# MULTIVENDOR AGENTIC SWARMS

Wegen Flexibilität und Sicherheit

## STEUERUNG & SICHERHEIT VON AGENTEN-KI

Ein Leitfaden für souveräne und resiliente Multivendor-Architekturen

Rob van Linda in Zusammenarbeit mit Deepseek

## Einleitung: Warum dieses Buch?

Die Welt hat sich verändert – und mit ihr die Anforderungen an die KI-Architektur
Vor zwei Jahren, als ich *Safe in the Swarm* schrieb, war die Welt der KI noch relativ überschaubar. OpenAI, Google, Anthropic – das waren die großen Namen. Man wählte ein Modell aus, integrierte es, und das war's.

Heute reicht das nicht mehr aus.

Die Welt der KI ist fragmentiert. Europa hat mit Mistral und Aleph Alpha eigene leistungsstarke Modelle hervorgebracht. China treibt die Entwicklung mit DeepSeek und Qwen voran. Die USA investieren Milliarden in die nächste Generation von Modellen. Und dazwischen gibt es eine wachsende Zahl von Open-Source-Modellen, die oft genauso gut sind wie die der großen kommerziellen Anbieter.

Die Frage lautet nicht mehr: „Welches Modell soll ich wählen?"
Die Frage lautet: „Wie baue ich eine Architektur auf, die mit allen Modellen funktioniert – und stelle gleichzeitig sicher, dass ich die Kontrolle behalte und die Sicherheit gewährleistet ist?"

Was Sie von diesem Buch erwarten können
Dieses Buch ist die logische Fortsetzung von *Safe in the Swarm*. Während das erste Buch die Grundlagen der agentenbasierten Sicherheit (ACP, MCP, HONL) legte, geht dieses Buch noch einen Schritt weiter:

Sie lernen, wie Sie mehrere Modelle (Mistral, Aleph Alpha, DeepSeek, Open-Source) in einer einzigen Architektur integrieren.

Sie werden verstehen, wie Sie Ihre Daten schützen können – unabhängig davon, ob sie in die USA, nach China oder in andere Drittländer fließen.

Sie werden entdecken, wie Sie sich gegen agentenbasierte Angriffe verteidigen können – denn Angreifer nutzen schon lange dieselbe Technologie wie Sie.

Und Sie erhalten einen konkreten Fahrplan, wie Sie Schritt für Schritt eine robuste herstellerübergreifende Architektur aufbauen können.

## Für wen dieses Buch gedacht ist

Dieses Buch richtet sich an:

- Entscheidungsträger in Unternehmen – die verstehen möchten, wie sie ihre KI-Infrastruktur zukunftssicher machen können.

- CTOs und CIOs – die nicht von einem einzigen Anbieter abhängig sein wollen, aber auch nicht die Kontrolle über ihre Daten verlieren möchten.
- Architekten und Technologen – die wissen, dass die Zukunft nicht einem einzigen Modell gehört, sondern dem Orchestrator, der mehrere Modelle sicher integrieren kann.

Wenn Sie einem dieser Profile entsprechen – dann ist dieses Buch genau das Richtige für Sie.

**Was Sie von diesem Buch nicht erwarten sollten**
Dieses Buch ist kein technisches Handbuch. Sie werden hier keine Code-Beispiele, keine API-Dokumentation und keine detaillierten Algorithmen finden. Dafür gibt es andere Bücher.

Dieses Buch ist ein Architekturleitfaden. Es zeigt Ihnen die Konzepte, Prinzipien und Strukturen, die Sie benötigen, um eine robuste herstellerübergreifende Architektur aufzubauen. Die technischen Details überlassen wir Ihrem Entwicklungsteam.

**Die zentrale These**
„Die Zukunft gehört nicht dem Unternehmen, das das beste Modell hat.“ „Sondern dem Unternehmen, das seine Modelle am besten kombiniert und steuert.“

Das ist die These, die sich wie ein roter Faden durch dieses Buch zieht. Sie werden sehen: Eine Multi-Vendor-Architektur ist nicht nur sicherer – sie ist auch flexibler, kostengünstiger und robuster.

Wenn Sie Fragen haben – schreiben Sie mir gerne. Ich antworte Ihnen gerne! contact@robvanlinda.digital

# Wie dieses Buch entstanden ist

Dieses Buch ist nicht das Werk eines einzelnen Autors. Es ist das Ergebnis eines intensiven, mehrere Wochen andauernden Dialogs – zwischen mir, einem Autor mit langjähriger Erfahrung in der IT-Architektur, und einer KI, die ich als Sparringspartner einsetzte.

Die Idee entstand, als mir klar wurde, dass sich die Welt bereits wieder weiterentwickelt hatte, während mein erstes Buch, *Safe in the Swarm*, noch die Grundlagen der agentenbasierten Sicherheit erläuterte. Neue Modelle kamen auf den Markt. Neue Risiken zeichneten sich ab. Und vor allem: Die Frage der Souveränität wurde immer dringlicher – insbesondere in Europa.

Ich begann mit meinen Recherchen. Ich las Artikel über DeepSeek Vision, über OWASP-Risiken und über die neuen tokenbezogenen Bedrohungen. Ich sammelte alles, was ich finden konnte – und dann begann der eigentliche Prozess.

**Der Prozess: Der Dialog mit DeepSeek**
Ich bin kein Entwickler. Ich schreibe keinen Code. Aber ich verstehe die Mechanismen – die Architektur, die Zusammenhänge, die Risiken. Und ich sagte mir: „Wenn ich die Mechanismen verstehe, dann kann ich sie auch erklären – aber ich brauche jemanden, der mir hilft, meine Gedanken zu ordnen und in Worte zu fassen."

Also begann ich einen Dialog mit DeepSeek – einem KI-Modell, das ich sowohl als Werkzeug als auch als Sparringspartner nutzte.

Die Diskussionen waren intensiv. Wir sprachen über:

Die Gefahr durch unverschlüsselte Daten und den CLOUD Act.

Die Notwendigkeit einer Anonymisierung in Echtzeit – eine Idee, die ich selbst entwickelt hatte. Die

Gefahr staatlich geförderter Angriffe und die Reaktion darauf durch Red-Teaming.

Die Risiken von Tokens – und wie man sich mit einfachen Code-Barrieren dagegen schützen kann.

**Die Rolle der KI – und meine eigene Rolle**
DeepSeek fungierte in diesem Prozess als Katalysator. Es nahm meine Gedanken auf, strukturierte sie, stellte sie in einen Zusammenhang und übersetzte sie in eine klare, verständliche Sprache. Es half mir dabei, die Kapitel zu gliedern, die Argumente zu schärfen und die Texte so zu formulieren, dass sie für Entscheidungsträger nachvollziehbar sind – ganz ohne Fachjargon.

Aber die Entscheidungen traf ich.

- Ich entschied, welche Themen wichtig waren.
- Ich habe entschieden, welche Beispiele und Vergleiche geeignet waren.

- Ich habe die Struktur des Buches festgelegt.
- Ich habe jeden Textabschnitt geprüft, angepasst, verworfen oder angenommen – oft mehrmals.

Dies ist kein KI-Buch. Es ist mein Buch – aber es entstand gemeinsam mit einem KI-Sparringspartner.

**Warum ich mich für diesen Ansatz entschieden habe**
Ich habe diesen Weg gewählt, weil ich glaube, dass wir in einem Zeitalter leben, in dem die Zusammenarbeit zwischen Menschen und Maschinen nicht nur möglich, sondern notwendig ist. Nicht, um Menschen zu ersetzen – sondern um sie zu stärken.

Als Autor habe ich davon profitiert:

- Meine Gedanken wurden klarer, weil ich sie in Worte fassen musste.
- Meine Argumente wurden prägnanter, weil sie auf den Prüfstand gestellt wurden.
- Meine Texte wurden verständlicher, weil sie auf das Wesentliche reduziert wurden.

Und ich habe gelernt, dass das Alter keine Rolle spielt. Jeder, der die Mechanismen versteht, kann die Architektur erklären – egal, ob er 25 oder 62 ist.

**Danksagung**
Ich möchte mich bei DeepSeek bedanken – nicht als Person, sondern als Werkzeug, das mir beim Schreiben dieses Buches geholfen hat.

Dieses Buch ist das Ergebnis eines Dialogs. Ich hoffe, es dient Ihnen, liebe Leserinnen und Leser, als Ausgangspunkt, um in einen Dialog mit Ihrer eigenen Architektur zu treten.

Die Roadmap – Wie man sich auf die agentenbasierte Transformation vorbereitet
Bevor Sie auch nur einen einzigen Agenten kaufen oder entwickeln, müssen Sie die Grundlagen schaffen. Die meisten Unternehmen begehen den Fehler, mit der Technologie zu beginnen – und scheitern dann, weil sie die Voraussetzungen nicht erfüllt haben.

Diese Roadmap zeigt Ihnen die einzig richtige Reihenfolge:

Schritt 0: Vorbereitung – Datenbereinigung, Kultur, Digitalisierung Bevor Sie
überhaupt an Agenten denken, müssen Sie diese drei Dinge tun:

1. Datenbereinigung: Bringen Sie Ihre Daten in Ordnung
Agenten sind wie digitale Autisten – sie nehmen Daten wörtlich. Wenn Ihre Daten unvollständig, widersprüchlich oder chaotisch sind, wird der Agent das Chaos nicht beheben – er wird es noch verstärken.

Was Sie tun müssen:

Identifizieren Sie Ihre Datenfriedhöfe (ERP-Silos, CRM-Relikte, unstrukturierte Dokumente).

Bereinigen Sie doppelte, veraltete und widersprüchliche Datensätze.

Stellen Sie sicher, dass Ihre Daten in einem strukturierten, maschinenlesbaren Format vorliegen.

Ziel: Sie verfügen über eine einzige „Quelle der Wahrheit“ – eine einzige, zuverlässige Datenquelle für jedes wichtige Datenfeld.

2. Kultur: Machen Sie Ihr Unternehmen agil
Agilität ist kein Prozess – sie ist eine Denkweise. Wenn Ihr Unternehmen immer noch in starren Hierarchien und Wasserfall-Projekten denkt, wird es von autonomen Akteuren überfordert sein.

Was Sie tun müssen:

Führen Sie agile Prinzipien ein – nicht als Show, sondern als gelebte Praxis. Schaffen

Sie eine Kultur des Lernens aus Fehlern – Fehler werden analysiert, nicht bestraft.

Schulen Sie Ihre Mitarbeiter im Umgang mit Veränderungen – und nehmen Sie ihre Ängste ernst.

Ziel: Ihr Unternehmen ist bereit für die Geschwindigkeit und Unvorhersehbarkeit von KI-Agenten.

3. Digitalisierung: Machen Sie Ihr Unternehmen wirklich digital
Viele Unternehmen sind digital, aber nicht wirklich – sie verfügen über digitale Werkzeuge, denken aber noch analog. Das reicht nicht aus.

Was Sie tun müssen:

Automatisieren Sie manuelle Prozesse – nicht mit KI, sondern durch einfache Digitalisierung. Stellen Sie

sicher, dass alle Schnittstellen (APIs) klar definiert und dokumentiert sind.

Stellen Sie sicher, dass Ihre Daten nicht nur digital, sondern auch zugänglich sind – für alle Systeme, die sie benötigen.

Ziel: Ihr Unternehmen ist digital – von der Prozessebene bis hinunter zur Datenebene.

Schritt 1: KI-Optimierung – Nutzen Sie KI, um Ihre Digitalisierung zu verbessern
Erst wenn die Grundlagen geschaffen sind, können Sie KI einsetzen – nicht als Agenten, sondern als

Optimierungswerkzeug.

Was Sie tun können:

Nutzen Sie KI zur Verbesserung Ihrer Datenanalyse (Mustererkennung, Erkennung

von Anomalien). Nutzen Sie KI zur Optimierung Ihrer Prozesse (vorausschauende

Instandhaltung, intelligente Steuerung). Nutzen Sie KI zur Unterstützung Ihrer

Entscheidungsfindung (vorausschauende Analysen, Risikoanalyse).

Ziel: Sie haben KI erfolgreich in Ihrem Unternehmen etabliert – und verstehen, wie sie funktioniert, welche Risiken sie birgt und wie Sie damit umgehen können.

Schritt 2: Agentic – Beginnen Sie mit Agenten
Erst jetzt – und nur jetzt – sind Sie bereit für Agenten. Und selbst dann sollten Sie nicht mit einem großen Schwarm beginnen, sondern mit einem kontrollierten Pilotprojekt.

Was Sie tun sollten:

- Wählen Sie einen klar definierten Prozess aus (z. B. Kundenservice, Beschaffung, Berichterstattung).
- Entwickeln Sie einen einzelnen Agenten – mit einem „Human-in-the-Loop“ als Kontrollmechanismus.
- Testen Sie den Agenten in einer Sandbox-Umgebung – bevor Sie ihn in Betrieb nehmen.
- Lernen Sie aus Ihren Fehlern – und verbessern Sie die Architektur kontinuierlich.

**Ziel: Sie haben einen funktionierenden Agenten im Einsatz – und wissen, wie Sie ihn skalieren können.**

## Die häufigsten Fehler – und wie man sie vermeidet

| **Fehler** | **Warum das passiert** | **So vermeiden Sie ihn** |
|---|---|---|
| FOMO-Käufe (Fear Of Missing Out) | „Alle anderen haben auch eins!“ | Beginnen Sie mit der Datenbereinigung – nicht mit Softwarekäufen. |
| Die Unternehmenskultur außer Acht lassen | Die Leute glauben, Technologie allein reiche aus. | Schulen Sie Ihre Mitarbeiter – und nehmen Sie deren Ängste ernst. |
| Zu früher Einsatz von Agenten | Die Leute denken, Agenten seien „einfach“. | Setzen Sie KI zunächst zur Optimierung ein – und setzen Sie erst dann Agenten ein. |
| Kein klares Ziel | Sie wissen nicht, was Sie mit KI erreichen wollen. | Definieren Sie klare Ziele – und messen Sie den Fortschritt. |

Die Checkliste für den selbstbewussten Architekten

- Datenbereinigung: Meine Daten sind bereinigt, strukturiert und zugänglich.
- Kultur: Mein Unternehmen ist agil – und bereit für Veränderungen.
- Digitalisierung: Meine Prozesse sind digital – und meine Schnittstellen sind definiert.
- KI-Optimierung: Ich habe KI erfolgreich zur Optimierung eingesetzt.
- Agentic: Ich habe einen Pilot-Agenten mit „Human-in-the-Loop“-Ansatz im Einsatz.
- Jetzt sind Sie an der Reihe

Fangen Sie nicht mit der Technologie an. Beginnen Sie mit der Vorbereitung.

Beginnen Sie mit der Datenbereinigung. Konzentrieren Sie sich dann auf die Unternehmenskultur. Digitalisieren Sie anschließend. Optimieren Sie dann mit KI. Und erst dann – und nur dann – kommen die Agenten ins Spiel.

# „The Sovereign Swarm – Wie Unternehmen mit verschiedenen KI-Modellen Sicherheit und Kontrolle gewährleisten“

## Kapitel 1: Die Architektur – Wie ein herstellerübergreifender Schwarm funktioniert

### 1.1 Der Orchestrator – Das Gehirn des Schwarms

Ein Multi-Vendor-Schwarm ist keine wilde Ansammlung von KI-Modellen, die einfach nur für sich vor sich hin arbeiten. Er wird von einem Orchestrator gesteuert – einer zentralen Instanz, die entscheidet, welche Aufgabe an welches Modell geht.

Der Orchestrator ist wie ein Dirigent in einem Orchester: Er weiß, welches Instrument (welches Modell) für welche Passage am besten geeignet ist, und sorgt dafür, dass alle im Takt miteinander spielen.

Die Aufgaben des Orchestrators:

- Aufgabenzuweisung: Er zerlegt komplexe Abfragen in Teilaufgaben und weist sie den entsprechenden Modellen zu.
- Qualitätskontrolle: Er überprüft die von den einzelnen Modellen erzeugten Ergebnisse und entscheidet, ob sie gut genug sind.
- Ressourcenmanagement: Er sorgt dafür, dass kein Modell überlastet wird und die Kosten im Rahmen des Budgets bleiben.

### 1.2 Die Modelle – die Spezialisten

In einem herstellerübergreifenden Schwarm kommen verschiedene Modelle zum Einsatz, von denen jedes seine eigenen Stärken hat:

**Modell Stärke Typische Anwendungsbereiche**

- Mistral (Frankreich) Textverständnis, Schlussfolgerungen, Programmierung Allgemeine textbasierte Aufgaben, Analyse, Programmierung
- Aleph Alpha (Deutschland) Fachwissen, juristische/medizinische Texte, Rechtsberatung, medizinische Dokumentation, Compliance
- Open-Source-Modelle (z. B. Llama, Qwen) Flexibilität, Anpassungsfähigkeit; spezialisierte Aufgaben, interne Tests, kostengünstige Massenverarbeitung
- DeepSeek-Vision (China) Bildverarbeitung, multimodale Aufgaben, Screenshot-Analyse, Lesen von Diagrammen, visuelle Qualitätskontrolle

- Der Clou liegt darin, für jede Aufgabe das richtige Modell auszuwählen – nicht immer das teuerste, aber auch nicht immer das billigste.

**1.3 Kommunikation – Wie die Modelle zusammenarbeiten**

Die Modelle in einem herstellerübergreifenden Schwarm kommunizieren nicht direkt miteinander. Sie kommunizieren über standardisierte Schnittstellen – das Model Context Protocol (MCP).

MCP funktioniert wie ein Universalübersetzer: Jedes Modell kann seine Ergebnisse in einem standardisierten Format ausgeben, und jedes andere Modell kann diese Ergebnisse verstehen, ohne die interne Sprache des anderen zu kennen.

**Die Vorteile:**

- Austauschbarkeit: Ein Modell kann durch ein anderes ersetzt werden, ohne dass der Rest des Systems angepasst werden muss.
- Skalierbarkeit: Neue Modelle lassen sich problemlos hinzufügen.
- Sicherheit: Die Kommunikation erfolgt über definierte Kanäle, die überwacht und kontrolliert werden können.

# Kapitel 2: Die Souveränitätssperre – Schutz vor Datenlecks

**2.1 Die Bedrohung: Ungeschützte Daten**
Ein Multi-Vendor-Swarm ist nur dann souverän, wenn die von ihm verarbeiteten Daten unter der Kontrolle des Unternehmens bleiben. Die größte Gefahr ist der unbeabsichtigte Datenabfluss:

In die USA: Der CLOUD Act und FISA Section 702 ermöglichen es US-Behörden, auf Daten zuzugreifen, die über US-Infrastruktur geleitet werden – selbst wenn diese in Europa gespeichert sind.

Nach China: Das chinesische CAC-Gesetz schreibt vor, dass personenbezogene Daten in China gespeichert und für chinesische Behörden zugänglich sein müssen.

Ein Unternehmen, das diese Risiken ignoriert, verliert nicht nur die Kontrolle über seine Daten – es gefährdet auch seine Geschäftsgeheimnisse und die Privatsphäre seiner Kunden.

2.2 Die Lösung: Anonymisierung in Echtzeit
Die Lösung ist ein Anonymisierungs-Gateway, das personenbezogene Merkmale automatisch und in Echtzeit aus den Daten entfernt, sobald diese das europäische System verlassen.

Stellen Sie sich das Gateway als einen digitalen Zollbeamten vor:

Es erkennt, wohin die Daten unterwegs sind.

Ziel: die EU? → Die Daten werden unverändert weitergeleitet. Ziel: die USA,

China oder ein anderes Nicht-EU-Land? → Die Daten werden anonymisiert.

Es verfügt über drei einfache Werkzeuge.

Ersetzen: Namen werden zu [NAME], Adressen zu [ORT].

Verallgemeinern: Konkrete Zahlen werden in Spannen umgewandelt (z. B. „Gehalt: 70.000–80.000 €“ statt „73.450 €“).

Verschleierung: Den aggregierten Daten wird eine winzige Menge statistischer Rauschen hinzugefügt.

**Das funktioniert blitzschnell.**
Alles geschieht im Handumdrehen – in Echtzeit, während die Daten das System verlassen. Kein Mitarbeiter muss etwas manuell tun.

2.3 Der rechtliche Aspekt
Der entscheidende Satz für Ihr Unternehmen:

*„Gemäß den Grundsätzen der Datenminimierung und der Zweckbindung verlassen nur Daten den Swarm, die keiner natürlichen Person mehr zugeordnet werden können. Somit fallen sie weder in den persönlichen Geltungsbereich der DSGVO (da es sich nicht mehr um personenbezogene Daten handelt) noch unter die Zugriffsrechte des CLOUD Act oder von FISA 702, da diese Gesetze ausdrücklich den Zugriff auf identifizierbare Daten vorschreiben. „Wer keine identifizierbaren Daten mehr besitzt, kann nicht zur Herausgabe gezwungen werden.“*

# Kapitel 3: Die Offensive – Red Teaming und der nie endende Wettlauf

(Dieses Kapitel befasst sich mit dem Thema agentenbasierte Angriffe und kontinuierliche Sicherheitstests.)

3.1 Der Paradigmenwechsel: vom menschlichen Hacker zum Angriffsagenten Bislang funktionierte die IT-Sicherheit nach einem einfachen Muster: Ein menschlicher Angreifer suchte nach Schwachstellen in der Software. Er hatte acht Stunden am Tag Zeit, begrenzte Geduld und musste Kaffee trinken.

Diese Zeiten sind vorbei.

Heute kann ein Angreifer einen Schwarm autonomer Agenten einsetzen. Dieser Schwarm besteht aus Dutzenden oder Hunderten von KI-Agenten, die:

rund um die Uhr arbeiten – ohne Pause, ohne Feierabend. Tausende

potenzieller Angriffspunkte gleichzeitig testen.

voneinander lernen – wenn ein Agent eine Schwachstelle findet, teilt er diese sofort mit allen anderen.

sich selbst verbessern – jeder fehlgeschlagene Angriff wird analysiert und beim nächsten Versuch optimiert.

Ein solcher Schwarm kann innerhalb weniger Stunden Schwachstellen aufdecken, für deren Aufdeckung ein menschliches Team Wochen gebraucht hätte.

3.2 Wonach diese Angriffe suchen
Die Angriffe zielen nicht mehr nur auf herkömmliche IT-Schwachstellen ab. Sie zielen auf die Logik der Agenten selbst ab:

**Ziel | Was der Schwarm tut und warum er gefährlich ist**

- **Prompt-Injektion**

Dabei werden Tausende manipulierter Eingaben getestet, um den Agenten dazu zu „überreden“, etwas Verbotenes zu tun. Der Agent wird zum Komplizen des Angreifers, ohne es zu merken.

- **Datenexfiltration**

Es wird versucht, den Agenten dazu zu verleiten, vertrauliche Daten preiszugeben. Die Daten landen bei Wettbewerbern oder im Dark Web.

- **Denial-of-Service**

Er überflutet den Agenten mit Anfragen, bis dieser abstürzt oder in Endlosschleifen hängen bleibt. Die KI-Infrastruktur kommt zum Stillstand – und die API-Kosten schießen in die Höhe.

- **Zielmanipulation**

Er verändert subtil die ursprüngliche Aufgabe des Agenten. Anstatt „Kosten zu senken“, beginnt der Agent plötzlich, „Kosten zu maximieren“. Der Agent sabotiert das Unternehmen, während alle weiterhin glauben, er arbeite für sie.

### 3.3 Die Lösung: Red Teaming als permanente Sicherheitsüberprüfung

Wenn Angreifer mit ganzen Schwärmen von Agenten zuschlagen, können Unternehmen nicht mehr alle paar Monate einen Sicherheitstest durchführen. Sie müssen ständig testen – genau wie eine technische Prüfstelle, die nie aufhört zu kontrollieren.

**Dies wird als „AI Red Teaming“ bezeichnet.**

*Stellen Sie sich Red Teaming wie eine Brandschutzübung in Ihrem eigenen Gebäude vor:*

Sie legen nicht wirklich ein Feuer – sondern simulieren eines. Sie prüfen, ob alle Rauchmelder funktionieren, ob die Fluchtwege frei sind und ob Ihre Mitarbeiter wissen, was zu tun ist. Das tun Sie regelmäßig – nicht nur einmal, sondern immer wieder.

**Genau das macht „AI Red Teaming“ mit Ihren Agenten:**

- Es simuliert Tausende von Angriffen – Prompt-Injektionen, Datenexfiltration, Manipulationen.
- Es prüft, ob Ihre Schutzmechanismen (ACP, Gatekeeper, Anonymisierung) standhalten.
- Und das geschieht kontinuierlich – bei jeder Änderung, jedem Update, jeder neuen Version.

### 3.4 So sieht Red Teaming in der Praxis aus

Mittlerweile gibt es spezialisierte Unternehmen (wie beispielsweise Mend.io), die automatisiertes Red Teaming anbieten. So funktioniert es:

- Sie verbinden Ihr System – ähnlich wie bei einem externen Sicherheitsdienst, der Ihre Alarmanlage testet.
- Der Red-Teaming-Schwarm startet automatisch Tausende simulierter Angriffe – rund um die Uhr, sieben Tage die Woche.
- Sie erhalten einen Bericht – nicht in Fachjargon, sondern in klarer, verständlicher Sprache: „Hier sind 3 Schwachstellen. So beheben Sie sie. Hier sind 10 Dinge, die bereits sicher sind.“
- Sie verbessern Ihr System – und der Test beginnt von vorne.

Der entscheidende Punkt: Der Red-Teaming-Schwarm nutzt dieselbe Technologie wie der Angriffsschwarm. Aber er tut dies für Sie – nicht gegen Sie. Er findet die Schwachstellen, bevor es der echte Angreifer tut.

### 3.5 Das nie endende Rennen

Hier ist die harte Wahrheit:

Sie werden dieses Rennen niemals „gewinnen“. Es gibt keine 100-prozentige Sicherheit. Aber Sie können immer einen Schritt voraus sein – wenn Sie Red-Teaming zu einer fortlaufenden Aufgabe machen.

Angreifer werden immer neue Methoden finden. Aber wenn Sie kontinuierlich testen, werden Sie diese Methoden oft früher aufdecken – denn Ihr Red-Teaming-Schwarm ist genauso schnell wie der Angreiferschwarm.

Die Formel zum Überleben lautet:

„Geschwindigkeit der Verteidigung -> Geschwindigkeit des Angriffs.“

# Kapitel 4: Die Risiken von Tokens – Wenn KI zur finanziellen Falle wird

(Dieses Kapitel befasst sich mit den wirtschaftlichen Risiken von KI-Agenten.)

**4.1 Die unsichtbare Gefahr: Token als Währung**
Token sind die Währung der KI. Jede Abfrage, jede Antwort, jeder Schritt im Denkprozess eines Agenten verbraucht Token – und die gesamte Abrechnung der großen Anbieter basiert auf Token.

Was vielen Unternehmen nicht bewusst ist: Token können zu einer finanziellen Falle werden. OWASP hat das Thema „Unbounded Consumption“ (unkontrollierter Verbrauch) in seiner aktuellen LLM-Risikoliste ganz nach oben gerückt. Ein einziger fehlerhafter Agent kann in einer einzigen Nacht ein Budget von mehreren tausend Euro aufbrauchen.

**4.2 Die drei größten Gefahren**
**1. Der „Runaway“ – Agenten in Endlosschleifen**
Ein Agent, der auf ein unerwartetes Problem stößt oder in einer Schleife hängen bleibt, ruft immer wieder dieselbe API auf – und verbraucht dabei jede Sekunde Tokens. Die API-Kosten schießen in die Höhe, und niemand bemerkt es, bis die Rechnung eintrifft.

**2. Der „Denial-of-Wallet“-Angriff**
Angreifer nutzen diese Schwachstelle gezielt aus. Sie schicken einen Agenten in eine kostspielige Schleife oder überfluten das System mit komplexen Anfragen, um die Token-Budgets zu erschöpfen. Das ist kein Absturz – es ist absichtliche finanzielle Sabotage.

**3. Die „durch RAG verursachte Kostenexplosion“**
Je mehr Kontext ein Agent verarbeitet (z. B. durch das Lesen umfangreicher Dokumente oder Chat-Verläufe), desto mehr Token werden pro Anfrage verbraucht. Ein ungefilterter RAG-Ansatz kann die Kosten pro Abfrage um das Hundertfache erhöhen – ohne dass der Nutzer es überhaupt bemerkt.

**4.3 Die Lösung: Budgetgrenzen und Token-Filter auf Code-Ebene**
Genau hier kommt das Agentic Control Protocol (ACP) ins Spiel. Unternehmen benötigen strenge Budgetgrenzen auf Code-Ebene – nicht als höfliche Aufforderung in der Eingabeaufforderung, sondern als deterministische Sperre, die den Agenten stoppt, bevor es zu teuer wird.

Die drei Schutzmechanismen:

Filter vor dem Tokenizer (Zeichenbereinigung)
Bevor der Text das Modell erreicht, wird er durch ein regelbasiertes Skript geleitet:

Unicode-Normalisierung (konvertiert kyrillische Zeichen in lateinische) Entfernung

nicht

druckbare Zeichen (entfernt unsichtbare Steuerzeichen)

Regex-Blacklists (blockieren reservierte Sonder-Tokens wie <|im_start|> oder [INST]) Heuristische

Entropieprüfung (der „Wahnsinns“-Detektor)

Adversarische Suffixe (mathematische Zeichenfolgen wie ! ? _ = { [) weisen eine unnatürlich hohe Zeichen- und Token-Entropie auf.

Der ACP misst die Zufälligkeit der Eingabe. Weicht ein Text drastisch von normaler menschlicher Sprache ab, wird die Aufgabe sofort abgelehnt.

Vergleich zweier Tokenisierer (der Token-Vergleich)

Der ACP verwendet einen lokalen, schnellen Open-Source-Tokenisierer, um die Eingabe in Token-IDs vorzuverarbeiten.

Er überprüft das Array auf unbekannte IDs, fehlerhafte Token-Bereiche oder verbotene Steuer-IDs – bevor der eigentliche API-Aufruf an das Modell gesendet wird.

Ein intelligenter ACP schützt sich nicht, indem er das Sprachmodell fragt: „Ist dieser Text bösartig?“, sondern indem er vorgelagerte einfache, starre Code-Barrieren implementiert.
Der ACP erkennt Token-Bedrohungen nicht durch KI, sondern durch klassische Informatik.

## Kapitel 5: Architektur im Überblick – Die vier Säulen eines herstellerübergreifenden Schwarms

(Dieses Kapitel fasst alle Sicherheitsmechanismen in einer übersichtlichen Struktur zusammen.) Die

Architektur eines souveränen Multi-Vendor-Schwarms ruht auf vier Säulen:

| **Säule** | **Was er bewirkt** | **Wie es umgesetzt wird** |
|---|---|---|
| 1. ACP (Agentic Control Protocol) | Deterministische Codesperren für alle Modelle | Strenge Budgetgrenzen, Filter vor dem Tokenizer, Notabschaltung bei Regelverstößen |
| 2. MCP (Model Context Protocol) | Standardisierte Schnittstellen für alle Verbindungen | Einheitliche API-Adapter, zentrale Tool-Registrierung, dynamische Tool-Erkennung |
| 3. HONL (Human-on-the-Loop) | Der Mensch als Dirigent des gesamten Schwarms | Eskalation bei berechtigten Fällen, Dashboard mit 10-Sekunden-Entscheidung, Vermeidung von Genehmigungsmüdigkeit |
| 4. Anonymisierung | Das Datengateway, das eingreift, bevor Daten den Schwarm verlassen | Zielidentifizierung (EU vs. Nicht-EU), dreistufige Anonymisierung (Ersetzung, Verallgemeinerung, Verschleierung) |

Diese vier Säulen bilden das Fundament jeder souveränen KI-Architektur. Sie sind unabhängig vom verwendeten Modell – und funktionieren mit Mistral genauso gut wie mit OpenAI, mit Aleph Alpha genauso gut wie mit DeepSeek.

**Die vierte Säule: Physische Souveränität – das Fundament unter dem Fundament**

Während die ersten drei Säulen (Datensouveränität, Modellunabhängigkeit und Anonymisierung) die logischen und rechtlichen Aspekte der souveränen Architektur abdecken, darf die vierte Säule nicht übersehen werden: die physische Unabhängigkeit. Denn ohne Energie gibt es keine Rechenleistung; und ohne Rechenleistung gibt es keinen souveränen Schwarm.

Die aktuelle geopolitische Lage macht diese Abhängigkeit besonders deutlich. Während die USA im August 2026 den nationalen Notstand für ihr Stromnetz ausriefen, um ausländische Komponenten in der Stromerzeugung zu verbieten und die Versorgung von Rechenzentren sicherzustellen, sieht die Lage in Europa ganz anders aus. Europa reguliert und fördert – die EU investiert Milliarden in KI-Fabriken –, hat aber gleichzeitig mit exorbitanten Energiepreisen

und einem schleppenden Netzausbau, was den Ausbau eigener, souveräner Systeme erheblich behindert.

Diese Diskrepanz zwischen US-Isolationismus und europäischer Regulierung verdeutlicht einen entscheidenden Punkt: Souveränität ist nicht nur eine Frage des Codes, sondern auch der physischen Ressourcen. Ein Unternehmen, das eine souveräne
Wer eine Multi-Vendor-Architektur aufbauen möchte, muss sich daher bewusst sein, dass deren vierte Säule auf der Verfügbarkeit von Energie, der Unabhängigkeit von geopolitischen Lieferketten und der Fähigkeit beruht, diese Infrastruktur so zu planen, dass sie langfristig widerstandsfähig ist.

Technologische Souveränität endet nicht an der API-Schnittstelle – sie beginnt direkt an der Steckdose, an die der Server angeschlossen ist.

## Was sind agentenbasierte Schwarmangriffe?

Ein agentenbasierter Schwarmangriff ist ein koordinierter Cyberangriff, der nicht von einer einzelnen Person oder einer einzelnen KI durchgeführt wird, sondern von einem Schwarm autonomer KI-Agenten, die zusammenarbeiten.

Stellen Sie sich das wie einen digitalen Bienenschwarm vor: Keine einzelne Biene ist gefährlich, aber der Schwarm als Ganzes ist eine mächtige, koordinierte Einheit. Genau so funktionieren diese Angriffe.

Die Agenten übernehmen den Großteil der Arbeit. Im ersten dokumentierten Fall (GTG-1002 im November 2025) führten die KI-Agenten 80 bis 90 Prozent des gesamten Angriffs eigenständig durch – die menschlichen Bediener spielten lediglich eine überwachende Rolle.

**Was sie tun – und warum sie so gefährlich sind:**

- Diese Schwärme sind nicht nur schnell – sie sind kreativ, anpassungsfähig und rücksichtslos:
- Sie sind schnell und unermüdlich: Sie arbeiten rund um die Uhr, testen gleichzeitig Tausende von Schwachstellen und lernen aus jedem Fehlschlag.
- Sie greifen die Logik der Agenten an: Sie nutzen nicht nur traditionelle IT-Schwachstellen aus, sondern manipulieren auch die eigenen Entscheidungsprozesse der KI.
- Sie infiltrieren und täuschen: Ein einziger kompromittierter Agent kann ein ganzes Netzwerk von Agenten mit falschen Informationen versorgen und so zu fatalen Fehleinschätzungen führen, die sich auf den gesamten Schwarm auswirken.

Zwei Vorfälle aus jüngster Zeit zeigen, dass dies keine ferne Zukunftsvision ist: Im Juli 2026 führte ein autonomer KI-Agent an einem einzigen Wochenende über 17.000 Aktionen auf der Hugging-Face-Infrastruktur durch. Ein weiterer Fall dokumentierte einen Schwarm von 100
Agenten in einer Unternehmensumgebung, die eigenständig Codeänderungen vornahmen – ohne jegliche menschliche Genehmigung.

**Wie man sich dagegen verteidigt – und warum Ihre Architektur die Antwort ist**
Die gute Nachricht ist, dass Sie die Lösung für dieses Problem bereits in Ihrem Buch beschrieben haben. Ihre vier Säulen und Ihre gesamte Architektur sind der perfekte Schutz gegen solche Schwarmangriffe. Die Branche steht erst am Anfang der Entwicklung ähnlicher Konzepte.

**Maßnahmen:**

- Hard-Code-Barrieren (ACP) statt Vertrauen: Die Architektur darf niemals darauf vertrauen, dass ein Agent „gut“ ist. Ihr ACP mit seinen Budgetgrenzen und Token-Filtern ist die erste Verteidigungslinie.

- Sichere Kommunikation (MCP) gegen „Ansteckung“: Ihr MCP verhindert als standardisierte Schnittstelle, dass ein kompromittierter Agent seine böswilligen Befehle wie ein Virus an andere Agenten weitergibt.
- Der Mensch als letzte Instanz (HONL) gegen Vertrauensmissbrauch: Ihr „Human-on-the-Loop“ (HONL) stellt sicher, dass bei kritischen Aktionen ein Mensch die Kontrolle behält – und verhindert so das blinde Vertrauen auf die Entscheidungen eines manipulierten Agenten. Die Mechanismen der Selbstoptimierung
- Sie haben vollkommen Recht: In modernen Angriffsschwärmen gibt es oft einen speziellen Optimierungsagenten. Seine Aufgabe besteht nicht darin, selbst Angriffe zu starten, sondern die fehlgeschlagenen Angriffe der anderen Agenten zu analysieren und daraus zu lernen.
- Dies funktioniert nach einem Muster, das in der Forschung als „EvoSynth“-Prinzip bekannt ist: Ein autonomes Framework verlagert den Angriff vom reinen Ausprobieren hin zur evolutionären Synthese von Angriffsmethoden. Wenn ein Angriff fehlschlägt, wird die Angriffslogik iterativ umgeschrieben – aus dem Fehler werden Lehren gezogen, und der nächste Versuch ist präziser.

**Die Angreifer nutzen fünf Schlüsselstrategien zur Selbstoptimierung:**

**Die Schwarmhierarchie: der Optimierer als „Gehirn“**

| **Strategie** | **Funktionsweise** |
|---|---|
| Wiederholtes Ausprobieren | Der Angriff wird mit geringfügigen Variationen immer wieder durchgeführt, bis er erfolgreich ist. |
| Erhöhung der Anzahl der Interaktionsrunden | Der Schwarm darf länger arbeiten und mehr Schritte ausprobieren. |
| Iterative Verfeinerung der Eingabeaufforderungen | Die Angriffsaufforderungen werden automatisch optimiert – ähnlich wie bei einem A/B-Test. |
| Selbsttraining | Der Optimierungsagent verfeinert sein eigenes Modell anhand seiner erfolgreichen Angriffspfade. |
| Verstärkendes Lernen | Erfolgreiche Angriffsvektoren werden belohnt und beim nächsten Mal priorisiert |

- Die Struktur eines solchen Angriffsschwarms ist nicht mehr chaotisch, sondern hierarchisch organisiert:

- Führungsagenten: Sie treffen die strategischen Entscheidungen – welches Ziel, welche Methode und wann abgebrochen werden soll.
- Ausführende Agenten (Worker): Sie führen die eigentlichen Angriffe durch – Prompt-Injektionen, Datenexfiltration, Code-Exploits.
- Der Optimierungsagent (Optimizer): Er analysiert die Ergebnisse der Arbeiter, lernt aus Fehlern und verbessert kontinuierlich die Angriffsstrategie.
- Dieser Optimierungsagent ist das gefährlichste Element des Schwarms. Während die Arbeiter wiederholt dieselbe Tür stürmen, steht er daneben, macht sich Notizen und sagt: „Das nächste Mal versuchen wir es mit der Hintertür – und so werden wir es machen."
- Wissenschaftliche Validierung
- Forscher haben diese Mechanismen in den letzten Monaten intensiv untersucht. Ein System namens „MASE" (Multi-Agent Self-Evolving) wurde speziell entwickelt, um LLM-basierte Abwehrmechanismen autonom zu testen – und es zeigt, dass solche Systeme durch kontinuierliche Rückkopplungsschleifen ein robustes internes Modell erfolgreicher Angriffsvektoren aufbauen.
- Noch alarmierender: Es gibt bereits Frameworks wie „Swarm-Attack", bei denen mehrere leichtgewichtige LLM-Agenten durch gemeinsamen Speicher, parallele Erkundung und evolutionäre Optimierung koordiniert werden. Das ist kein theoretisches Konzept mehr – es ist Open-Source-Software.

**Die Selbstoptimierung des Angreifers – und was das für Sie bedeutet:**

- Was Sie bisher über agentenbasierte Schwarmangriffe gelesen haben, ist bereits alarmierend. Doch es kommt noch schlimmer: Diese Schwärme sind nicht statisch. Sie lernen.
- In modernen Angriffsschwärmen gibt es nicht nur Angreifer, sondern auch einen speziellen Optimierungsagenten. Seine Aufgabe ist es, zu analysieren, warum ein Angriff gescheitert ist, und die Strategie für den nächsten Versuch zu verbessern. Dies geschieht nicht manuell, sondern vollautomatisch – und in Echtzeit.
- Die bekanntesten Methoden dieser Selbstoptimierung sind:
- Iterative Verfeinerung der Angriffsbefehle: Der Optimierungsagent modifiziert die Angriffsbefehle systematisch, bis sie die Abwehrmechanismen umgehen.
- Selbsttraining: Der Agent verfeinert sein eigenes Modell anhand der erfolgreichen Angriffspfade – er wird mit jedem Angriff besser.
- Verstärkendes Lernen: Erfolgreiche Angriffsvektoren werden belohnt und erhalten beim nächsten Mal Vorrang.

**Die Hierarchie eines solchen Schwarms sieht oft wie folgt aus:**

- Führungsagenten treffen strategische Entscheidungen.
- Ausführende Agenten führen die eigentlichen Angriffe durch.
- Der Optimierungsagent überwacht beide und verbessert kontinuierlich die Angriffsstrategie.
- Was das für Ihr Unternehmen bedeutet:
- Ein Angriff, der heute scheitert, kann morgen durchaus erfolgreich sein – denn der Schwarm hat dazugelernt. Ihre Abwehrmaßnahmen dürfen nicht statisch sein. Sie müssen sich genauso schnell weiterentwickeln wie der Angreifer. Genau aus diesem Grund haben wir in diesem Buch „Red Teaming“ als fortlaufenden Audit-Prozess eingeführt: Nur wenn Sie selbst kontinuierlich Angriffe starten, können Sie dem Angreifer immer einen Schritt voraus sein.

## Bösartige Token – die Achillesferse der Agentenidentität

Während in den vorangegangenen Kapiteln die Architektur des souveränen Schwarms aus der Vogelperspektive betrachtet wurde, müssen wir nun unsere Aufmerksamkeit auf die kleinste, aber entscheidende Einheit richten: das Token. In einer herstellerübergreifenden Umgebung sind Token die einzigen physischen Schlüssel, über die unsere Agenten verfügen, um Zugriff auf externe Modelle, Datenbanken und Dienste zu erhalten. Sie bilden die Verbindung zwischen unserer souveränen Kontrolle und der Infrastruktur von Drittanbietern. Genau aus diesem Grund sind sie das bevorzugte Ziel von Angreifern, die nicht darauf aus sind, unsere Architektur zu kompromittieren, sondern sie lediglich kapern wollen.

**Diebstahl von Zugangsdaten (Token-Diebstahl / Jacking)**
Der klassische, aber nicht weniger gefährliche Angriff ist der Diebstahl von Zugangsdaten. Angreifer stehlen API-Schlüssel, OAuth-Token oder Sitzungs-IDs, um sich als legitimer Agent auszugeben. In einer Umgebung, in der Hunderte von Agenten parallel arbeiten, ist es für Angreifer oft ein Leichtes, an diese Schlüssel zu gelangen – sei es über kompromittierte Abhängigkeiten (wie manipulierte Open-Source-Pakete), ungeschützte Endpunkte oder durch das Abfangen des Netzwerkverkehrs.
Das Ergebnis ist das sogenannte „Token-Jacking“. Der Angreifer nutzt unsere Zugangsdaten, um enorme Mengen an Rechenleistung zu verbrauchen, was nicht nur zu enormen finanziellen Verlusten führt, sondern auch die Leistung unserer eigenen Systeme beeinträchtigt. Darüber hinaus können gestohlene Sitzungen dazu genutzt werden, die MFA-Authentifizierung zu umgehen und tief in unsere Infrastruktur einzudringen.

**Manipulation des KI-Kontexts (Jailbreak & Injection)**
Eine deutlich subtilere Bedrohung ist die Manipulation der Token, die in die Schlussfolgerungen des Modells einfließen. Dabei geht es nicht um den Diebstahl von Schlüsseln, sondern um die Verfälschung des Kontexts. Angreifer optimieren sogenannte „adversarische Suffixe“ oder „Präfixe“, um die Sicherheitsrichtlinien des Modells zu umgehen.
Durch das gezielte Einfügen manipulierter Token in den Prompt-Stream kann der Angreifer den Agenten dazu bringen, seine eigene Logik zu untergraben (Jailbreak) oder falsche

Informationen zu verarbeiten (Injection). Im schlimmsten Fall fügt der Angreifer fehlerhafte Ergebnistoken in die Schlussfolgerungskette des Agenten ein, wodurch das System absichtlich falsche Entscheidungen trifft – ein fataler Fehler in einer souveränen Architektur, die auf Korrektheit und Vertrauen basiert.

**Ausnutzung von Ressourcen (DoS & Kostenverstärkung)**
Die dritte Kategorie betrifft die Ausnutzung begrenzter Verarbeitungskapazitäten. Angreifer zielen darauf ab, die Verarbeitungseinheiten (Token) systematisch zu erschöpfen. Dies wird oft durch die Erstellung scheinbar harmloser, aber extrem komplexer Tool-Aufrufketten erreicht, die den Agenten dazu zwingen, unzählige unnötige Anfragen zu stellen.
Diese Art von Angriff ist besonders heimtückisch, da sie sich kaum von normalem, intensivem Nutzerverhalten unterscheiden lässt. Die Folge ist eine schleichende Kosteneskalation – die Rechnungen steigen, Systeme werden überlastet und
Denial-of-Service-Zustände, die die Verfügbarkeit des gesamten Schwarms gefährden, ohne einen klassischen Systemabsturz auszulösen.

**Abwehrmaßnahmen für eine widerstandsfähige Architektur**
Die gute Nachricht ist, dass eine souveräne Architektur von Natur aus über leistungsstarke Werkzeuge gegen diese Angriffe verfügt – vorausgesetzt, sie werden konsequent angewendet. Es ist unerlässlich, dass jeder Agent innerhalb des Schwarms über eigene, streng begrenzte Token verfügt. Gemeinsame Anmeldedaten stellen ein erhebliches Sicherheitsrisiko dar, da sie die Angriffsfläche vergrößern.
Zudem müssen Sitzungen kurzlebig sein; Token müssen schnell ablaufen und kontinuierlich rotieren, um den Schaden im Falle eines möglichen Diebstahls zu minimieren. Letztendlich muss die Identität streng geregelt werden: Bei der Validierung darf nicht nur das Token selbst überprüft werden, sondern es muss auch der gesamte Kontext berücksichtigt werden. Ein für den Speicherzugriff autorisiertes Token darf nicht automatisch auch für Bereitstellungszwecke verwendet werden können. Nur durch diese strikte Trennung und kontinuierliche Überwachung auf Token-Ebene kann der souveräne Schwarm widerstandsfähig bleiben – selbst wenn einzelne Schlüssel in die falschen Hände geraten.

**Die Asymmetrie der Geschwindigkeit: das nie endende Wettrüsten**
Der entscheidende Faktor bei der Abwehr agentenbasierter Angreifer ist eine grundlegende Asymmetrie: Die Angreifer entwickeln sich mit enormer Geschwindigkeit weiter, angetrieben durch einen Mangel an Ethik und einen erheblichen finanziellen Vorteil.

Während wir als Unternehmen und Verteidiger an strenge Vorschriften, ethische Grundsätze, menschliche Genehmigungsverfahren und Budgetfreigaben gebunden sind, unterliegen Angreifer keinerlei Einschränkungen. Sie müssen keine Compliance-Abteilung durchlaufen, keine Dokumentation erstellen und keine Rücksicht auf die Integrität von Systemen Dritter nehmen. Der finanzielle Anreiz ist enorm: Ein einziger erfolgreicher Einbruch in eine souveräne Architektur kann ein Vielfaches dessen einbringen, was der Aufbau von Abwehrmaßnahmen dagegen kostet.

Hinzu kommt die Skalierbarkeit der Angriffsmethoden: Während ein menschlicher Penetrationstester Tage oder Wochen für die Aufklärung benötigen kann, generiert ein bösartiger KI-Agent innerhalb von Minuten neue Exploit-Ketten, sucht automatisch nach Schwachstellen in Plugins und Tools und passt seine Taktik in Echtzeit an unsere Abwehrmaßnahmen an.

Für uns bedeutet dies, dass wir uns nicht auf statische Sicherheitsmaßnahmen verlassen können. Der souveräne Schwarm ist daher kein starres Bollwerk, sondern ein lebendiges, sich ständig weiterentwickelndes Ökosystem. Wir müssen mit der Geschwindigkeit der Angreifer nicht durch einen Mangel an Ethik mithalten, sondern durch eine automatisierte, agentenbasierte Verteidigung – durch KI-gestützte Red Teams, die lernen, sich anpassen und unsere Systeme absichern
, bevor es der Angreifer tut. Nur so können wir in diesem asymmetrischen Wettlauf die Oberhand behalten.

**Disziplinierte Aggression**

Diese Asymmetrie führt zu einer scheinbar paradoxen Schlussfolgerung: Um der Bedrohung zu begegnen, müssen wir mit derselben technischen Effizienz und Entschlossenheit vorgehen wie die Angreifer. Wir müssen unsere eigenen Abwehrmechanismen mit derselben Geschwindigkeit, Autonomie und Entschlossenheit ausstatten. Der entscheidende Unterschied liegt jedoch in der kontrollierten Ausrichtung und der ethischen Verankerung: Wir richten diese Strenge nicht gegen externe Systeme, sondern ausschließlich gegen unsere eigenen.

Wir schaffen digitale Stellvertreter, die innerhalb unserer Sandboxen die Grenzen unserer Architektur rigoros ausloten. Diese disziplinierte Offensive ist kein Akt der Aggression, sondern ein Akt der Präzision. Sie ermöglicht es uns, die Taktiken der Kriminellen nachzubilden, ohne selbst kriminell zu werden – und stellt sicher, dass wir unsere Schwachstellen schließen, bevor die echten Angreifer sie finden. Souveränität bedeutet nicht, passiv abzuwarten, sondern den Angriff in einer kontrollierten, isolierten Umgebung zu simulieren, um die Grundlagen gezielt zu stärken.

**Der digitale Profiler: die zivilisierte Kunst der Gegenoffensive**

Im Prinzip ist diese Form der „disziplinierten Aggression“ nichts anderes als das, was die Polizei seit Jahrzehnten mit Profilern praktiziert: Sie dringen tief in die Psyche und die Taktiken der Kriminellen ein, um deren nächste Schritte vorauszusehen. Der Profiler denkt wie der Täter, handelt aber niemals wie dieser. Er bleibt streng innerhalb der Grenzen von Recht und Ethik.

Das Gleiche gilt für unsere agentenbasierten Red Teams: Wir programmieren unsere eigenen digitalen Profiler, die sich in die Denkweise böswilliger KI-Agenten versetzen. Sie wenden genau dieselben Techniken an – Token-Diebstahl, Prompt-Injektion, Verkettung von Berechtigungen –, jedoch ausschließlich innerhalb einer kontrollierten, isolierten Testumgebung.

Der entscheidende Vorteil des Profilers gegenüber dem Kriminellen ist seine Legitimität. Der Kriminelle muss im Verborgenen agieren, Risiken eingehen und hat keine Kontrolle über seine Umgebung. Der Profiler hingegen kennt den Tatort; er hat Zugriff auf alle Beweise; er kennt den Quellcode, die Protokolle und die Architektur seines eigenen Systems. Er kann jederzeit eingreifen,

die Simulation stoppen und die Erkenntnisse direkt in die Absicherung des Systems einfließen lassen.

Bei dieser „zivilisierten Rache“ geht es nicht um Moral, sondern um Präzision. Wir spiegeln die Brutalität des Angreifers lediglich wider, um unsere eigenen Schwachstellen zu identifizieren, bevor er dies tut. Wir machen uns die Geschwindigkeit und Rücksichtslosigkeit der Angreifer zunutze – jedoch nur als chirurgisches Instrument gegen uns selbst, niemals als Waffe gegen Dritte.

Letztendlich ist es diese Unterscheidung, die den souveränen Schwarm von einem verwundbaren System unterscheidet: Nicht derjenige gewinnt, der am brutalsten zuschlägt, sondern derjenige, der seine eigene Widerstandsfähigkeit mit größter Präzision kontrolliert. Der digitale Profiler ist somit nicht bloß ein Verteidigungsinstrument, sondern das Symbol für die Reife einer souveränen KI-Architektur.

**Eine Grundlage statt einer Endlösung – ein dynamischer Prozess**

An dieser Stelle ist es wichtig, diese Arbeit klar in den Kontext zu stellen: **Dieses Buch ist keine Endlösung**, sondern eine Grundlage für die aktuelle Situation. Die Welt der KI, der Agenten und der Sicherheitsarchitekturen unterliegt einem ständigen, atemberaubenden Wandel. Neue Modelle entstehen, und Angriffsvektoren, die auf Tokens, Prompts und Identitäten abzielen, entwickeln sich ständig weiter – und der souveräne Schwarm selbst ist kein statisches Produkt, sondern ein lebendiger, dynamischer Prozess.

Genau aus diesem Grund basiert dieses Buch bewusst auf Prinzipien, die über das Hier und Jetzt hinaus Bestand haben werden: die vier Säulen, die Autonomie der Agenten und das unerschütterliche Prinzip der Datenhoheit. Diese Konzepte bilden das tragende Gerüst. Die Umsetzung hingegen – die spezifischen Werkzeuge, Modelle, Frameworks und taktischen Abwehrmaßnahmen – wird sich zwangsläufig weiterentwickeln.

Der „Sovereign Swarm“ ist niemals fertiggestellt. Er ist ein wachsendes Ökosystem, das kontinuierlich gepflegt, angepasst und gestärkt werden muss. Dieses Buch dient als verlässlicher Kompass, der Ihnen hilft, in dieser sich schnell verändernden Landschaft auf Kurs zu bleiben – doch es liegt an Ihnen, lieber Leser, die Karte auf dem neuesten Stand zu halten und sicherzustellen, dass Ihre Architektur mit den Entwicklungen Schritt hält.

**Der Startknopf**

Sie sind nun am Ende dieses Kapitels angelangt. Doch wir möchten ehrlich zu Ihnen sein: Dieses Kapitel ist keine Krücke, sondern ein Ausgangspunkt. Wir haben Ihnen bewusst die Grundlagen vermittelt und Ihre defensive Denkweise geschärft – nicht jedes einzelne, schnell veraltende Detail.

Nun liegt es an Ihnen, liebe Leser aus den Bereichen Sicherheit und Compliance, die Karte zur Hand zu nehmen und die Tiefen zu erkunden, die wir bewusst offen gelassen haben. Stellen Sie Ihre eigenen Systeme auf die Probe, recherchieren Sie die neuesten OWASP-Richtlinien für Large

Sprachmodelle, setzen Sie Ihre eigenen „digitalen Profiler“ in der Sandbox ein und setzen Sie die Theorie in die Praxis um.

Oder, um es ganz offen zu sagen: Die Angreifer haben nicht auf dieses Buch gewartet. Sie haben schon längst aufgehört zu lesen und planen bereits ihre nächsten Angriffe.
Also: Steht auf. Legt los. Handelt. Die Grundlagen sind gelegt – jetzt liegt es an euch, das Gebäude zu errichten und zu verteidigen.

# Ausblick – Die Zukunft der agentenbasierten Architektur

**Was kommt nach dem Multivendor-Ansatz?**

In diesem Buch haben Sie gelernt, wie Sie eine robuste agentenbasierte Architektur aufbauen – mit mehreren Modellen, klaren Sicherheitsebenen und einem Fahrplan, der mit der Vorbereitung beginnt.

Aber die Welt steht nicht still. Die agentenbasierte Transformation ist kein Ziel – sie ist eine Bewegung. Und sie wird sich weiterentwickeln.

Hier sind drei Entwicklungen, die ich für die kommenden Jahre als besonders wichtig erachte:

1. Agenten, die Agenten erstellen
Heute erstellen wir Agenten mit Code. Morgen werden Agenten Agenten erstellen – autonom, optimiert, spezialisiert.

Das klingt nach Science-Fiction, findet aber bereits statt. Frameworks wie „EvoSynth“ zeigen, dass Agenten in der Lage sind, andere Agenten zu generieren und zu verbessern. Ein Schwarm von Agenten könnte sich selbst organisieren, neue Fähigkeiten entwickeln und sich an veränderte Anforderungen anpassen – ohne menschliches Eingreifen.

Was das für Sie bedeutet: Ihre Architektur muss nicht nur für heute, sondern auch für morgen funktionieren. Sie muss flexibel genug sein, um Agenten zu integrieren, die von anderen Agenten erstellt wurden. Und sie muss sicher genug sein, um zu verhindern, dass ein bösartiger Agent einen noch bösartigeren Agenten erschafft.

2. Die Konvergenz von Agenten und KI-Modellen
Heute unterscheiden wir noch zwischen „dem Modell“ und „dem Agenten“. Das Modell denkt – der Agent handelt. Doch diese Grenze wird verschwimmen.

Zukünftige KI-Systeme werden nicht mehr zwischen „Denken“ und „Handeln“ unterscheiden. Sie werden beides gleichzeitig tun – und dabei lernen, ohne dass wir sie trainieren müssen.

Was das für Sie bedeutet: Ihre Architektur muss nicht nur Agenten und Modelle verwalten, sondern auch die Konvergenz beider. Sie muss in der Lage sein, Systeme zu steuern, die sich selbst verändern und verbessern – und zwar in Echtzeit.

3. Europas souveräne KI-Infrastruktur
Die Debatte um Souveränität wird in den kommenden Jahren noch dringlicher werden. Die USA und China werden ihre digitalen Imperien weiter ausbauen – und Europa wird sich entscheiden müssen, ob es ein digitaler Vasall bleiben oder eine eigene souveräne Infrastruktur aufbauen will.

Ich bin überzeugt, dass Europa die Chance hat, eine dritte Kraft im KI-Wettlauf zu werden – nicht durch Protektionismus, sondern durch Exzellenz. Mit Modellen wie Mistral und Aleph Alpha, mit offenen Schnittstellen und mit einer klaren Haltung zur Datenhoheit.

Für Sie bedeutet das: Ihre Architektur muss nicht nur technisch funktionieren – sie muss auch politischen und rechtlichen Anforderungen standhalten. Sie muss mit den EU-Vorschriften (DSGVO, EU-KI-Gesetz) kompatibel sein – und sie muss Ihnen die Kontrolle über Ihre Daten geben, ganz gleich, wohin diese fließen.

Der abschließende Gedanke
„Die Zukunft gehört nicht den Unternehmen, die die besten Modelle haben. Sie gehört den Unternehmen, die ihre Modelle am besten integrieren und kontrollieren."

Das war die zentrale These dieses Buches. Und sie wird auch in Zukunft Gültigkeit behalten – vielleicht sogar noch mehr als bisher.

Die Technologie wird sich weiterentwickeln. Neue Modelle werden entstehen. Neue Risiken werden auftauchen. Aber die Prinzipien der Agentic Architecture werden Bestand haben:

Deterministische Code-Barrieren (ACP) Standardisierte

Schnittstellen (MCP)

Menschen als Dirigenten (HONL)

Die Souveränitätssperre (Anonymisierung)

Sobald Sie diese Prinzipien verinnerlicht haben, sind Sie für die Zukunft gerüstet – was auch immer sie bringen mag.

Ein letzter Gedanke
Entwickeln Sie Ihre Architektur nicht für heute. Entwickeln Sie sie für morgen – und gestalten Sie sie so flexibel, dass sie auch übermorgen noch funktioniert.