FOM Hochschule für Oekonomie & Management

Hochschulzentrum DLS

Exposé

Berufsbegleitender Studiengang
7. Semester
im Studiengang Cyber Security Management (M.Sc.)

im Rahmen der Lehrveranstaltung

über das Thema

Vom Kontrollkatalog zum Erfüllungsgrad: KI-gestützte Klassifikation der Azure-Policy-Automatisierbarkeit von BSI-Grundschutz++-Anforderungen

von

Philipp Rose

Betreuer :
Matrikelnummer : 399061
Abgabedatum : 18. August 2026

Inklusionshinweis

Zur besseren Lesbarkeit wird in dieser Arbeit das generische Maskulinum verwendet. Die in dieser Arbeit verwendeten Personenbezeichnungen beziehen sich – sofern nicht anders kenntlich gemacht – auf alle Geschlechter.

KI-Disclaimer

In dieser Arbeit wurde Künstliche Intelligenz (KI) - in Form von Claude (Anthropic), u. a. über das paper-search-mcp-Werkzeug zur Literaturrecherche - zur Unterstützung bei der Recherche sowie bei der Erstellung und Überarbeitung von Texten eingesetzt. Die KI diente als Hilfsmittel zur Verbesserung der Effizienz und Qualität, wobei alle Inhalte einer sorgfältigen fachlichen Prüfung unterzogen wurden. Entscheidungen über die inhaltliche Ausgestaltung und wissenschaftliche Argumentation wurden ausschließlich vom Autor getroffen. Die Verantwortung für die Richtigkeit und Integrität der Arbeit liegt vollständig beim Verfasser. Der Einsatz von KI erfolgte unter Berücksichtigung ethischer und wissenschaftlicher Standards.

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Abkürzungsverzeichnis

BSI	Bundesamt für Sicherheit in der Informationstechnik
DSR	Design Science Research
DSRM	Design Science Research Methodology
FEDS	Framework for Evaluation in Design Science Research
KI	Künstliche Intelligenz
LLM	Large Language Model
NIST	National Institute of Standards and Technology
OSCAL	Open Security Controls Assessment Language

1 Einleitung und Titelvorschlag

Sicherheitsanforderungen werden in Cloud-Umgebungen zunehmend als maschinenlesbare Open Security Controls Assessment Language (OSCAL)-Artefakte dokumentiert und lösen dokumentbasierte Nachweisverfahren ab(Vgl. Muth & Margraf, 2026b). Das Bundesamt für Sicherheit in der Informationstechnik (BSI) hat seine Stand-der-Technik-Bibliothek Grundschutz++ inzwischen offiziell im OSCAL-Format veröffentlicht, wodurch sie über § 44 Abs. 1 BSIG im Rahmen von NIS-2 gesetzlich verpflichtend wird(Vgl. Bundesamt für Sicherheit in der Informationstechnik, 2025). Damit entsteht eine strukturierte, standardisierte Ausgangsbasis für automatisierte Compliance-Prüfungen auf gesetzlicher Grundlage.

Trotz dieser Grundlage fehlt für Microsoft Azure eine offizielle Zuordnungstabelle der BSI-Grundschutz++-Anforderungen zu konkreten Policy-Mechanismen, obwohl eine vergleichbare Zuordnung für den National Institute of Standards and Technology (NIST)-800-53-Katalog bereits existiert(Vgl. Microsoft, 2026). Frühere Arbeiten zum automatisierten Mapping zwischen Compliance-Dokumenten belegen, dass sich derartige Zuordnungen algorithmisch unterstützen lassen, allerdings nur im Kontext einzelner, bereits etablierter Kontrollkataloge, die vor der erst 2025 erfolgten OSCAL-Standardisierung von Grundschutz++ vorlagen(Vgl. Tupsamudre et al., 2022, S. 12629-12635)(Vgl. Agarwal et al., 2021, S. 136-146). Für Organisationen, die Grundschutz++-Anforderungen auf Azure durchsetzen wollen, bleibt die Übersetzung einzelner Kontrollen in automatisiert prüfbare Policies daher eine manuelle, wenig nachvollziehbare Aufgabe. Der Umfang wird bei rund 998 Grundschutz++-Anforderungen(Vgl. Bundesamt für Sicherheit in der Informationstechnik, 2025) und einer angenommenen Prüfzeit von 30 bis 60 Minuten pro Anforderung sichtbar: ein einmaliger manueller Prüfaufwand in der Größenordnung von 60 bis 125 Personentagen, eine eigene Schätzung ohne Anspruch auf Präzision.

Diese Arbeit widmet sich dieser Lücke, indem sie einen Künstliche Intelligenz (KI)-gestützten Agenten entwickelt und evaluiert, der einzelne Grundschutz++-Anforderungen gegen die Automatisierbarkeit durch Azure-Policy klassifiziert. Die Klassifikation erfolgt dreistufig in vollständig, teilweise oder nicht erfüllbar und stützt sich, wo möglich, auf Kandidaten aus Microsofts bestehender NIST-800-53-Zuordnung. Bestehende Arbeiten verifizieren Sicherheitskonzepte gegen OSCAL oder überführen Freitext-Systembeschreibungen in OSCAL-Artefakte(Vgl. Muth & Margraf, 2026a)(Vgl. Muth & Margraf, 2026b). Der eigene Ansatz setzt dagegen auf bereits strukturiertem OSCAL-Input auf und verlangt keine Extraktion, sondern eine bewertende Klassifikation. Dieser Unterschied verschiebt den methodischen Schwerpunkt von der Dokumentenanalyse auf die Korrektheitsmessung einer automatisierten Bewertung.

Zwei angrenzende Aufgaben werden bewusst nicht Teil des Artefakts. Die organisationsspezifische Anpassung eines Katalogs ist über den OSCAL-Profile-Mechanismus (Import, Merge, Modify) bereits standardisiert gelöst und wird nicht neu konstruiert. Das Compliance-Reporting nach erfolgter Policy-Zuweisung deckt Microsoft Defender for Cloud bereits nativ ab. Beide Mechanismen fließen lediglich als Einordnung in die Motivation dieser Arbeit ein, sind jedoch kein Gegenstand von Konstruktion, Demonstration oder Evaluation. Diese Abgrenzung hält den Untersuchungsgegenstand auf die Crosswalk-Klassifikation selbst fokussiert, statt eine vollständige Compliance-Plattform nachzubilden.

Titelvorschläge:

- KI-gestützter Crosswalk von BSI-Grundschutz++-Anforderungen zu automatisiert durchsetzbaren Azure-Policies: eine Abdeckungsanalyse für Compliance-as-Code
- Nachvollziehbare Crosswalk-Klassifikation: Ein KI-Agent zur Bewertung der Azure-Policy-Automatisierbarkeit von BSI-Grundschutz++-Anforderungen
- Vom Kontrollkatalog zum Erfüllungsgrad: KI-gestützte Klassifikation der Azure-Policy-Automatisierbarkeit von BSI-Grundschutz++-Anforderungen

2 Forschungsfrage und Zielsetzung der Arbeit

Automatisierte Klassifikation strukturierter Compliance-Kataloge unterscheidet sich grundlegend von der Extraktion aus Freitext-Dokumentation. Bestehende Ansätze verifizieren legacy Sicherheitskonzepte gegen OSCAL oder überführen Systembeschreibungen aus Freitext in OSCAL-Artefakte(Vgl. Muth & Margraf, 2026a)(Vgl. Muth & Margraf, 2026b). Der Input liegt in der vorliegenden Arbeit dagegen bereits vollständig strukturiert als OSCAL-Catalog vor. Dadurch verschiebt sich die zentrale Herausforderung von der Dokumentenanalyse auf eine belastbare Bewertung, in welchem Grad einzelne Anforderungen durch vorhandene Cloud-Mechanismen automatisiert durchsetzbar sind. Für Grundschutz++ und Azure ist diese Bewertungsaufgabe bislang ungelöst, woraus sich die folgende Forschungsfrage ergibt.

Die zentrale Forschungsfrage lautet:

> Wie lässt sich mittels eines KI-Agenten für die seit September 2025 in OSCAL vorliegenden, nach NIS-2 (§ 44 Abs. 1 BSIG) verpflichtenden BSI-Grundschutz++-Anforderungen (Stand-der-Technik-Bibliothek) nachvollziehbar klassifizieren, in welchem Grad (vollständig / teilweise / nicht erfüllbar) sie über bestehende Azure-Policy-Mechanismen automatisiert durchsetzbar sind, und wie lässt sich die Korrektheit dieser Crosswalk-Klassifikation objektivierbar operationalisieren?

Der Begriff nachvollziehbar bezeichnet in dieser Arbeit bewusst Prozess-Nachvollziehbarkeit im Sinne eines Audit Trails, nicht Explainable AI. Eingelöst wird diese Anforderung über eine dokumentierte Zuordnung je klassifizierter Anforderung, welche NIST-800-53-Äquivalenz oder welcher Azure-Policy-Kandidat der Klassifikation zugrunde liegt oder weshalb keiner existiert. Ergänzend werden versionierte Prompts und protokollierte Modellparameter dokumentiert. Beantwortet wird damit, was zugeordnet wurde und wie reproduzierbar diese Zuordnung ist, nicht warum das Modell intern zu einer bestimmten Entscheidung gelangt. Diese Abgrenzung begrenzt den Untersuchungsgegenstand auf überprüfbare Ausgaben, statt eine Interpretierbarkeit der internen Modellmechanik anzustreben.

Dabei werden folgende Unterfragen adressiert:

- Welche Agenten-Architektur eignet sich für eine nachvollziehbare, reproduzierbare Crosswalk-Klassifikation?

- Wie lässt sich eine Grundschutz++-Anforderung ohne offizielle Crosswalk-Tabelle auf eine NIST-800-53-Äquivalenz beziehungsweise auf Kandidaten aus Microsofts bestehender NIST-800-53-zu-Azure-Policy-Zuordnung abbilden?
- Wie lässt sich der Erfüllungsgrad (vollständig, teilweise, nicht erfüllbar) einer Anforderung zuverlässig bestimmen, statt nur binär zwischen automatisierbar und nicht automatisierbar zu unterscheiden?
- Wie lässt sich die Korrektheit der Klassifikation objektivierbar messen, einschließlich eines verpflichtenden Baseline-Vergleichs gegenüber klassischem Embedding- oder TF-IDF-Similarity-Matching ohne Large Language Model (LLM)?
- Verbessert eine mehrstufige Agenten-Architektur gegenüber naivem Prompting gezielt jene Fälle, in denen vergleichbare Bestandssysteme laut aktueller Literatur am schwächsten abschneiden, insbesondere bei adversarischer Diskriminierung ambiger Grenzfälle und konvergenter Validität?

Ziel der Arbeit ist die Entwicklung und Evaluation eines KI-Agenten, der Grundschutz++-Anforderungen automatisiert und nachvollziehbar gegen die Azure-Policy-Automatisierbarkeit bewertet. Der Korrektheitsanspruch dieser Klassifikation wird bewusst zweistufig differenziert. Die prozedurale Zuverlässigkeit lässt sich vollautomatisiert über die gesamte Stichprobe prüfen, ohne dass dafür eine inhaltliche Ground Truth vorliegen muss: Konsistenz über wiederholte Durchläufe, Stabilität gegenüber Paraphrasen sowie Trennschärfe bei adversarisch konstruierten Grenzfällen. Die inhaltliche Korrektheit der Klassifikation selbst lässt sich dagegen nur für einen eng begrenzten Anker belastbar beurteilen, eine vorab unabhängig kuratierte Teilmenge mit bereits bekannter NIST-800-53-Äquivalenz. Für den überwiegenden Teil der rund 998 Grundschutz++-Anforderungen(Vgl. Bundesamt für Sicherheit in der Informationstechnik, 2025) existiert keine belastbare Ground Truth, eine unmittelbare Folge derselben fehlenden offiziellen Zuordnung, die bereits die Forschungslücke dieser Arbeit begründet, keine davon losgelöste zweite Lücke.

Die eigenständige wissenschaftliche Leistung liegt in der operationalen Definition der drei Erfüllungsgrade vollständig, teilweise und nicht erfüllbar für die Azure-Policy-Durchsetzbarkeit von Grundschutz++-Anforderungen, einem eigenständigen Konstrukt, das bislang nicht vorliegt. Hinzu kommt die Zweiteilung der konvergenten Validität in einen unabhängig kuratierten NIST-800-53-Äquivalenz-Anker und einen begrenzten ISO-27001-Sanity-Check als weitere domänenspezifische Konstruktionsentscheidung. Die zugrunde liegende fünfdimensionale Evaluationsmethodik selbst ist bereits publiziert und framework-agnostisch angelegt(Vgl. Voicu, 2026). Die vorliegende Arbeit überträgt sie

erstmals auf die konkrete Domänenkombination aus Grundschutz++ und Azure-Policy-Automatisierbarkeit und schließt damit die Lücke einer fehlenden, methodisch abgesicherten Bewertung.

3 Methodik und Vorgehen

Das methodische Vorgehen dieser Masterthesis basiert auf der praxisorientierten Design Science Research (DSR) Methodologie nach Hevner et al. (Vgl. Hevner et al., 2004, S. 75-105). Die Abbildung 1 veranschaulicht den angewendeten DSR-Zyklus mit seinen drei zentralen Komponenten. Der Relevance Cycle verbindet die reale Praxis mit der Forschung, indem vorhandene Probleme und Anforderungen identifiziert werden. Der Rigor Cycle stellt die Verbindung zur Wissenschaft her, durch Nutzung vorhandenen und Generierung neuen Wissens. Im Zentrum des Design Cycle steht die iterative Artefaktentwicklung, deren Ergebnisse in die nächste Iteration einfließen.

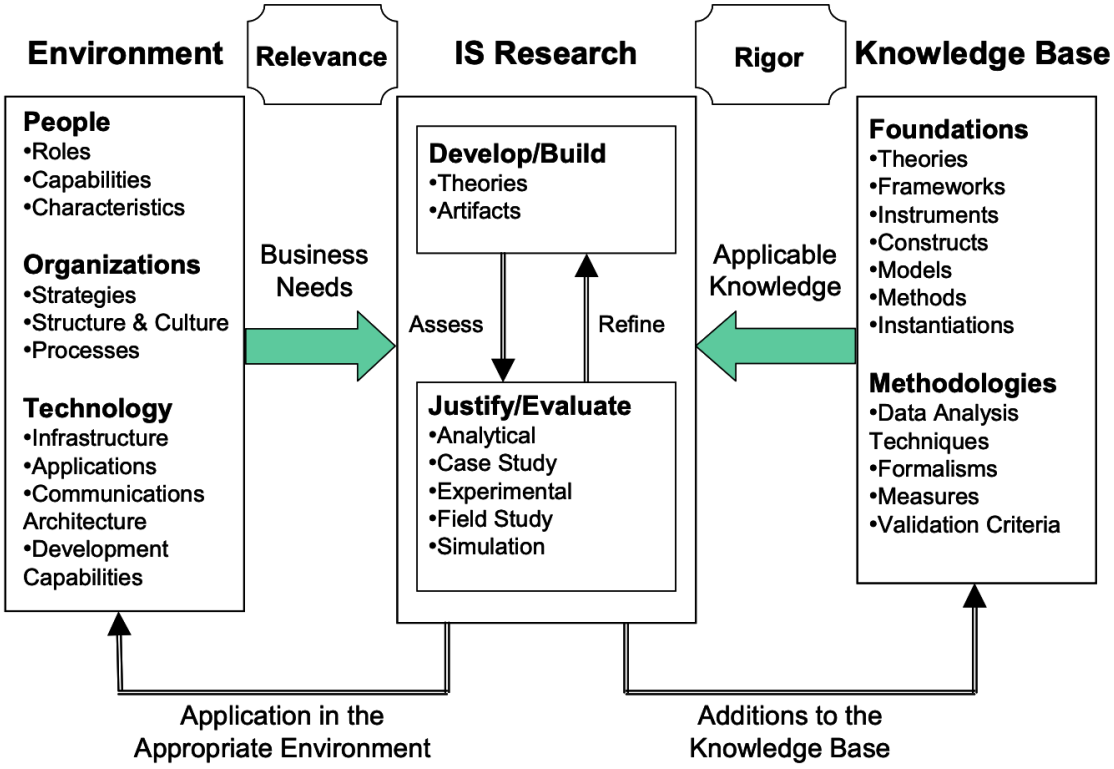

Abbildung 1: Design Science Research - Framework[1]

Der Relevance Cycle verankert den Klassifikationsagenten in einer aktuellen praktischen Notwendigkeit: Die BSI-Stand-der-Technik-Bibliothek ist über NIS-2 gesetzlich verpflichtend, eine offizielle Zuordnung zu Azure-Policy-Mechanismen fehlt jedoch. Der Rigor Cycle bindet bestehendes Wissen zu automatisiertem Compliance-Mapping ein, etwa „live crosswalks“ zur Nachverfolgung von Änderungen zwischen Compliance-Dokumenten (Vgl. Tupsamudre et al., 2022, S. 12629-12635) sowie Ansätze zur Integration heterogener Kon-

[1] Vgl. Hevner et al., 2004, S. 80.

trollkataloge (Vgl. Martinez et al., 2024, S. 90), und generiert zugleich neues Wissen zur Klassifikationsgüte KI-gestützter Verfahren für Grundschutz++ und Azure.

Als konkretes Vorgehensmodell innerhalb des DSR-Zyklus dient die Design Science Research Methodology (DSRM) nach Peffers et al. mit ihren sechs Phasen Problem Identification, Objectives, Design and Development, Demonstration, Evaluation und Communication (Vgl. Peffers et al., 2007, S. 45-77). Problem Identification und Objectives entsprechen der zuvor hergeleiteten Forschungslücke sowie der Forschungsfrage. Design and Development umfasst die Programmierung des Klassifikationsagenten als lauffähigen Prototyp samt Anbindung an Microsofts bestehende NIST-800-53-zu-Azure-Policy-Zuordnung, iterativ entwickelt und angepasst an Testausschnitten der Stichprobe, bevor er auf die vollständige Stichprobe angewendet wird. Demonstration erfolgt an dieser Stichprobe von Grundschutz++-Anforderungen, ergänzt um eine exemplarische Policy-Zuweisung beziehungsweise Ticket-Erstellung als Nachweis der praktischen Anschlussfähigkeit. Evaluation und Communication schließen den Zyklus mit der im Folgenden beschriebenen mehrdimensionalen Prüfung sowie der Einordnung der Ergebnisse ab.

Die Praxisrelevanz ergibt sich nicht aus einem einzelnen Anwendungsunternehmen, sondern aus der Allgemeingültigkeit der Domänenkombination: Grundschutz++ ist über § 44 Abs. 1 BSIG für alle NIS-2-betroffenen Einrichtungen verpflichtend, Azure Policy ist der native Durchsetzungsmechanismus jeder Azure-Umgebung. Eine organisationsspezifische Fallstudie unter Sperrvermerk würde die Communication-Phase der DSRM und den Rückfluss in die Rigor-Cycle-Wissensbasis blockieren. Der praktische Ergebnistyp der Communication-Phase ist eine Abdeckungsanalyse je Grundschutz++-Anforderung, deren nicht erfüllbare Fälle zugleich eine priorisierte Liste für die manuelle Nachkontrolle durch einen Informationssicherheitsbeauftragten bilden.

Der Klassifikationsagent gibt je Anforderung einen von drei Erfüllungsgraden aus, operational definiert, damit Kernagent, Baseline und alle Evaluationsdimensionen denselben Maßstab verwenden:

- **Vollständig erfüllbar**: Mindestens eine Azure-Policy-Definition deckt die Anforderung vollständig ab und setzt sie automatisiert durch, ohne dass eine ergänzende manuelle Kontrolle nötig bleibt.
- **Teilweise erfüllbar**: Eine oder mehrere Azure-Policy-Definitionen decken nur einen Teilaspekt ab, eine ergänzende manuelle Kontrolle bleibt notwendig.
- **Nicht erfüllbar**: Keine passende Azure-Policy-Definition existiert.

Der Umfang von Design and Development wird in eine verpflichtende Minimalstufe und eine optionale Ausbaustufe gestuft, um den Aufwand realistisch zum Zeitrahmen einer Masterthesis in Bezug zu setzen:

- **Minimalstufe** (verpflichtend): Kernagent als solider LLM-Wrapper mit strukturierter Ausgabe, einfache TF-IDF-/Embedding-basierte Baseline mit definierter Score-zu-Klassifikationsgrad-Übersetzung, die vier automatisierten Evaluationsdimensionen auf der vollständigen Stichprobe.
- **Ausbaustufe** (optional, bei ausreichendem Zeitrahmen): erweiterte Agenten-Architektur mit Retrieval-Augmented Generation über den BSI- und NIST-Katalog, iterativer Selbstkorrektur und deterministischer Regelprüfschicht, verbunden mit einer größeren Stichprobe und einer vollständigen Azure-Demonstration anstelle der exemplarischen Policy-Zuweisung.

Die Wahl der Evaluationsstrategie folgt dem Framework for Evaluation in Design Science Research (FEDS)-Framework nach Venable et al., das zwischen künstlicher und naturalistischer Evaluation unterscheidet(Vgl. Venable, Pries-Heje & Baskerville, 2016, S. 77-89). Da das Artefakt ohne Produktivbezug und ohne unmittelbare Sicherheitsrelevanz für Dritte evaluiert wird, rechtfertigt das geringe Risiko die von Venable et al. benannte Strategie „Technical Risk & Efficacy“, angemessen, wenn primär die technische Funktionsfähigkeit im Vordergrund steht und nicht das Verhalten menschlicher Stakeholder. Die Strategie stützt sich ausschließlich auf die im Folgenden beschriebenen automatisierten Prüfdimensionen und Referenzabgleiche.

Zwei Abhängigkeiten begrenzen die Demonstration und werden als Risiko benannt: der derzeit nicht gesicherte Zugriff auf eine für die exemplarische Policy-Zuweisung geeignete Azure-Testumgebung beziehungsweise -Subscription, Gegenstand der Betreuerabsprache, sowie die Abhängigkeit von Microsofts NIST-800-53-zu-Azure-Policy-Zuordnung(Vgl. Microsoft, 2026), einer extern gepflegten Fremdressource ohne eigene Kontrolle über Verfügbarkeit oder Änderungen. Beide Abhängigkeiten wirken sich in erster Linie auf die Demonstration aus, nicht auf die Kernevaluation der Klassifikationsgüte, die auf dem OSCAL-Catalog selbst und den automatisierten Prüfdimensionen basiert. Für beide Risiken ist ein entsprechender Puffer in der Zeitplanung vorgesehen.

Abbildung 2 veranschaulicht das Zusammenspiel aus Kernagent, Baseline, der dreistufigen Klassifikation und den vier im Folgenden erläuterten Evaluationsdimensionen.

Die Notwendigkeit einer rigorosen automatisierten Prüfung generierter Sicherheitsartefakte belegt die Literatur zur Bewertung LLM-generierter Infrastructure-as-Code-Konformität (Vgl. Hase et al., 2026, S. 1-6). Die Korrektheit wird über vier weitgehend automatisierte

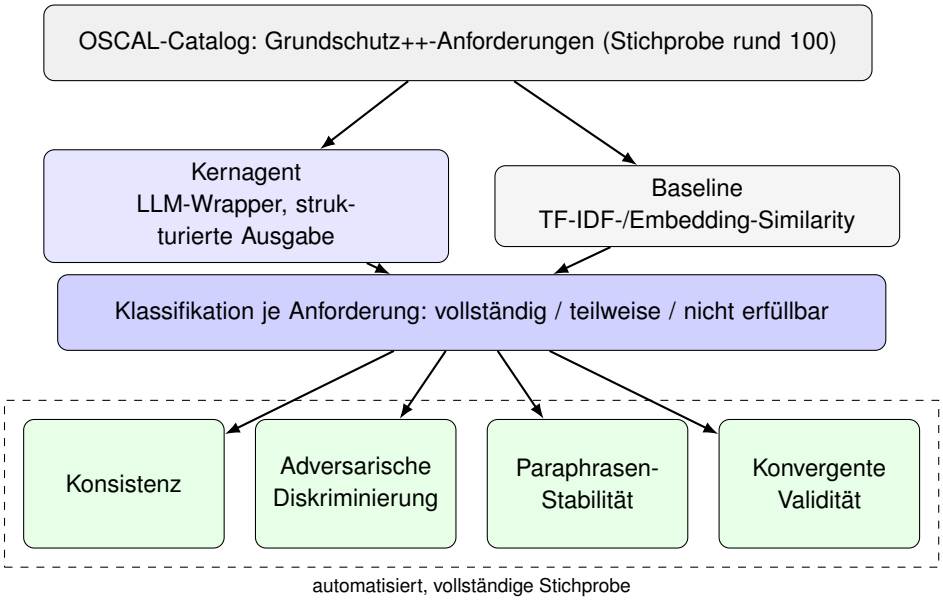

Abbildung 2: Systemarchitektur und Evaluationspipeline: vom OSCAL-Catalog über Kernagent und Baseline zur dreistufigen Klassifikation und den vier Evaluationsdimensionen

Prüfdimensionen operationalisiert(Vgl. Voicu, 2026), von denen drei die prozedurale Zuverlässigkeit messen, ohne eine inhaltliche Ground Truth vorauszusetzen:

- **Konsistenz** prüft die strukturelle Konformität der Klassifikationsausgabe über wiederholte Durchläufe.
- **Adversarische Diskriminierung** misst die Trennschärfe des Agenten bei konstruierten Minimal-Paaren inhaltlich ähnlicher, aber unterschiedlich zu klassifizierender Anforderungen, nach demselben Trennschärfeprinzip wie die diskriminante Validität nach Campbell und Fiske(Vgl. Campbell & Fiske, 1959, S. 81).
- **Paraphrasen-Stabilität** erfasst die Konsistenz bei automatisch umformulierten Anforderungen samt einer Konfidenzeinschätzung je Klassifikation.

Alle drei Prüfungen laufen vollautomatisiert über die gesamte Stichprobe und belegen die Zuverlässigkeit des Verfahrens, nicht die inhaltliche Richtigkeit einzelner Klassifikationsergebnisse.

Die vierte Dimension, konvergente Validität, misst einen Ausschnitt der inhaltlichen Korrektheit(Vgl. Campbell & Fiske, 1959, S. 81) und wird deshalb zweigeteilt konstruiert. Der genutzte Anker ist eine kleine, vorab und unabhängig vom Klassifikationsagenten kuratierte Teilmenge von schätzungsweise 15 bis 20 der rund 100 Stichprobenanforderungen mit manuell festgestellter, unstrittiger NIST-800-53-Äquivalenz. Da dieser Anker nicht vom

selben Agenten erzeugt wird, der auch die zu validierende Klassifikation liefert, prüft die Dimension eine externe inhaltliche Referenz statt interner Konsistenz zwischen zwei Ausgaben desselben Systems: ob die Klassifikation konsistent mit Existenz oder Nichtexistenz einer passenden Azure-Policy-Definition in Microsofts NIST-800-53-Zuordnung ist(Vgl. Microsoft, 2026). Diese binäre Existenzprüfung stützt vor allem die Abgrenzung zwischen nicht erfüllbar und den beiden übrigen Graden, die Unterscheidung zwischen vollständig und teilweise beruht zusätzlich auf der zuvor definierten Klassifikationsskala.

Die bereits bestehende Zuordnungstabelle zwischen ISO 27001 und dem älteren BSI-IT-Grundschutz bleibt daneben als zusätzlicher Sanity-Check erhalten, ist jedoch ausdrücklich auf inhaltliche Nähe zu ISO 27001 begrenzt und beansprucht keine Aussage über Automatisierbarkeit. Für den überwiegenden Teil der Stichprobe außerhalb dieser beiden Referenzpunkte existiert keine belastbare Ground Truth zur inhaltlichen Korrektheit, dieser Umstand begründet die vorangehend beschriebenen prozeduralen Prüfungen als eigenständige, davon unabhängige Evaluationsebene.

Die Evaluation stützt sich auf eine Stichprobe von rund 100 Grundschutz++-Anforderungen, etwa 10 Prozent der insgesamt rund 998 Anforderungen(Vgl. Bundesamt für Sicherheit in der Informationstechnik, 2025), die proportional geschichtet über alle 20 Practices gezogen wird und je Practice eine Mindestquote vorsieht, damit auch kleine Practices vertreten bleiben.

Gemeinsam ergeben die vier automatisierten Prüfdimensionen und der zweigeteilte Abgleich gegen Referenzstrukturen ein mehrdimensionales Bild der Klassifikationsgüte, das über eine einzelne Aggregat-Metrik hinausgeht und durch den Vergleich mit der nichtgenerativen Baseline zusätzlich abgesichert wird.

4 Literaturrecherche

Die Literaturrecherche verfolgt zwei Ziele: die in der Einleitung skizzierte Forschungslücke zu bestätigen und eng verwandte Arbeiten zur Abgrenzung des eigenen Vorhabens zu identifizieren. Eine systematische, booleschbasierte Suche mit dokumentierten Trefferzahlen sichert das Ergebnis ab.

Durchsucht wurden arXiv, EBSCO und Google Scholar (Tabelle 1) mit booleschen Operatoren (`AND`/`OR`) und Phrasensuche, EBSCO ergänzt einen aktivierbaren Peer-Review-Filter. Der Suchzeitraum ist bei arXiv nativ auf die letzten drei Jahre begrenzt (`submittedDate:[20230101 TO 20261231]`), bei EBSCO über das Startdatum 01/2023 mit aktiviertem Peer-Review-Filter, bei Google Scholar mangels Datumsparameter durch manuelle Prüfung des Publikationsdatums je Treffer. Ausgenommen sind die vier methodischen Grundlagenwerke zu Design Science Research (Vgl. Hevner et al., 2004, S. 75-105)(Vgl. Peffers et al., 2007, S. 45-77)(Vgl. March & Smith, 1995, S. 251-266)(Vgl. Venable, Pries-Heje & Baskerville, 2016, S. 77-89), die unabhängig vom Erscheinungsjahr gezielt über Autor und Titel gesucht wurden, da methodische Standardwerke dauerhaft zitierfähig bleiben.

Tabelle 1: Systematische Boolesche Suche nach Themencluster

Themencluster	Suchstring	Treffer gesamt	Relevant
OSCAL + KI-Verfahren	`(OSCAL OR "Open Security Controls Assessment Language") AND ("large language model" OR LLM OR "generative AI")` EBSCO (ohne KI-Zusatz): `"OSCAL" OR "Open Security Controls Assessment Language"`	arXiv 0 / GS 18 / EBSCO 35	5

Fortsetzung auf nächster Seite

Tabelle 1 (Fortsetzung)

Themencluster	**Suchstring**	**Treffer gesamt**	**Relevant**
Policy-as-Code + KI	`("policy as code" OR "compliance as code") AND ("large language model" OR LLM)`	arXiv 27 / GS 55 (40 eindeutig) / EBSCO 65	7
Azure-Policy-Enforcement	`("Azure Policy" OR "policy enforcement") AND ("large language model" OR automation) AND (cloud OR compliance)`	arXiv 21 / GS 58 (45 eindeutig)	1
Crosswalk / Control-Mapping	`(crosswalk OR "control mapping") AND (compliance OR "security controls") AND (automation OR "natural language processing")`	arXiv 1 / GS 76 (63 eindeutig)	3

Bei Google Scholar variiert die Rohtrefferzahl je Themencluster stark, Mehrfachnennungen (Preprint, Verlagsversion, Repository-Spiegel desselben Beitrags) wurden dedupliziert (eindeutige Trefferzahl in Klammern). Beim Cluster Azure-Policy-Enforcement stammte der überwiegende Teil der Treffer aus Zeitschriften ohne erkennbares Peer-Review-Verfahren (generische Titel, keine nachvollziehbare Methodik, auffällig hohe Publikationsfrequenz) und wurde konsequent ausgeschlossen, ein Hinweis auf die bislang begrenzte Bearbeitung der geplanten Azure-Policy-Erweiterung im durchsuchten wissenschaftlichen Diskurs. EBSCO wurde ergänzend nur für die beiden priorisierten Cluster OSCAL und Policy-as-Code durchsucht, beide Trefferzahlen liegen in einer mit arXiv und Google Scholar vergleichbaren Größenordnung. Die 16 dokumentierten Treffer bilden nicht die gesamte im Folgenden diskutierte Literaturbasis ab, ein Teil der Arbeiten wurde bereits vor dieser systematischen Suche während der Themenfindung identifiziert und fließt ergänzend in Tabelle 2 ein.

Die Recherche identifiziert zwei aktuelle Arbeiten, die LLM-Agenten zur Erzeugung von OSCAL-Artefakten einsetzen. Tabelle 2 stellt diese sowie weitere eng verwandte Arbeiten samt h-Index der Erstautor:innen (Semantic Scholar, Stand 16.08.2026) den eigenen Anforderungen gegenüber. Keine der Arbeiten adressiert die Kombination aus bereits strukturiertem OSCAL-Input, der Crosswalk-Klassifikation von Grundschutz++-Anforderungen gegen die Azure-Policy-Automatisierbarkeit und einer mehrdimensionalen, objektivierbaren Validierungsmethodik. Am nächsten kommt Muhammad et al. (2026) mit einer vergleichbaren Abbildung von Governance-Anforderungen auf technisch prüfbare Regeln, allerdings für das Deployment-Gating von KI-Systemen statt für einen Kontrollkatalog-Crosswalk(Vgl. Muhammad, Yow & Alsenan, 2026, S. 1759211). Die systematische Übersichtsarbeit von Herrera et al. (2026) bestätigt diese Lücke unabhängig, indem sie über 55 gesichtete Studien hinweg die neuro-symbolische Verifikation gegen Halluzinationsrisiken in Policy-as-Code-Compliance-Pipelines als offenen Punkt benennt(Vgl. Herrera et al., 2026, S. 85058-85074).

Tabelle 2: Eng verwandte Arbeiten, Abgrenzung zur eigenen Forschungslücke und h-Index der Erstautor:innen (Semantic Scholar, Stand 16.08.2026)

Literatur	Kernaussage / Abgrenzung zur eigenen Arbeit	h-Index
Muth/Margraf (2026b), Reverse Engineering Compliance (Muth & Margraf, 2026b)	Extrahiert legacy IT-Sicherheitskonzepte in einen Dokumentgraphen, exportiert schema-valides OSCAL. Fokus auf Verifikation statt Neugenerierung aus strukturierten Kontrollen, kein Multi-Cloud-Bezug.	2
Muth/Margraf (2026a), From Legacy Documentation to OSCAL (Muth & Margraf, 2026a)	MCP-Multi-Agent-Pipeline überführt Freitext-Systembeschreibungen in OSCAL-Artefakte (SSP, SAR). Anderer Input (Freitext statt strukturierte Kontrolle), kein Multi-Cloud-Crosswalk.	2

Fortsetzung auf nächster Seite

Tabelle 2 (Fortsetzung)

Literatur	**Kernaussage / Abgrenzung zur eigenen Arbeit**	**h-Index**
Romeo et al. (2025), ARPaCCino (Romeo et al., 2025)	LLM, RAG und Tool-Validierung generieren und korrigieren Rego-Regeln für Terraform-IaC iterativ. Setzt bestehende IaC-Pipeline voraus, kein OSCAL-Bezug.	3
Gupta/Sreenivasamurthy (2026), Prose2Policy (Gupta & Sreenivasamurthy, 2026)	LLM-Pipeline übersetzt Freitext-Zugriffsregeln in ausführbare Rego-Policies, 95,3 % Compile-Rate. Fokus auf Access-Control und Zero-Trust, kein Compliance-Katalog- oder OSCAL-Bezug.	3
Madan (2025), ArGen (Madan, 2025)	An Open Policy Agent angelehnter Layer steuert generative KI-Systeme selbst via GRPO und Reward-Scoring. Adressiert KI-Verhalten statt Infrastruktur, kein OSCAL-Bezug.	2
Tupsamudre et al. (2022) (Tupsamudre et al., 2022) / Agarwal et al. (2021) (Agarwal et al., 2021)	Klassisches beziehungsweise frühes KI-gestütztes Mapping zwischen Compliance-Dokumenten und Security-Controls. Keine generative LLM-Transformation, kein OSCAL-Zielformat, für Framework-Revisionen statt Erstgenerierung konzipiert.	3 / 12
Chowdhary et al. (2025), AutoPAC (Chowdhary et al., 2025)	LLM-gestützte Policy-zu-Code-Konvertierung mit Judge-LLM-Validierungsschritt. Kein OSCAL-Zielformat, kein Multi-Cloud-Bezug.	2

Fortsetzung auf nächster Seite

Tabelle 2 (Fortsetzung)

Literatur	**Kernaussage / Abgrenzung zur eigenen Arbeit**	**h-Index**
Tasew et al. (2026), Security Control Recommender (Tasew et al., 2026)	ML-Empfehlung passender NIST-SP-800-53-Controls aus operativen Compliance-Artefakten, Human-in-the-Loop-Prüfung, 84,18 % Top-1-Trefferquote. Empfiehlt Controls bottom-up statt Anforderungen top-down zu klassifizieren, kein OSCAL-Zielformat.	0
Muhammad et al. (2026), Audit-as-Code (Muhammad, Yow & Alsenan, 2026)	Bildet Governance-Anforderungen auf prüfbare Regeln ab, MLOps-/CI-CD-fokussiert mit Assured-Readiness-Score für automatisiertes Deployment-Gating. Zielt auf Deployment-Gating von KI-Systemen statt Kontrollkatalog-Crosswalk, kein Grundschutz++- oder Azure-Bezug.	3
Herrera et al. (2026), From Policy to Practice (Herrera et al., 2026)	PRISMA-2020-konforme SLR zu 55 Studien zu KI-/NLP-gestütztem Cybersecurity-Wissensmanagement, identifiziert eine Lücke bei neuro-symbolischer Verifikation gegen Halluzinationsrisiken. Übersichtsarbeit statt eigenes System, bestätigt die Forschungslücke unabhängig.	3
Voicu (2026), Validating LLM-Generated Control Mappings (Voicu, 2026)	Fünfdimensionale psychometrische Methodik validiert LLM-generiertes Mapping zwischen zwei Compliance-Frameworks, vergleichbare Aufgabenstellung. Generisch und framework-agnostisch, kein Bezug zu Grundschutz++, Azure Policy oder OSCAL, eigene Arbeit überträgt vier der fünf Dimensionen erstmals auf diese Domänenkombination.	n. v.

Weitere identifizierte, thematisch einschlägige Literatur zu einzelnen methodischen Detailfragen umfasst strukturierte Wissensextraktion aus Freitext via Zero-Shot-LLM(Vgl. Caufield et al., 2024), nutzerzentrierte Anforderungen an strukturierte LLM-Ausgaben(Vgl. Liu et al., 2024, S. 1-9), Integration von Kontrollkatalogen in Modellierungswerkzeuge(Vgl. Shaked, 2024, S. 264-277) sowie Prompt-Engineering-Strategien für sichere

Codegenerierung(Vgl. Bruni et al., 2025). Hinzu kommen OSCAL-gestützte Risikobewertung über Angriffsgraphen(Vgl. Koufos et al., 2025, S. 319-328), Compliance-as-Code für KI-gestützte Identitätssysteme(Vgl. Nweke & Yeng, 2026, S. 28258-28281), empirische Adoption von Policy-as-Code in Open-Source-Projekten(Vgl. Foalem et al., 2026, S. 113028) sowie ein konzeptioneller Ansatz zur deterministischen Policy-Durchsetzung auf KI-Ausgaben(Vgl. Smith & McCarthy, 2026, S. 394). Deren Erstautor:innen weisen einen h-Index zwischen 3 und 19 auf (Median 5), mit Ausnahme von Smith/McCarthy (2026), für den aus Namenskollisionsgründen kein Wert ermittelbar war.

Die h-Index-Werte in Tabelle 2 reichen von 0 bis 12 und liegen mehrheitlich im niedrigen einstelligen Bereich, nahezu alle zugrunde liegenden Arbeiten stammen aus den Jahren 2025 und 2026, was das geringe Alter dieses Forschungsfelds widerspiegelt. Der Wert 0 bei Tasew et al. (2026) bezeichnet einen gemessenen, bislang unzitierten h-Index und ist von einem nicht ermittelbaren Wert (n. v.) zu unterscheiden. Für Voicu (2026) liegt ein solcher nicht ermittelbarer Wert vor, da es sich um einen nicht peer-reviewten Blogbeitrag der Cloud Security Alliance ohne klassischen Publikationsindex handelt.

Über die tabellarische Abgrenzung hinaus liefert Voicu (2026) konkrete Referenzwerte für Bestandssysteme auf einer vergleichbaren Aufgabe, darunter eine schwache konvergente Validität und eine geringe Diskriminierungsgüte bei adversarisch konstruierten, mehrdeutigen Grenzfällen(Vgl. Voicu, 2026). Diese Referenzwerte begründen unmittelbar die für die eigene Arbeit übernommene mehrdimensionale Evaluationsmethodik, die sich nicht auf einzelne Aggregat-Metriken verlässt.

5 Vorläufige Gliederung der Arbeit

Die vorläufige Gliederung folgt dem funktionalen Schema des FOM-Leitfadens (Einleitung – Grundlagen – Umsetzung – Schluss).

1. **Einleitung** (12,5%)
    - Problemstellung und Motivation
    - Forschungsfrage und Zielsetzung
    - Methodisches Vorgehen (Design Science Research)
    - Abgrenzung und Aufbau der Arbeit
2. **Theoretische Grundlagen und Stand der Technik** (25,0%)
    - OSCAL und die BSI-Stand-der-Technik-Bibliothek Grundschutz++
    - Microsofts NIST-800-53-zu-Azure-Policy-Zuordnung als bestehendes Enforcement-Mapping
    - Grundlagen der Crosswalk- und Control-Mapping-Forschung
    - Systematische Literaturrecherche
3. **Methodik** (Teil von „Umsetzung“, 48,5%)
    - Design-Science-Research-Vorgehen im Detail, DSRM-Phasenmodell und FEDS-Evaluationsstrategie
    - Anforderungsanalyse an das Artefakt
4. **Projektumsetzung** (Teil von „Umsetzung“, 48,5%)
    - Artefaktkonstruktion: Crosswalk-Klassifikationsagent für Grundschutz++-Anforderungen gegen Azure-Policy-Automatisierbarkeit
    - Minimalstufe (verpflichtend): Kernagent als LLM-Wrapper mit strukturierter Ausgabe, Pflicht-Baseline (TF-IDF-/Embedding-Similarity mit definierter Score-zu-Klassifikationsgrad-Übersetzung), die vier automatisierten Evaluationsdimensionen auf der vollständigen Stichprobe
    - Ausbaustufe (optional, bei ausreichendem Zeitrahmen): erweiterte Agenten-Architektur (RAG über BSI-/NIST-Katalog, iterative Selbstkorrektur, deterministische Regelprüfschicht), größere Stichprobe, vollständige Azure-Demonstration

- Demonstration über exemplarische Policy-Zuweisung beziehungsweise GitHub-Issue-Erstellung
- Mehrdimensionale Evaluation der Klassifikationsgüte

5. **Bewertung und Diskussion** (Teil von „Schluss“, 14,0%)
    - Kritische Würdigung der Ergebnisse
    - Grenzen des Ansatzes
6. **Fazit und Ausblick** (Teil von „Schluss“, 14,0%)
    - Beantwortung der Forschungsfrage
    - Ausblick auf zukünftige Forschung

6 Zeit- und Projektplanung

Die Bearbeitung beginnt am 01.11.2026 und endet mit der Abgabe am 01.04.2027, insgesamt rund fünf Monate. Die Implementierung erhält dabei den größten Zeitanteil, entsprechend dem mit 48,5 % dominierenden Umsetzung-Block des zugrunde liegenden Gliederungs-Blueprints.

Tabelle 3: Projektplan und Zeitplan der Masterthesis

Zeitraum	**Arbeitsschritte / Meilensteine**	**Bemerkungen / Puffer**
01.11.2026 – 30.11.2026	• Abschluss der Literaturrecherche und Konzeptualisierung • Detaillierte Anforderungsanalyse an das Artefakt • Festlegung der Stichprobenziehung: rund 100 Grundschutz++-Anforderungen, proportional geschichtet über alle 20 Practices mit Mindestquote je Practice	• Keine externe Abhängigkeit in dieser Phase, dient zugleich als zeitlicher Puffer für die nachfolgenden Phasen
01.12.2026 – 31.01.2027	• Entwurf und prototypische Implementierung des Kernagenten (LLM-Wrapper mit strukturierter Ausgabe) sowie der Pflicht-Baseline (TF-IDF-/Embedding-Similarity mit definierter Score-zu-Klassifikationsgrad-Übersetzung) • Bei ausreichendem Zeitrahmen: Ausbaustufe mit RAG über BSI-/NIST-Katalog, iterativer Selbstkorrektur und deterministischer Regelprüfschicht	• Fallback bei knappem Zeitrahmen: Abschluss mit der Minimalstufe, Kernevaluation bleibt davon unberührt

Fortsetzung auf nächster Seite

Tabelle 3 (Fortsetzung)

Zeitraum	**Arbeitsschritte / Meilensteine**	**Bemerkungen / Puffer**
01.02.2027 – 05.03.2027	• Demonstration über exemplarische Policy-Zuweisung beziehungsweise GitHub-Issue-Erstellung • Durchführung der vier automatisierten Evaluationsdimensionen (Konsistenz, konvergente Validität, adversarische Diskriminierung, Paraphrasen-Stabilität) auf der vollständigen Stichprobe • Iterative Verbesserung des Artefakts	• Fallback bei fehlendem Zugriff auf eine geeignete Azure-Testumgebung: simulierte beziehungsweise dokumentierte Demonstration ohne echte Policy-Zuweisung • Fallback bei Nichtverfügbarkeit oder Änderung von Microsofts NIST-800-53-zu-Azure-Policy-Zuordnung: Rückgriff auf den zuletzt dokumentierten Stand der Zuordnung
06.03.2027 – 01.04.2027	• Ausarbeitung der schriftlichen Masterthesis • Überarbeitung, finale Anpassungen, Abgabe	• Zeitpuffer für Korrekturschleifen mit dem Betreuer

Literaturverzeichnis

Agarwal, V., Bar-Haim, R., Eden, L., Gupta, N., Kantor, Y., & Kumar, A. (2021). AI-Assisted Security Controls Mapping for Clouds Built for Regulated Workloads, 136–146. https://doi.org/10.1109/cloud53861.2021.00027

Campbell, D. T., & Fiske, D. W. (1959). Convergent and Discriminant Validation by the Multitrait-Multimethod Matrix. *Psychological Bulletin*, *56*(2), 81–105.

Caufield, J. H., Hegde, H., Emonet, V., Harris, N. L., Joachimiak, M. P., Matentzoglu, N., Kim, H., Moxon, S., Reese, J. T., Haendel, M. A., Robinson, P. N., & Mungall, C. J. (2024). Structured Prompt Interrogation and Recursive Extraction of Semantics (SPIRES): A Method for Populating Knowledge Bases Using Zero-Shot Learning. *Bioinformatics*, *40*(3). https://doi.org/10.1093/bioinformatics/btae104

Chowdhary, N., Dutta, T., Chattopadhyay, S., & Chakraborty, S. (2025). AutoPAC: Exploring LLMs for Automating Policy to Code Conversion in Business Organizations, 667–675. https://doi.org/10.1109/comsnets63942.2025.10885751

Foalem, P. L., Silva, L. D., Khomh, F., & Merlo, E. (2026). An Empirical Study of Policy-as-Code Adoption in Open-Source Software Projects. *Journal of Systems and Software*, *242*, 113028. https://doi.org/10.1016/j.jss.2026.113028

Hase, R., Wang, Y., Koike-Akino, T., Liu, J., Parsons, K., & Hato, J. (2026). Evaluating Security Policy Compliance in Infrastructure as Code Generated by Large Language Models. *2026 14th International Symposium on Digital Forensics and Security (ISDFS)*, 1–6. https://doi.org/10.1109/isdfs69419.2026.11458930

Herrera, L.-C., Janulevičius, J., Pinto, A., & Donoso, Y. (2026). From Policy to Practice: A Systematic Literature Review of AI-Driven Knowledge Base Construction for Cybersecurity Compliance. *IEEE Access*, *14*, 85058–85074. https://doi.org/10.1109/ACCESS.2026.3700565

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design Science in Information Systems Research. *MIS Quarterly*, 75–105.

Koufos, I., Christopoulou, M., Xilouris, G., Kourtis, M.-A., Souvalioti, M., & Trakadas, P. (2025). Towards the Automation of Attack Graph-Based Risk Assessment with OSCAL. Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-76459-2_30

Liu, M. X., Liu, F., Fiannaca, A. J., Koo, T., Dixon, L., Terry, M., & Cai, C. J. (2024). „We Need Structured Output“: Towards User-Centered Constraints on Large Language Model Output, 1–9. https://doi.org/10.1145/3613905.3650756

March, S. T., & Smith, G. F. (1995). Design and Natural Science Research on Information Technology. *Decision Support Systems*, *15*(4), 251–266. https://doi.org/10.1016/0167-9236(94)00041-2

Martinez, C., Etxaniz, I., Molinuevo, A., & Alonso, J. (2024). MEDINA Catalogue of Cloud Security Controls and Metrics: Towards Continuous Cloud Security Compliance. *Open Research Europe*, *4*, 90. https://doi.org/10.12688/openreseurope.16669.1

Muhammad, A. E., Yow, K.-C., & Alsenan, S. (2026). Audit-as-Code: A Policy-as-Code Framework for Continuous AI Assurance. *Frontiers in Artificial Intelligence*, *9*, 1759211. https://doi.org/10.3389/frai.2026.1759211

Nweke, L. O., & Yeng, P. K. (2026). Compliance-as-Code for AI-Driven Identity Systems: Clause-to-Control Traceability and Machine-Readable Evidence. *IEEE Access*, *14*, 28258–28281. https://doi.org/10.1109/access.2026.3665991

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems*, *24*(3), 45–77. https://doi.org/10.2753/mis0742-1222240302

Romeo, F., Arena, L., Blefari, F., Pironti, F. A., Lupinacci, M., & Furfaro, A. (2025). ARPaC-Cino: An Agentic-RAG for Policy as Code Compliance. https://doi.org/10.1007/978-3-032-05727-3_39

Shaked, A. (2024). Facilitating the Integrative Use of Security Knowledge Bases within a Modelling Environment. *Journal of Cybersecurity and Privacy*, *4*(2), 264–277. https://doi.org/10.3390/jcp4020013

Smith, C. B., & McCarthy, D. (2026). Deterministic Governance for Generative Systems: A Policy-Aligned Runtime for Validating AI Outputs. *AI and Ethics*, *6*(4), 394. https://doi.org/10.1007/s43681-026-01172-6

Tasew, J. M., Samal, P., Pakshad, P., Omar, M., & Dawson, M. E. (2026). A Security Control Recommender Approach with Human in the Loop for Cybersecurity Compliance Auditing. *Discover Networks*, *2*(1), 19. https://doi.org/10.1007/s44354-026-00033-2

Tupsamudre, H., Kumar, A., Agarwal, V., Gupta, N., & Mondal, S. (2022). AI-Assisted Controls Change Management for Cybersecurity in the Cloud. *Proceedings of the AAAI Conference on Artificial Intelligence*, *36*(11), 12629–12635. https://doi.org/10.1609/aaai.v36i11.21537

Venable, J., Pries-Heje, J., & Baskerville, R. (2016). FEDS: A Framework for Evaluation in Design Science Research. *European Journal of Information Systems*, *25*(1), 77–89. https://doi.org/10.1057/ejis.2014.36

Internetquellen

Bruni, M., Gabrielli, F., Ghafari, M., & Kropp, M. (2025, 9. Februar). *Benchmarking Prompt Engineering Techniques for Secure Code Generation with GPT Models*. arXiv: 2502.06039 [cs.SE]. Verfügbar 2026-08-11 unter https://arxiv.org/abs/2502.06039

Bundesamt für Sicherheit in der Informationstechnik. (2025). *Stand der Technik: OSCAL*. Verfügbar 2026-08-14 unter https://www.bsi.bund.de/DE/Themen/Unternehmen-und-Organisationen/Standards-und-Zertifizierung/Grundschutz-in-der-Informationssicherheit/Stand-der-Technik/OSCAL/oscal.html

Gupta, V., & Sreenivasamurthy, D. (2026, 16. März). *Prose2Policy (P2P): A Practical LLM Pipeline for Translating Natural-Language Access Policies into Executable Rego*. arXiv: 2603.15799 [cs.AI]. Verfügbar 2026-08-08 unter https://arxiv.org/abs/2603.15799

Madan, K. (2025, 6. September). *ArGen: Auto-Regulation of Generative AI via GRPO and Policy-as-Code*. arXiv: 2509.07006 [cs.CY]. Verfügbar 2026-08-08 unter https://arxiv.org/abs/2509.07006

Microsoft. (2026). *Regulatory Compliance Details for NIST SP 800-53 Rev. 5 - Azure Policy*. Verfügbar 2026-08-14 unter https://learn.microsoft.com/en-us/azure/governance/policy/samples/nist-sp-800-53-r5

Muth, L. R., & Margraf, M. (2026a, 9. Juli). *From Legacy Documentation to OSCAL: An MCP-Based Agent Pipeline for Threat-Informed Continuous Compliance in Critical Infrastructure*. arXiv: 2607.08288. Verfügbar 2026-08-11 unter https://arxiv.org/abs/2607.08288

Muth, L. R., & Margraf, M. (2026b, 9. Juli). *Reverse Engineering Compliance: A Dual-Graph Verification Framework for Auditing Legacy IT Security Concepts*. arXiv: 2607.08292. Verfügbar 2026-08-11 unter https://arxiv.org/abs/2607.08292

Voicu, L. (2026, 2. Juli). *Validating LLM-Generated Control Mappings Beyond Aggregate Accuracy*. Verfügbar 2026-08-14 unter https://cloudsecurityalliance.org/blog/2026/07/02/validating-llm-generated-control-mappings-beyond-aggregate-accuracy

KI-Hilfsmittelverzeichnis

Tool	Beschreibung	Einsatzbereich	Version
Claude (Anthropic)	KI-gestützte Literaturrecherche (über paper-search-mcp), Textentwurf und -überarbeitung, LaTeX-Strukturierung	Recherche, Texterstellung, Faktencheck gegen Primärquellen	Claude Sonnet 5

Eigenständigkeitserklärung

Hiermit versichere ich, dass ich die angemeldete Prüfungsleistung in allen Teilen eigenständig ohne Hilfe von Dritten anfertigen und keine anderen als die in der Prüfungsleistung angegebenen Quellen und zugelassenen Hilfsmittel verwenden werde. Sämtliche wörtlichen und sinngemäßen Übernahmen inklusive KI-generierter Inhalte werde ich kenntlich machen. Diese Prüfungsleistung hat zum Zeitpunkt der Abgabe weder in gleicher noch in ähnlicher Form, auch nicht auszugsweise, bereits einer Prüfungsbehörde zur Prüfung vorgelegen; hiervon ausgenommen sind Prüfungsleistungen, für die in der Modulbeschreibung ausdrücklich andere Regelungen festgelegt sind. Mir ist bekannt, dass die Zuwiderhandlung gegen den Inhalt dieser Erklärung einen Täuschungsversuch darstellt, der das Nichtbestehen der Prüfung zur Folge hat und daneben strafrechtlich gem. § 156 StGB verfolgt werden kann. Darüber hinaus ist mir bekannt, dass ich bei schwerwiegender Täuschung exmatrikuliert und mit einer Geldbuße bis zu 50.000 EUR nach der für mich gültigen Rahmenprüfungsordnung belegt werden kann. Ich erkläre mich damit einverstanden, dass diese Prüfungsleistung zwecks Plagiatsprüfung auf die Server externer Anbieter hochgeladen werden darf. Die Plagiatsprüfung stellt keine Zurverfügungstellung für die Öffentlichkeit dar.

Bielefeld, 18.8.2026
(Ort, Datum)

Philipp Rose
(Eigenhändige Unterschrift)