzen, um Machtkonzentration und Missbrauch zu verhindern. Übertragen auf KI-Agentenarchitekturen bedeutet dies, dass keine einzelne Instanz alleinige Entscheidungs- oder Ausführungsgewalt besitzt. Stattdessen werden Zuständigkeiten, Kontrollmechanismen und Eskalationspfade bewusst verteilt.

In einer solchen Architektur übernehmen mehrere Agenten unterschiedliche Rollen, beispielsweise Entscheidungsfindung, Validierung, Policy-Prüfung oder Ausführung. Ein Agent kann eine Handlung vorschlagen, ein anderer bewertet sie anhand definierter Sicherheits- oder Compliance-Kriterien, während eine separate Instanz die tatsächliche Ausführung kontrolliert. Dadurch entsteht eine strukturelle Gewaltenteilung innerhalb des Systems.

Ziel ist die Vermeidung von Machtkonzentration auf Ebene eines einzelnen Modells oder Prozesses. Selbst wenn eine Instanz durch Prompt Injection, Fehlkonfiguration oder Modellverhalten zu einer problematischen Entscheidung gelangt, kann eine andere Instanz korrigierend eingreifen oder die Aktion blockieren. Die gegenseitige Überwachung reduziert systemische Risiken und erhöht die Resilienz gegenüber Manipulation und Fehlverhalten.

Ein solches Architekturprinzip erhöht allerdings die Komplexität und erfordert klar definierte Rollen, Zuständigkeiten und Entscheidungsregeln. Es ersetzt nicht grundlegende Sicherheitsmaßnahmen wie Berechtigungsbeschränkungen oder Netzwerksegmentierung, sondern ergänzt diese um eine strukturelle Kontrollschicht. Insbesondere in regulierten oder sicherheitskritischen Umgebungen kann das Prinzip der gegenseitigen Kontrolle einen wesentlichen Beitrag zur Reduzierung operativer Risiken leisten.

Governance-Integration

Technische Schutzmaßnahmen allein gewährleisten keinen belastbaren und nachhaltigen Betrieb von KI-Agenten. Ebenso erforderlich ist die systematische Einbettung in bestehende Governance- und Compliance-Strukturen. Nur durch organisatorische Verankerung lassen

sich Verantwortlichkeiten klären, Risiken steuern und regulatorische Anforderungen nachvollziehbar erfüllen.

Der Agent sollte als reguläres informationsverarbeitendes System in das bestehende ISMS integriert werden. Hierzu gehört eine dokumentierte Risikoanalyse, die sowohl technische als auch prozessuale und regulatorische Aspekte berücksichtigt. Änderungen an Modellversionen, Integrationen, Berechtigungen oder Betriebsumgebungen sind in bestehende Change-Management-Prozesse einzubinden und freigabepflichtig zu gestalten.

Ergänzend sind regelmäßige Sicherheitsüberprüfungen vorzusehen, um Konfiguration, Berechtigungsmodelle und Integrationen fortlaufend zu evaluieren. Auch die Schulung verantwortlicher Mitarbeitender ist ein zentraler Bestandteil der Governance-Integration. Personen mit administrativen oder fachlichen Steuerungsaufgaben müssen die spezifischen Risiken und Besonderheiten von KI-Agenten verstehen, um fundierte Entscheidungen treffen zu können.

KI-Agenten sind nicht als experimentelle Nebenlösung zu betrachten, sondern als reguläre Bestandteile der IT-Landschaft. Entsprechend sind sie in bestehende Steuerungs-, Kontroll- und Compliance-Mechanismen einzubetten und dauerhaft organisatorisch zu verankern.

Zusammenfassung

Secure-by-Design im Kontext von KI-Agenten bedeutet, Autonomie technisch zu begrenzen und kontrollierbar zu gestalten. Ziel ist nicht die vollständige Verhinderung selbstständiger Entscheidungen, sondern deren Einbettung in klar definierte und überprüfbare Rahmenbedingungen.

Statt einer unkontrollierten Systemintegration ist eine konsequente Segmentierung der Architektur erforderlich. KI-Agenten dürfen nicht als uneingeschränkt vernetzte Orchestrierungskomponenten betrieben werden, sondern müssen in abgegrenzten, überwachten Umgebungen arbeiten. Ebenso ist eine strukturierte Governance unverzichtbar. Der dauerhafte Betrieb im experimentellen Modus ohne formale Einbindung in bestehende Prozesse ist mit sicherheitskritischen Anforderungen nicht vereinbar.

Transparenz ist dabei ein zentrales Prinzip. Entscheidungsprozesse, Tool-Interaktionen und Datenflüsse müssen nachvollziehbar dokumentiert werden, um Sicherheit, Compliance und Prüfbarkeit sicherzustellen. Ein Black-Box-Betrieb ohne Logging, Monitoring und klare Verantwortlichkeiten ist im Unternehmenskontext nicht vertretbar.

Diese Architekturprinzipien bilden die Grundlage für einen regulatorisch anschlussfähigen und technisch verantwortbaren Einsatz von KI-Agenten in Unternehmensumgebungen. Sie schaffen die strukturellen Voraussetzungen, um Innovation mit kontrollierter Risikosteuerung zu verbinden.

7. Vulnerability- und Schwachstellenmanagement

Der sichere Betrieb von KI-Agenten endet nicht mit der initialen Architekturabsicherung. Wie bei jeder produktiv eingesetzten Software ist ein kontinuierliches und strukturiertes Schwachstellenmanagement erforderlich. Sicherheitsmaßnahmen müssen über den gesamten Lebenszyklus hinweg wirksam bleiben und regelmäßig überprüft werden.

KI-Agenten weisen dabei zusätzliche Besonderheiten auf, die das Risikoprofil erweitern. Sie sind häufig von externen Sprachmodellen abhängig, nutzen Drittbibliotheken und Frameworks, integrieren sich in externe Systeme und können durch Plugins oder Skills funktional erweitert werden. Jede dieser Ebenen kann potenzielle Schwachstellen einführen – sei es durch fehlerhafte Implementierungen, unsichere Konfigurationen oder kompromittierte Abhängigkeiten.

Schwachstellen können somit nicht nur im eigenen Anwendungscode entstehen, sondern auch im Modellverhalten, in verwendeten Softwarekomponenten, in Integrationen oder in Erweiterungsmodulen. Ein wirksames Schwachstellenmanagement muss diese Mehrschichtigkeit berücksichtigen und entsprechende Prüf- und Überwachungsmechanismen etablieren.

Erforderlich sind regelmäßige Sicherheitsprüfungen, strukturierte Update- und Patchprozesse, eine kontinuierliche Beobachtung sicherheitsrelevanter Veröffentlichungen sowie klare Verantwortlichkeiten für Bewertung und Behebung identifizierter Risiken. Nur durch einen systematischen und dokumentierten Prozess lässt sich der langfristig sichere Betrieb eines KI-Agenten gewährleisten.

Systematische Identifikation von Schwachstellen

Ein wirksames Schwachstellenmanagement setzt eine strukturierte und wiederkehrende Identifikation potenzieller Sicherheitslücken voraus. Dabei ist ein ganzheitlicher Ansatz erforderlich, der nicht nur den Anwendungscode, sondern die gesamte Betriebs- und Integrationsumgebung umfasst.

Zu den zentralen Maßnahmen zählen regelmäßige technische Schwachstellenscans der Betriebsumgebung, die systematische Überprüfung eingesetzter Bibliotheken und Abhängigkeiten sowie das kontinuierliche Monitoring sicherheitsrelevanter Veröffentlichungen zu verwendeten Frameworks, Integrationen und Plattformkomponenten. Ebenso ist eine gezielte Bewertung von Konfigurationsfehlern notwendig, da Fehlkonfigurationen häufig eine ebenso große Risikowirkung entfalten wie programmatische Schwachstellen.

Für eine belastbare Bewertung ist zwischen mehreren Ebenen zu unterscheiden. Auf Infrastrukturebene sind Betriebssystem, Container-Technologie, Virtualisierungsschicht und gegebenenfalls Cloud-Umgebung zu betrachten. Auf Anwendungsebene sind das eingesetzte Agenten-Framework, Konfigurationen sowie Erweiterungen oder Plugins zu analysieren. Hinzu kommen die Integrationsschnittstellen zu internen und externen Systemen sowie die zugrunde liegenden Zugriffs- und Berechtigungskonzepte.

Erst die konsolidierte Betrachtung dieser Ebenen ermöglicht eine realistische Einschätzung des Gesamtrisikos. Schwachstellenmanagement im Kontext von KI-Agenten ist damit nicht punktuell, sondern systemisch zu verstehen.

CVE-Tracking und Abhängigkeitsmanagement

KI-Agenten basieren häufig auf komplexen Open-Source-Stacks, die aus zahlreichen Bibliotheken, Frameworks und Laufzeitkomponenten bestehen. Diese Abhängigkeiten erweitern die Angriffsfläche und machen ein strukturiertes Abhängigkeitsmanagement erforderlich.

Zentral ist zunächst die vollständige Transparenz über alle verwendeten Komponenten einschließlich Versionen und transitativer Abhängigkeiten. Nur auf dieser Grundlage lassen sich bekannte Schwachstellen systematisch identifizieren. Ein kontinuierliches Monitoring öffentlich gemeldeter Sicherheitslücken, insbesondere anhand von CVE-Datenbanken, ist daher unerlässlich.

Sicherheitsmeldungen müssen dokumentiert bewertet werden. Dabei ist zu prüfen, ob und in welchem Umfang die eigene Systemarchitektur tatsächlich betroffen ist. Kritische Findings sind priorisiert zu behandeln und zeitnah durch Updates, Konfigurationsanpassungen oder kompensierende Maßnahmen zu adressieren. Die Bewertung und Behebung sollte nachvollziehbar dokumentiert werden, um regulatorischen und auditbezogenen Anforderungen zu genügen.

Ein aktuelles Komponentenverzeichnis – beispielsweise im Sinne einer Software Bill of Materials (SBOM) – erleichtert die Nachverfolgbarkeit von Abhängigkeiten und unterstützt eine schnelle Reaktionsfähigkeit im Falle neu veröffentlichter Schwachstellen. Im Kontext regulatorischer Anforderungen gewinnt eine strukturierte SBOM-Dokumentation zunehmend an Bedeutung und sollte integraler Bestandteil des Schwachstellenmanagements sein.

Schwachstellenmanagement im regulatorischen Kontext

Ein strukturiertes Schwachstellenmanagement ist nicht nur eine technische Best Practice, sondern in vielen regulatorischen Rahmenwerken ausdrücklich oder implizit gefordert. Anforderungen aus ISO/IEC 27001, NIS2, dem Cyber Resilience Act, branchenspezifischen Sicherheitsgesetzen oder KRITIS-Vorgaben setzen voraus, dass bekannte Schwachstellen systematisch identifiziert, bewertet und angemessen behandelt werden.

Im regulatorischen Kontext sind insbesondere folgende Aspekte relevant:

- **Nachvollziehbare Prozesse:** Die Identifikation, Bewertung und Behebung von Schwachstellen muss in dokumentierten Verfahren geregelt sein. Zuständigkeiten, Entscheidungswege und Eskalationsmechanismen sind klar zu definieren.
- **Risikobasierte Priorisierung:** Nicht jede Schwachstelle erfordert dieselbe Reaktionsgeschwindigkeit. Maßgeblich ist eine strukturierte Bewertung hinsichtlich Eintrittswahrscheinlichkeit, Ausnutzbarkeit und potenzieller Auswirkung auf Verfügbarkeit, Integrität und Vertraulichkeit.
- **Fristen und Patch-Management:** Kritische Schwachstellen sind innerhalb definierter Zeiträume zu adressieren. Patch- und Updateprozesse müssen kontrolliert, getestet und dokumentiert erfolgen.

- **Nachweisfähigkeit:** Maßnahmen zur Schwachstellenbehandlung müssen prüffähig dokumentiert sein, um gegenüber interner Revision, Zertifizierungsstellen oder Aufsichtsbehörden belegt werden zu können.

Im Kontext von KI-Agenten erweitert sich der regulatorische Betrachtungsrahmen um zusätzliche Ebenen. Neben klassischen Softwarekomponenten sind auch Modellabhängigkeiten, Integrationen, Plugins sowie Betriebsumgebungen einzubeziehen. Insbesondere bei produktnahen oder sicherheitskritischen Einsätzen kann eine unzureichende Schwachstellenbehandlung zu haftungs- oder aufsichtsrechtlichen Konsequenzen führen.

Ein regulatorisch belastbares Schwachstellenmanagement zeichnet sich daher durch Kontinuität, Dokumentation und Integration in das übergeordnete Risikomanagement aus. Es ist kein punktueller Prozess, sondern ein dauerhaft etablierter Bestandteil der Sicherheitsgovernance.

Sicherheitsvalidierung und unabhängige Prüfungen

Neben automatisierten Schwachstellenscans sind strukturierte Sicherheitsprüfungen ein wesentlicher Bestandteil eines belastbaren Sicherheitskonzepts. Automatisierte Verfahren identifizieren bekannte technische Schwachstellen, erfassen jedoch nicht zwangsläufig architekturelle Risiken, Fehlkonfigurationen oder modellbasierte Angriffsszenarien.

Empfehlenswert sind daher regelmäßige Architektur-Reviews, bei denen Integrationslogik, Berechtigungsmodelle und Netzwerksegmente systematisch analysiert werden. Ergänzend sollten Konfigurationsprüfungen durchgeführt werden, um sicherzustellen, dass definierte Sicherheitsvorgaben tatsächlich umgesetzt sind. Die Simulation typischer Angriffsszenarien – etwa in Form strukturierter Penetrationstests – ermöglicht eine praxisnahe Bewertung der Resilienz gegenüber realistischen Bedrohungen.

Im Kontext von KI-Agenten gewinnen darüber hinaus LLM-spezifische Testmethoden an Bedeutung. Hierzu zählen beispielsweise strukturierte Prompt-Injection-Tests, die Überprüfung von Tool-Call-Kontrollen oder Tests zur Evaluierung von Datenabflussrisiken. Solche Prüfungen adressieren die semantische und modellbasierte Angriffsebene, die durch klassische Tests nicht vollständig abgedeckt wird.

Die Durchführung technischer Prüfungen kann – abhängig von Größe, Kompetenzprofil und Ressourcen der Organisation – durch spezialisierte externe Partner erfolgen. Entscheidend ist jedoch, dass die Ergebnisse fachlich gesteuert, risikoorientiert bewertet und regulatorisch eingeordnet werden. Sicherheitsvalidierung ist kein isoliertes Audit-Ereignis, sondern Teil eines kontinuierlichen Verbesserungsprozesses innerhalb der Governance-Struktur.

Besonderheiten bei LLM- und Agentensystemen

Im Unterschied zu klassischer Software entstehen Risiken bei LLM- und Agentensystemen nicht ausschließlich durch Programmierfehler oder technische Implementierungsschwächen. Sicherheitsrelevante Probleme können ebenso aus dem Modellverhalten, aus manipulierbaren Kontextinformationen, aus der Integrationslogik oder aus fehlkonfigurierten Tool-Freigaben resultieren. Die Angriffsfläche ist damit nicht nur technisch, sondern auch semantisch geprägt.

Das Schwachstellenmanagement muss dieser Besonderheit Rechnung tragen. Neben klassischen Code- und Infrastrukturprüfungen sind spezifische Testfälle für Prompt Injection vorzusehen. Ebenso sollten indirekte Manipulationsszenarien überprüft werden, bei denen externe Inhalte automatisiert in den Agentenkontext einfließen. Die Validierung von Tool-Call-Beschränkungen ist ein weiterer zentraler Prüfpunkt, um sicherzustellen, dass modellbasierte Entscheidungen nicht zu unkontrollierten Systemaktionen führen. Ergänzend ist eine regelmäßige Überprüfung der Berechtigungsstruktur erforderlich, um überprivilegierte Identitäten oder unklare Rechtezuweisungen zu identifizieren.

Die Sicherheitsbewertung von KI-Agenten muss somit zweigleisig erfolgen. Einerseits sind klassische technische Angriffsszenarien zu adressieren, andererseits sind semantische und modellbasierte Manipulationsmöglichkeiten systematisch zu testen. Nur die Kombination beider Perspektiven ermöglicht eine realistische Einschätzung des tatsächlichen Risikoprofils.

Kontinuierliche Verbesserung

Ein wirksames Vulnerability Management ist kein einmaliges Projekt, sondern ein fortlaufender Prozess. Die Bedrohungslage, technologische Abhängigkeiten und regulatorischen Rahmenbedingungen entwickeln sich kontinuierlich weiter. Entsprechend muss auch die Sicherheitsbewertung von KI-Agenten regelmäßig überprüft und angepasst werden.

Empfehlenswert sind strukturierte Re-Assessments in definierten Intervallen oder bei wesentlichen Änderungen der Systemarchitektur. Erkenntnisse aus Sicherheitsvorfällen, Tests oder externen Prüfungen sollten systematisch als dokumentierte „Lessons Learned“ erfasst und in bestehende Sicherheitsmaßnahmen integriert werden. Ebenso ist eine enge Verzahnung mit dem Change-Management erforderlich, damit Änderungen an Modellversionen, Integrationen oder Berechtigungen unmittelbar sicherheitsseitig bewertet werden.

Darüber hinaus sollten neue regulatorische Anforderungen oder branchenspezifische Vorgaben frühzeitig berücksichtigt und in bestehende Prozesse integriert werden. Die Sicherheit eines KI-Agenten ist kein statischer Zustand, sondern entwickelt sich mit dessen Einsatzszenario, Funktionsumfang und Integrationsgrad.

Eine kontinuierliche Verbesserung stellt sicher, dass Schutzmaßnahmen nicht nur formal bestehen, sondern wirksam bleiben. Sie ist damit ein zentraler Bestandteil einer nachhaltigen und regulatorisch belastbaren Sicherheitsstrategie für KI-Agenten.

8. Reifegradmodell für KI-Agent Security

Der sichere Einsatz von KI-Agenten ist kein binärer Zustand im Sinne von „sicher“ oder „unsicher“. Vielmehr entwickelt sich das Sicherheitsniveau schrittweise entlang technischer, organisatorischer und regulatorischer Reifegrade. Unternehmen durchlaufen typischerweise mehrere Entwicklungsstufen – von experimentellen Pilotprojekten bis hin zu vollständig integrierten, governance-gestützten Systemlandschaften.

Ein strukturiertes Reifegradmodell ermöglicht es, den aktuellen Stand der Sicherheitsmaßnahmen systematisch einzuordnen. Es schafft Transparenz über bestehende Defizite, unterstützt Priorisierungsentscheidungen und dient als Grundlage für strategische Weiterentwicklungsmaßnahmen. Dabei werden nicht nur technische Schutzmechanismen berücksichtigt, sondern auch Governance-Strukturen, regulatorische Einbindung, Dokumentationsgrad und organisatorische Verantwortlichkeiten.

Das nachfolgend dargestellte Reifegradmodell beschreibt typische Entwicklungsstufen für KI-Agent Security im Unternehmenskontext. Es ist als Orientierungsrahmen zu verstehen und kann – abhängig von Branche, Unternehmensgröße und regulatorischem Umfeld – angepasst oder erweitert werden.

Level 1 – Experimentell

Auf dieser Reifestufe werden KI-Agenten primär testweise oder explorativ eingesetzt. Der Fokus liegt auf Innovations- und Pilotprojekten, häufig initiiert durch Fachbereiche oder Entwicklungsteams. Eine formale Einbindung in bestehende Governance-, Sicherheits- oder Compliance-Strukturen ist in der Regel noch nicht erfolgt.

Typische Merkmale dieser Stufe sind der Einsatz in isolierten Proof-of-Concept-Szenarien, das Fehlen einer dokumentierten Risikoanalyse sowie die Nutzung überprivilegierter Service-Accounts aus pragmatischen Gründen. Protokollierungs- und Monitoring-Mechanismen sind meist nicht formalisiert, und externe Tools oder Integrationen werden ohne strukturierte Freigabeprozesse eingebunden.

Die Risikocharakteristik ist durch eine hohe Flexibilität bei gleichzeitig geringer Kontrolltiefe geprägt. Technische Experimente stehen im Vordergrund, während Sicherheits- und Governance-Aspekte nachgelagert behandelt werden. Für kurzfristige Innovationsphasen kann dieser Zustand tolerierbar sein; ein produktiver oder geschäftskritischer Einsatz ist auf dieser Reifestufe jedoch nicht verantwortbar.

Level 2 – Kontrolliert

Auf dieser Reifestufe wird der KI-Agent bewusst und strukturiert betrieben. Erste technische Sicherheitsmaßnahmen sind implementiert, und der Einsatz erfolgt nicht mehr ausschließlich experimentell. Dennoch ist die Integration in übergeordnete Governance- und Compliance-Strukturen noch unvollständig.

Typische Merkmale sind grundlegende Zugriffsbeschränkungen im Sinne eines eingeschränkten Berechtigungsmodells sowie der Betrieb in einer segmentierten Test- oder Vorproduktionsumgebung. Kritische Aktionen werden teilweise manuell überprüft oder

freigegeben. Die Systemarchitektur ist in Grundzügen dokumentiert, und ein Basis-Logging zur Nachvollziehbarkeit von Eingaben und Aktionen ist vorhanden.

Die Risikocharakteristik ist durch eine teilweise Kontrolle bei noch begrenzter organisatorischer Einbettung geprägt. Technische Schutzmaßnahmen existieren, sind jedoch nicht durchgängig standardisiert oder regulatorisch eingebettet. Für interne, nicht sicherheitskritische Anwendungsfälle kann dieses Niveau ausreichend sein; für geschäftskritische oder regulierte Einsatzszenarien ist eine höhere Reife erforderlich.

Level 3 – Segmentiert

Auf dieser Reifestufe ist der KI-Agent technisch klar isoliert und strukturiert abgesichert. Der Betrieb erfolgt in einer definierten, kontrollierten Umgebung, und zentrale Sicherheitsprinzipien sind konsequent umgesetzt. Die Architektur folgt erkennbar einem Secure-by-Design-Ansatz.

Typische Merkmale sind eine containerisierte oder virtualisierte Betriebsumgebung mit klarer Netzwerksegmentierung sowie ein strikt umgesetztes Identity- und Berechtigungskonzept nach dem Least-Privilege-Prinzip. Tool-Integrationen sind durch Whitelisting begrenzt, und ausgehende Netzwerkverbindungen unterliegen einer definierten Egress-Kontrolle. Darüber hinaus besteht ein strukturierter Prozess zur Identifikation und Bewertung technischer Schwachstellen.

Die Risikocharakteristik ist technisch solide. Die wesentlichen operativen Risiken sind begrenzt und kontrollierbar. Eine vollständige Einbindung in regulatorische, auditierbare Governance-Strukturen ist jedoch noch nicht zwingend umgesetzt. Für viele produktive Einsatzszenarien stellt diese Stufe eine belastbare Grundlage dar; in regulierten oder kritischen Umfeldern ist jedoch eine weitergehende Integration erforderlich.

Level 4 – Governance-integriert

Auf dieser Reifestufe ist der KI-Agent nicht nur technisch abgesichert, sondern vollständig in die bestehenden Sicherheits- und Risikomanagementprozesse der Organisation integriert. Der Betrieb erfolgt auf Basis klar definierter Governance-Strukturen, und sicherheitsrelevante Aspekte sind dokumentiert, überprüfbar und organisatorisch verankert.

Typische Merkmale sind eine dokumentierte Risikoanalyse im Rahmen des ISMS, regelmäßig durchgeführte Sicherheitsüberprüfungen sowie die Einbindung in bestehende Incident-Management-Prozesse. Logging- und Monitoring-Strukturen sind nachvollziehbar implementiert und in zentrale Überwachungssysteme integriert. Verantwortlichkeiten für Betrieb, Sicherheit, Freigaben und Änderungen sind eindeutig definiert und organisatorisch zugewiesen.

Die Risikocharakteristik ist durch eine strukturierte Steuerung und hohe Transparenz geprägt. Der Agentenbetrieb ist auditfähig vorbereitet und kann gegenüber interner Revision oder externen Prüfern nachvollziehbar dargestellt werden. Für regulierte Branchen oder geschäftskritische Anwendungen bildet diese Reifestufe in der Regel die Mindestanforderung für einen verantwortbaren produktiven Einsatz.

Level 5 – Regulatorisch belastbar

Auf dieser höchsten Reifestufe ist der Einsatz des KI-Agenten nicht nur technisch und organisatorisch abgesichert, sondern vollständig regulatorisch eingebettet und dauerhaft überprüfbar ausgestaltet. Sicherheit wird nicht reaktiv, sondern strategisch und kontinuierlich gesteuert.

Typische Merkmale sind die vollständige Einbettung in einschlägige regulatorische Anforderungen, etwa im Kontext von NIS2, Cyber Resilience Act oder ISO/IEC 27001. Schwachstellen- und Updateprozesse sind dokumentiert, klar definiert und revisionssicher umgesetzt. Nachweise über Risikoanalysen, Schutzmaßnahmen, Tests und Vorfälle können gegenüber internen Kontrollinstanzen ebenso wie gegenüber externen Prüfern oder Aufsichtsbehörden strukturiert erbracht werden.

Governance- und Kontrollmechanismen sind organisatorisch verankert, mit klaren Verantwortlichkeiten, Eskalationswegen und regelmäßigen Management-Reviews. Ergänzend erfolgen unabhängige Sicherheitsvalidierungen in definierten Intervallen, etwa durch externe Prüfungen oder spezialisierte Testverfahren.

Die Risikocharakteristik ist nachhaltig abgesichert. Der Agentenbetrieb ist revisions- und prüffähig ausgestaltet und kann auch in regulierten oder sicherheitskritischen Umgebungen belastbar betrieben werden. Diese Reifestufe steht für einen strategisch integrierten, langfristig verantwortbaren Einsatz von KI-Agenten im Unternehmenskontext.

Einordnung und Weiterentwicklung

Das Reifegradmodell dient nicht der Bewertung oder Klassifizierung einzelner Produkte, sondern der Einordnung des organisatorischen Umgangs mit KI-Agenten im jeweiligen Unternehmenskontext. Entscheidend ist, in welchem Maß Sicherheits- und Governance-Anforderungen technisch umgesetzt, organisatorisch verankert und regulatorisch nachvollziehbar dokumentiert sind.

Zur Selbsteinschätzung eignen sich insbesondere Leitfragen, die zentrale Kontrollpunkte abdecken. Dazu gehört, ob der Agent in das bestehende Risikomanagement integriert ist, ob Berechtigungen strikt nach dem Least-Privilege-Prinzip vergeben und dokumentiert werden, ob ein strukturiertes Schwachstellenmanagement etabliert ist und ob relevante regulatorische Anforderungen bewertet, abgeleitet und nachvollziehbar festgehalten sind. Ebenso zentral ist die Frage, ob Entscheidungswege und operative Aktionen – insbesondere Tool-Calls und Kontextänderungen – so protokolliert werden, dass sie auditierbar und im Incident-Fall rekonstruierbar sind.

Die Weiterentwicklung entlang der Reifegrade sollte schrittweise erfolgen. Maßnahmen müssen klar priorisiert, Verantwortlichkeiten eindeutig zugewiesen und die Wirksamkeit der umgesetzten Kontrollen überprüfbar gemacht werden. Das Ziel ist ein kontrollierter Ausbau von Funktionalität und Integrationsgrad, ohne dass die Sicherheits- und Governance-Fähigkeit der Organisation hinter der technischen Entwicklung zurückbleibt.

9. Fazit und Handlungsempfehlung

KI-Agenten wie OpenClaw eröffnen Unternehmen erhebliche Potenziale zur Automatisierung, Effizienzsteigerung und Unterstützung komplexer Geschäftsprozesse. Sie können Informationsflüsse bündeln, Entscheidungsprozesse beschleunigen und operative Abläufe strukturieren. Gleichzeitig verändern sie das sicherheitstechnische Risikoprofil der IT-Landschaft grundlegend.

Im Unterschied zu klassischer Software verbinden KI-Agenten semantische Kontextverarbeitung mit probabilistischer Entscheidungslogik, dynamischer Tool-Integration und unmittelbarem Zugriff auf interne wie externe Systeme. Sie fungieren als Integrations- und Orchestrierungsschicht zwischen Datenquellen, Anwendungen und operativen Ressourcen. Dadurch entstehen neuartige Angriffspunkte, insbesondere im Bereich Prompt Injection, Tool-Manipulation, Integrationsrisiken sowie unkontrollierter Datenaggregation.

Die Einführung eines KI-Agenten ist daher nicht als isoliertes Innovationsprojekt zu verstehen, sondern als sicherheitsrelevante Systemintegration. Unternehmen sollten den Einsatz risikoorientiert bewerten, frühzeitig in bestehende Governance- und Risikomanagementstrukturen einbinden und eine Secure-by-Design-Architektur implementieren. Technische Schutzmaßnahmen, strukturierte Schwachstellenprozesse, klare Berechtigungsmodelle sowie nachvollziehbare Logging- und Monitoring-Strukturen sind zentrale Voraussetzungen für einen verantwortbaren Betrieb.

Die Handlungsempfehlung lautet, KI-Agenten schrittweise und kontrolliert einzuführen, den Reifegrad regelmäßig zu überprüfen und Sicherheitsmaßnahmen kontinuierlich weiterzuentwickeln. Innovation und Sicherheit sind dabei nicht als Gegensätze zu verstehen, sondern als komplementäre Anforderungen an eine moderne, regulatorisch anschlussfähige IT-Architektur.

Zentrale Erkenntnisse

Aus technischer und regulatorischer Perspektive lassen sich mehrere zentrale Schlussfolgerungen ableiten.

Erstens sind KI-Agenten keine isolierten Werkzeuge, sondern integrative Systemkomponenten. Sie verbinden Sprachmodelle mit operativen Systemzugriffen, Datenquellen und Automatisierungslogik und wirken damit unmittelbar auf die bestehende IT-Architektur ein.

Zweitens entsteht Sicherheit primär durch eine belastbare Architektur und klare Kontrollmechanismen – nicht durch die bloße Auswahl eines bestimmten Modells oder Frameworks. Die Gestaltung von Integrationen, Berechtigungen, Netzwerkrestriktionen und Überwachungsmechanismen ist entscheidend für das tatsächliche Risikoprofil.

Drittens erfordert technische Autonomie zwingend ergänzende Kontrollinstanzen. Je größer der Entscheidungsspielraum eines Agenten ist, desto wichtiger sind externe Validierungs- und Freigabemechanismen, transparente Protokollierung und begrenzte Berechtigungen.

Viertens ist Schwachstellenmanagement als kontinuierlicher Prozess zu verstehen. Abhängigkeiten, Integrationen und regulatorische Rahmenbedingungen entwickeln sich fortlaufend weiter und erfordern eine regelmäßige Neubewertung.

Fünftens gelten regulatorische Anforderungen unabhängig davon, ob ein System Open Source ist oder im Innovationskontext eingesetzt wird. Maßgeblich ist die konkrete Funktion im Unternehmen und die damit verbundene Risikowirkung.

Der sichere Betrieb von KI-Agenten ist daher weniger eine Frage einzelner Konfigurationsparameter als vielmehr eine Frage strukturierter Governance, klar definierter Verantwortlichkeiten und nachhaltiger organisatorischer Einbettung.

Handlungsempfehlungen für Unternehmen

Für Organisationen, die KI-Agenten produktiv einsetzen oder deren Einsatz planen, ergeben sich klare prioritäre Handlungsfelder. Ziel ist es, Innovation strukturiert zu ermöglichen, ohne sicherheits- oder compliance-relevante Risiken unkontrolliert einzugehen.

1. Architektur vor Funktion priorisieren
Vor der produktiven Nutzung sollte eine strukturierte Architektur- und Risikoanalyse durchgeführt werden. Entscheidungsräume, Integrationen, Datenflüsse und Berechtigungen sind vorab zu definieren. Sicherheitsmechanismen müssen integraler Bestandteil des Systemdesigns sein und dürfen nicht erst nachgelagert implementiert werden.

2. Berechtigungen strikt minimieren
Service-Accounts, API-Tokens und Integrationen sind konsequent nach dem Least-Privilege-Prinzip zu konfigurieren. Jede technische Identität darf ausschließlich über die für den konkreten Anwendungsfall erforderlichen Rechte verfügen. Überprivilegierte Zugriffe sind zu vermeiden und regelmäßig zu überprüfen.

3. Automatisierung kontrollieren
Autonome Entscheidungen des Agenten sollten durch technische Validierungsmechanismen abgesichert werden. Kritische oder irreversible Aktionen sind durch Policy-Logik, Freigabeprozesse oder Human-in-the-Loop-Verfahren zu kontrollieren. Das Sprachmodell darf nicht die alleinige Entscheidungsinstanz für operative Eingriffe sein.

4. Schwachstellenmanagement etablieren
Ein kontinuierlicher Prozess zur Identifikation, Bewertung und Behebung von Schwachstellen ist aufzubauen. Komponenten, Integrationen und Konfigurationen sind regelmäßig zu prüfen und sicherheitsrelevante Erkenntnisse strukturiert zu dokumentieren. Abhängigkeiten und Updates müssen systematisch überwacht werden.

5. Regulatorische Einordnung dokumentieren
Die regulatorische Bewertung im Kontext einschlägiger Rahmenwerke – etwa NIS2, Cyber Resilience Act, AI Act, ISO/IEC 27001 oder sektoraler Vorgaben – sollte nachvollziehbar dokumentiert werden. Maßgeblich ist die konkrete Funktion des Agenten im Unternehmen sowie dessen Einfluss auf geschäftskritische Prozesse.

Zusammenfassend empfiehlt sich ein schrittweises, risikoorientiertes Vorgehen mit klar definierten Verantwortlichkeiten und überprüfbaren Maßnahmen. Der produktive Einsatz von KI-Agenten erfordert nicht nur technisches Know-how, sondern eine konsistente Verbindung von Architektur, Governance und regulatorischer Sorgfalt.

Strategische Perspektive

KI-Agenten werden in den kommenden Jahren zunehmend in produktive Geschäftsprozesse integriert werden. Parallel dazu ist mit einer weiteren Verdichtung regulatorischer Anforderungen sowie einer stärkeren aufsichtsrechtlichen Fokussierung auf KI-gestützte Systeme zu rechnen. Unternehmen stehen daher vor der Herausforderung, technologische Innovation mit struktureller Absicherung zu verbinden.

Organisationen, die frühzeitig eine belastbare Sicherheitsarchitektur implementieren, klare Governance-Strukturen etablieren sowie Nachvollziehbarkeit und Dokumentation systematisch verankern, schaffen eine tragfähige Grundlage für den nachhaltigen Einsatz von KI-Agenten. Diese Vorarbeit reduziert spätere Anpassungsaufwände, erleichtert regulatorische Prüfungen und erhöht die interne Steuerungsfähigkeit.

Sicherheit ist in diesem Kontext kein Innovationshemmnis, sondern eine strategische Voraussetzung für Skalierbarkeit und Vertrauenswürdigkeit. Nur wenn Entscheidungslogik, Integrationen und Datenflüsse kontrolliert und transparent gestaltet sind, lassen sich KI-Agenten verantwortungsvoll in geschäftskritische Prozesse integrieren.

Eine vorausschauende Sicherheits- und Governance-Strategie ermöglicht es Unternehmen, technologische Potenziale zu nutzen und gleichzeitig regulatorische Anforderungen sowie gesellschaftliche Erwartungen an verantwortungsvolle KI-Nutzung zu erfüllen.

Abschließende Einordnung

OpenClaw dient in diesem Leitfaden als praxisnahes Referenzbeispiel für eine gesamte Systemklasse: autonome KI-Agenten im Unternehmenskontext. Die dargestellten architektonischen, sicherheitstechnischen und regulatorischen Prinzipien sind jedoch nicht auf ein spezifisches Tool beschränkt. Sie sind grundsätzlich auf vergleichbare agentenbasierte Systeme übertragbar.

Entscheidend für das Risikoprofil ist nicht die Wahl eines bestimmten Agenten, sondern die Qualität seiner architekturellen Einbettung, die Ausgestaltung von Integrationen und Berechtigungen sowie die organisatorische Steuerung im Rahmen bestehender Governance-Strukturen. Sicherheit entsteht durch kontrollierte Systemintegration und nachvollziehbare Verantwortlichkeiten – nicht durch die bloße Auswahl einer Technologie.

Der sichere Einsatz von KI-Agenten ist daher kein singuläres Einführungsprojekt, sondern ein fortlaufender Entwicklungsprozess. Technische Maßnahmen, organisatorische Anpassungen und regulatorische Bewertungen müssen kontinuierlich überprüft und weiterentwickelt werden. Nur durch diese dauerhafte Steuerung lässt sich ein verantwortbarer und nachhaltiger Betrieb in produktiven Unternehmensumgebungen gewährleisten.

10. Über uns

Blackfort Technology ist spezialisiert auf Sicherheitsarchitektur, Security Governance und regulatorische Einordnung u.a. von AI-Agenten und KI-Systemen im Unternehmenskontext.

Der Fokus liegt auf der strukturellen Absicherung autonomer Systeme in regulierten und sicherheitskritischen Umgebungen – insbesondere dort, wo AI-Agenten operative Systemzugriffe erhalten.

Das Unternehmen unterstützt Organisationen bei:

- Architektur- und Integrationsbewertungen
- AI-Agent Security Assessments
- Regulatorischer Einordnung (NIS2, AI Act, ISO 27001)
- Entwicklung belastbarer Governance-Strukturen

Der Autor Christian Gebhardt ist Mitglied im Expertenkreis KI-Sicherheit des BSI und berät Unternehmen zur sicheren Integration autonomer AI-Agenten.

Strategischer Austausch

Organisationen, die AI-Agenten produktiv einsetzen oder regulatorisch absichern möchten, können ein strukturiertes Architektur- und Risiko-Assessment anfragen.

www.blackfort-tec.de
info@blackfort-tec.de

11. Haftungsausschluss

Dieses Dokument stellt keine Rechtsberatung dar.
Die dargestellten Inhalte dienen der fachlichen Einordnung sicherheitsrelevanter Aspekte beim Einsatz von AI-Agenten im Unternehmenskontext.

Impressum

Angaben gemäß § 5 TMG

Blackfort Technology Unternehmergesellschaft (haftungsbeschränkt)

Vertreten durch: Christian Gebhardt, Geschäftsführer

Telefon: +49 (0) 228 299780 61

E-Mail: info@blackfort-tec.de
Web: www.blackfort-tec.de

Registergericht: Bonn

Registernummer: HRB 23120

Verantwortlich für den Inhalt nach § 55 Abs. 2 RStV: Christian Gebhardt