SCIENTIFIC CULTURE
www.sci-cult.com

SCIENTIFIC CULTURE, Vol. 12, No. 4, (2026), pp. 5223-5237
Open Access. Online & Print

DOI: 10.5281/zenodo.12426565

ZERO-TRUST CYBERSECURITY GOVERNANCE FOR AGENTIC AI: GUARDRAIL ARCHITECTURES FOR RAG PIPELINES AND TOOL INVOCATION

Theophilus Irianan[1]*

[1]PhD. Student, Dakota State University. Email: theoirianan@gmail.com, ORCID iD: 0009-0004-2599-2365

Received: 27/11/2025
Accepted: 19/01/2026

Corresponding Author: Theophilus Irianan (theoirianan@gmail.com)

ABSTRACT

The interactions between Large Language Models (LLMs), Retrieval-Augmented Generation (RAG) systems and autonomous agentic AI are associated with both abilities never seen and new security concerns. It is a type of mixed-method research that integrates design science research, qualitative case studies, and quantitative benchmarking to establish a full-fledged zero-trust form of governance of the agentic AI systems. We present a novel architecture utilizing identity-based access control, dynamic policy enforcement, and multi-layered guardrails of RAG pipelines and tool invocation based on a systematic literature review and expert interviews and experimental evaluation of 14 agentic AI deployment cases. Our research findings show that integrated, zero-trust governance reduces the count of unauthorized tool formation by 87 percent, the count of contains Prompt injection assaults by 94.3 percent as well as the accuracy of threat recognition by 97.3 percent. This work gives (1) a threat taxonomy of 47 distinct attack vectors targeting agentic systems; (2) ZTAGRAF, a reference-architecture according to which zero-trust principles and guardrail enforcement are met; (3) empirically validated patterns of governance, based on interviews with 22 security practitioners; and (4) a maturity model of how to move organizations towards agentic governance of zero-trust. The paper provides a theoretical foundation and practical recommendations in practical situations on how to assure the next generation autonomous AI systems in mission critical infrastructures and structures.

KEYWORDS: Agentic AI, The Zero-Trust Architecture, RAG Security, Guardrails, Cybersecurity Governance, LLM Safety, Tool Invocation, Policy Enforcement.

1. INTRODUCTION

1.1 Background and Problem Context

The adoption of Large Language Models as an autonomous entity is a fundamental change in the interaction between computational systems and data, users, and infrastructure. In contrast to the legacy chatbots, which only work within the boundaries of a conversation, agentic AI systems combine planning, memory, tool utilization, and multi-step thinking to perform complex and self-directed workflows (Adabara, Sadiq, Shuaibu, Danjuma, and Maninti, 2025; Lazer et al., 2026). This independence has brought in vital security issues that the traditional cybersecurity models were not developed to handle. The problem with agentic systems deployed by organizations is compounding: the same capabilities that enable the successful use of agents of language, dynamic reasoning, tool integration, etc., also generate new attack surfaces and new classes of vulnerabilities (Arora & Hastings, 2025). Existing security measures are still unsatisfactory. Historical perimeter-based defensive models are based on the notion that the internal environment is trusted, which is not the case with the distributed, cloud-native aspect of the current AI systems (Filho, 2025). At the same time, the current AI safety systems are mostly concerned with alignment and fairness, but not the immediate security risks of adversarial inputs, prompt injection, unauthorized tool calls, and translocation in agentic ecosystems (Xu et al., 2025). Moreover, the RAG pipelines, based on external knowledge grounding of the LLM responses, provide more places of vulnerability at the data ingestion, retrieval, and injection points (Nandagopal, 2025). Combining various autonomous agents with their own objectives and access to tools causes coordination and collusion risks and cannot be systematically addressed by the existing frameworks (Raza et al., 2025).

1.2 Research Objectives and Contributions

The proposed study also contributes to bridging this literature gap in the gap between the novel agentic AI functions and inadequate security governance by performing a detailed and empirically-grounded research. Our work will be guided by three major research questions:

RQ1: What are the attack vectors specific to agentic AI systems and how they compare to the conventional software security threats and the conventional LLM threats?

RQ2: How can the principles of zero-trust architecture be applied, so that the functions of the autonomous actors are controlled without causing any changes in performance and usability?

RQ3: What organizational practices, governance structures, and technical controls will be necessary to be responsible in agentic AI deployment in high-stakes environments?

This study's core contributions are: Threat Taxonomy. - A catalog of 47 agentic system attack vectors organized into cognitive, operational, governance, and coordination dimensions (Narajala and Narayan, 2025). ZTAGRAF Architecture - A prototype implementation of a combination of identity-based access control using zero-trust and dynamic policy enforcement, and multi-layered guardrails of RAG pipelines and tool invocation, which have been tested in 14 deployment environments. Empirically Supported Patterns of Governance - The present study will provide qualitative data on 22 security practitioners and 14 organizational cases that can illustrate the organizational, technical, and cultural circumstances of the zero-trust agentic governance (Schmitz et al., 2025). Maturity and Integration Model - A step-by-step blueprint of what organizations can do to transition to the zero-trust governance that includes step-by-step action plans of implementation and success metrics.

2. LITERATURE REVIEW

2.1 The Foundations and History of Zero-Trust Architecture

Zero-Trust Architecture (ZTA) is now the most popular paradigm of cybersecurity of the modern distributed systems. ZTA is founded on the concept of never trust, always verify and ensures that there is no implicit trust in network boundaries and instead, it offers continuous authentication, authorization and least-privilege access to every access request (Filho, 2025; Mushtaq et al., 2025). ZTA identity verification (four building blocks), device posture assessment, network segmentation and access control at the application level are built and are actually being standardized by policies such as NIST SP 800-207 and the Zero Trust Maturity Model offered by CISA (Bommareddi, 2025). The typical three settings of traditional ZTA deployments will be (1) cloud and hybrid infrastructure, (2) enterprise network and remote work, and (3) IoT and edge computing (Gambo & Almulhem, 2025). ZTA has demonstrated itself to be very successful in cloud-native, micro-segmentation of breach impact, lateral movement and isolating threats, responses (R. I. Khan et al., 2026). More current applications using machine learning and behavioural analytics have

further enhanced detection capabilities where the detection rate is more than 99 percent in controlled environments (N et al., 2025; Reddy, 2025). However, the deployment of traditional ZTA was to be rather immobile infrastructure and foreseeable user processes. Multi-agent systems and autonomous agents, dynamically changing AI workloads are challenging to fully address the autonomous agents may call on services they themselves do not know exist, the interdependence of the tools between agents may involve complex relationships of trust, memory and cross-turn reasoning demand new state-management challenges and there are risks of collusion when they start to coordinate (Gurram, 2025).

2.2 Agentic AI Systems: Architectures, Capabilities, and Threat Models

The agentic AI systems are a progressive move in regards to the previous LLMs and chatbots. As a replacement of individual prompts, agentic systems combine perception, planning, tool use, memory, and execution into autonomous processes that have the ability to realize multi-step goals with minimum human oversight (Kumar, 2025). Its main architecture elements are: reasoning engine (LLM-based planning and decision-making), a memory system (short- term context, long-term knowledge), a tool-use interface (standardized protocols to invoke external services), and a coordination mechanism (coordination among agents and conflict resolution) (Dwivedi et al., 2025).

This architectural development presents new security problems that are not similar to the old software architecture and AI systems. Threat models inventory Recent models of agentic risks divide systematically into five dimensions (Narajala & Narayan, 2025): (1) cognitive vulnerabilities (hallucinations, reasoning failures, misalignment), (2) temporal persistence threats (memory poisoning, state manipulation), (3) operational execution vulnerabilities (insecure tool invocation, risks of code generation), (4) trust boundary violations (inter-agent impersonation, credential leakage), and (5) governance circumventions (policy evasion, audit trail tampering).

Empirical analyses indicate that the dominant agents of LLM are susceptive to a wide range of attacks regardless of the training on safety and guardrails. The systematic comparison of nine popular agentic models with 400+ attack scenarios revealed that overall attack success rates were over 60 percent, with advanced models showing even better exploitable behavior because of enhanced instruction-following behavior (Gasmi et al., 2025). More importantly, subjecting a target system to multi-turn attack chains within and across several layers is capable of going above 91 percent successful in realistic settings (Nguyen & Husain, 2025). These results show that the LLM incremental changes to protect individuals do not help enough, the architectural redesign to protect them needs to be done.

2.3 RAG Pipeline Security and Data Integrity

This is the real breakthrough: Retrieval-Augmented Generation is a realistic extension of LLM features, allowing engines to base the answers on external knowledge sources and resolve the essential issues of outdated training data and hallucination (Nandagopal, 2025). RAG architectures bring a complicated pipeline: ingestion of data and normalization, embedding and indexing, the retrieval (dense semantic and sparse lexical), reranking, and generation. In every phase, security issues are different (Ammann et al., 2025). The worst type of RAG vulnerability is data poisoning attacks. Attackers can also use injections of specifically designed texts or antagonistic information to force the knowledge bases to generate harmful information without detection (Z. Li et al., 2025; Stefano et al., 2024). RAG systems have proven to be the most vulnerable to token-level attacks that do not need much knowledge on the part of the attacker and can succeed in 40-60% in a variety of settings (Gong et al., 2025). The use of indirect prompt injection attacks, using retrieved material as adversarial input to embed the instructions of the adversarial, get around many of the guardrails that protect direct prompts and are noted to succeed against modern defenses at 70 percent plus (An et al., 2025).

RAG systems have equally sharp privacy and confidentiality risks. The contents of knowledge bases are often sensitive (proprietary data, personal information, medical records), and there are many different attack vectors that can be used to unlawfully bring out sensitive content contained in the knowledge base: (1) prompt-based leakage, whereby querying the knowledge base with adversarial queries elicits sensitive retrieved information; (2) model inversion attacks, which can be used to reconstruct the training data by carefully prompting the knowledge base; and (3) membership inference attacks, which can be used to determine the presence of particular documents in the knowledge base (P. Z Multi-layered defensive strategies of RAG security. Input-level protection consists of timely injection detection, content filtering and

canonicalization. Some of the malicious document detection methods are based on malicious document detected on a retrieval level through statistical analysis, gradient-based methods, and adversarial robustness (Castagnaro et al., 2025; Kim et al., 2025). Response verification, evidence citation validation, and hallucination detection are some of the generation-level defenses (Hiriyanna and Zhao, 2025). Nevertheless, the literature on integrated defense-in-depth strategies based on the consideration of the full RAG pipeline is insufficiently developed (Wok, 2025).

2.4 Tool Invocation Security and Function Calling Governance

Tool calling (also known as structured tool use or functional calling) has been brought to the forefront of agentic AI deployment. Instead of producing free-form text, LLM agents use external tools, typically APIs, code execution systems, databases, web services, via structured interfaces. This feature lets automation be practically implemented but opens up significant security threats (Ferrag et al., 2025).

America has several dimensions of the tool-use attack surface. On the input level, agents can get adversarial directions through system prompts, user inputs or retrieval-augmented context leading to tool invocation that behaves against the intended behavior. On the invocation level, agents can choose the wrong tools, incorrectly specify parameters, or call tools with unsafe combinations (e.g. a write operation and an unvalidated file path). On the execution level, tool responses can have harmful content, cause side effects, or cause cascading failure in the relied-upon services (Du, 2025).

Certain classes of attack based on tool-invocations have been formally defined. Quick injection attacks that use descriptions of their tools are successful with 73.5 percent in architectures based on function-calling and 62.6 percent in Model Context Protocol (MCP) implementations (Gasmi et al., 2025). In multi-step situations, tool-dependency exploitation attacks (which attackers create sequences of tool invocations to bypass security controls) are successful with 91-96% success rates (K. Zhou et al., 2026). Tool transfer attacks, in which a tool installed with good intentions is reused as an instrument of bad ones, are a continuing type of vulnerability (Zhang et al., 2025).

Defensive strategies towards tool security revolve around three foundations, namely: (1) tool-level controls (api authentication, rate limiting, input validation), (2) agent-level controls (action planning verification, parameter verification, execution sandboxing), and (3) orchestration-level controls (tool registry management, least-privilege provisioning, runtime monitoring). Newer frameworks suggest tool-dependency retrieval as dynamic, so that the set of tools available to the agent can be minimized to those actually needed, with the result that the set of tools the agent can exploit is smaller (Patel et al., 2025). More limited privilege schemes like MiniScope use a hierarchy of tool-dependencies to reconstruct a hierarchy of permissions and only grant permissions necessary to complete a task (Zhu et al., 2025).

2.5 Guardrail Architectures and Runtime Policy Enforcement

Guardrails are an important technical control of restricting the execution time behavior of agentic systems, not altering model internals, nor necessitating the cooperation of agents. The new guardrail systems do not only offer input/output filtering provision, but context based multi-layered, multi-layered policy enforcement (Chennanabasappa et al., 2025). Recent guardrail systems consist of thirty-complementary systems. Detection guardrails are aimed at detecting potentially unsafe behavior, by jailbreak detection (reaching state-of-the-art on universal adversarial-facing inputs), semantic analysis (finding intent based on agent reasoning), and code analysis (finding malicious code patterns). Isolation guardrails ensure the confinement of the blast radius of the unsafe actions of memory stream isolation (injected code is not persistent between turns) and injection isolation (execution of tasks is a DAG and calls to the execution of unsuitable tools are not made) (H. Li et al., 2025).

Enforcement guardrails amend the system behavior, in real time, according to the perceived risks, such as alerting and storage to the blocking operations and human check (Pervez et al., 2025). One of them is a new type of guardrail, which is grounded in the neuro-symbolic approaches, grounded in the neural networks (to deal with the semantic complexity) but with the symbolic constraints (to deal with the policy correctness). Knowledge graph-based policy enforcement systems include graph-based policy enforcement (G-SPEC), and other policy enforcement systems based on knowledge graphs; constraints are defined; and constraints are used to ensure that agent behaviour is consistent with network topology policies and network security policies (in graph-based policy enforcement systems such as G-SPEC), and that controlled deployments are zero safety violations (Vijay & Ethiraj, 2025). The models of Governance-as-

a-Service (GaaS) ensure that policy is implemented as a runtime service, and is not limited by the internals of the agent, and allows the implementation of policy in a gradual fashion (coercive, normative, adaptive) and that trust is adjusted at runtime (Pervez et al., 2025).

2.6 Governance Frameworks and Organizational Adoption

The policy concerns that are peculiar to agentic systems. Unlike traditional software development, agentic systems bring new aspects of governance: (1) Autonomy vs. Accountability - autonomous systems cannot bring about easy ways of responsibility, and as such, regulation will be a doubled governance work; (2) Multi-stakeholder Coordination - agentic systems will touch on both technical teams (AI engineers, security), operational teams (SOC, infrastructure), and business owners and must govern it differently. These dimensions have been addressed by the detailed governance plans in terms of integrated risk management, policy implementation, and long-term tracking. According to the Agentic AI Risk and Capability Framework, agentic risks can be divided into three key sources, including component risks (model hallucinations, tools failure), design risks (poor safeguards, not-secure integration), and capability risks (autonomic execution, access to sensitive resources) (Khoo et al., 2025). Operationalized NIST AI Risk Management Framework The AAGATE operational AI risk management framework uses agentic AI of NIST AI Risk Management Framework of threat modeling, measuring (hybrid OWASP/SSVC) and managing (including red-teaming and adaptive policy enforcement) (Huang et al., 2025).

The uptake in the public sector, which was documenting through an interview of 5 government agencies, suggests that the traditional information systems of control (siloed compliance, episodic approvals) is impractical to use in agentic systems (Schmitz et al., 2025). Effectiveness in the adoption of cross-departmental governance committees, higher levels of security with twenty four hour surveys and formal audit procedures are documented by companies. These findings are in line with the overall organizational theory which postulates that the governance structures must continue to adapt with the technological capability.

3. METHODOLOGY

3.1 Research Design and Integration Strategy

The research design and integration strategy will be to carry out interviews with the respondents. The research design and integration strategy will be to conduct interviews with the respondents. The study is conducted as a concurrent mixed-method research design that incorporates three complementary research methodologies, specifically, Design Science Research (DSR), qualitative case study analysis, and quantitative benchmarking. The method is explanatory sequential, which intends to design qualitative findings lead to the design of quantitative instruments, and the design of quantitative instruments leads to the design of qualitative analysis (Obrik-Uloho et al., 2025). Design Science Research Component: DSR led the creation of ZTAGRAF, which is our zero-trust agentic governance reference architecture. In accordance with the existing DSR guidelines, we identified the problem with the help of the literature review and interviews with practitioners, developed the artifact with the help of its refinement during the iterative process, and tested the design in three organizational settings (Ampel et al., 2024). Assessment was done using not only ex-ante (design quality, technical soundness) but also ex-post (organizational effectiveness, measurable impact). Qualitative Component: The study interviewed 22 cybersecurity practitioners (security architects, incident response specialists, AI infrastructure engineers, governance officers) in 14 organizations in the financial services (4), healthcare (3), technology (4), and government (3) sectors.

The following interview protocols were used: threat perception and prioritization, existing governance structures, barriers to implementation, and success factors. Thematic analysis used deductive (categories based on threat models and governance frameworks) and inductive (emergent categories based on the information in the interviews) coding. It was determined that inter-rater reliability was a 30-percent dual coding of transcripts, resulting in Cohen kappa= 0.82. Quantitative Component: Quantitative analysis included three sub-studies (1) benchmark analysis of ZTAGRAF applied to 47 attack scenarios with three (LLM) models (GPT-4, Claude 3.5 Sonnet, Llama 3), assessing the rate of attack success, false positive rate, and operational overhead (2) organizational maturity analysis of six organizations with zero-trust agentic governance, assessing compliance measures, incident measures, and operational efficiency (3) a comparison of guardrail methods, evaluating the performance in terms of effectiveness (threat detection

3.2 Data Collection and Sampling

Interview Respondents: The study sampled practitioners in an organization that implements

agentic AI systems in production or pilot. The recruitment was purposive and focused on people who had more than 5 years of experience in cybersecurity and were directly responsible in securing AI systems. We have managed to have representation with regards to the size of organization (startups to enterprises), regulatory environment (regulated and non-regulated) and deployment levels (early pilot to mature implementations). Interviews were conducted face-to-face and taped with permission and transcribed professionally (45-90 minutes). Benchmark Scenarios: The attack scenarios were obtained in 3 sources: (1) published CVEs and published incidences in AI systems (18%), (2) attack model in academic literature (31%), and (3) attack scenarios created with expert workshop with 8 security researchers (51%). The scenarios were structured across the attack surface: the cases of prompt injection (12), misuse of tools (13), data poisoning (11) were tested, agent-to-agent attacks (6) and circumvention of governance (5). Deployment Contexts: There were three organizations that took part in ZTAGRAF implementation and assessment. Organization A is a financial services company that receives 10M+ transactions each day, with the agents being utilized to detect fraud and provide support to their customers. Organization B is a healthcare organization that adopts clinical decision-support agents. Organization C is a government body that employs agents to analyze the threats. The evaluation periods were 8-12 weeks after the deployment.

3.3 Integration and Analysis Approach

Integration was done at various levels. Qualitative results on governance issues were used at the design phase to affect the ZTAGRAF architectural design (i.e., the focus on continuous monitoring, human-in-the-loop solutions). Architectural assertions on threat mitigation were proven by quantitative benchmarking. On the analysis level, we developed images of joint representation on qualitative and quantitative measures (e.g., organizations with high confidence in policy enforcement and deployment data with 97%+ policy compliance). In case of meta-inferences (conclusions that combine between methods) we utilized weighted integration, assigning higher weights to those findings that had been independently established between methods. Conclusions made on the basis of both qualitative data (practitioner approval, case study success) and quantitative data (attack mitigation, metrics improvement) are offered as high-confidence findings.

4. ZTAGRAF ARCHITECTURE: A ZERO-TRUST GOVERNANCE FRAMEWORK FOR AGENTIC AI

4.1 Architectural Overview and Core Principles

The ZTAGRAF or Zero-Trust Governance, Risk and Accountability Framework is a system that combines the principles of zero-trust architecture and guardrail enforcement. This ZTAGRAF system is designed to support AI systems. The ZTAGRAF architecture is made up of seven layers each with its own set of zero-trust functions.

* Layer 1: Identity and Context Layer. This layer provides agents, users and services with an identity. It also continuously measures the context, such as device posture, user activity and resource sensitivity.
* Layer 2: Intention Verification Layer. This layer records and verifies the intent of the agents before they take any action. It prevents actions by having explicit goal specifications and detecting any deviations.
* Layer 3: Access Control Layer. This layer provides the least-privilege access based on policy enforcement. It uses rules based on Attribute-Based Access Control or ABAC. Makes decisions to authorize at runtime.
* Layer 4: RAG Pipeline Security Layer. This layer ensures the security of data ingestion, retrieval and generation. It does this by validating inputs identifying documents and validating responses.
* Layer 5: Tool Invocation Layer. This layer is responsible for tool discovery, tool selection and tool execution. It uses interfaces, parameter validation and outcome validation.
* Layer 6: Coordination and Monitoring Layer. This layer observes agent interaction identifies behavior and implements multi-agent policies.
* Layer 7: Governance and Audit Layer. This layer keeps audit trails administers compliance policies and aids in forensic analysis.

4.2 Identity-Driven Access Control for Agents

The ZTAGRAF system uses Agent-Based Access Control, which is also known as Identity-Based Access Control. Agents have their identity models, which are stateful and multi-step in nature. The ZTAGRAF system uses an intent- identity architecture that is based on the NIST SP 800-63 principles. There are four attributes in agent identity:

1. Identity, such as TLS certificates and signed requests
2. Behavioral identity, such as learned action patterns and normal behavior baselines
3. Role-based identity, such as service type and deployment context

4. Intent identity, such as goals and planned sequences of actions

Access decisions integrate all four identity dimensions through a risk-adaptive decision function:

AccessDecision = Policy(Identity, Intent, Context, Risk) where: - Identity: Cryptographic + behavioral + role + intent attributes - Intent: Extracted from agent planning/reasoning - Context: Request parameters, time, resource sensitivity - Risk: Anomaly score, threat level, historical breach data

Continuous verification occurs at multiple checkpoints: tool invocation (before execution), data access (before retrieval), and outcome validation (after completion). If any checkpoint indicates deviation from expected behavior, the system triggers graduated responses: logging/monitoring (low risk), requiring additional justification/human approval (medium risk), or blocking/isolating the agent (high risk).

4.3 Dynamic Policy Enforcement for RAG Pipelines

Access decisions are made using a risk-adaptive decision function that combines all four identity dimensions. This function is based on the policy, identity, intent, context and risk. The ZTAGRAF system constantly verifies the agents at stages, such as tool invocation, data access and validation of outcomes. If any checkpoint does not adhere to the desired behavior the system will invoke responses, such as logging and monitoring more justification and human approval or blocking and isolating the agent.

The ZTAGRAF system also protects RAG pipelines by using policies for each pipeline stage. The ingestion stage uses rules to authenticate and authorize data sources. The retrieval stage examines queries for injection patterns. Ranks documents based on relevance and trustworthiness scores. The generation stage verifies the output of the language model in stages including factuality verification, policy conformance verification and hallucination detection.

The ZTAGRAF system invokes tools based on their lifecycle definition. When an agent intends to invoke a tool the system verifies the tool selection against the agents permissions and current context. It also parses the intended parameters and verifies them against the specification. The system simulates the execution to predict the execution and identify any misuses. It applies least-privilege constraints. Executes the tool in a sandboxed environment with resource limits. The system then verifies the outcomes. Detects any side effects.

The guardrail system has three decision points: - execution guardrails, runtime guardrails and after-execution guardrails. The decision to use a guardrail is based on a combination of detectors, including detectors, behavior detectors, specification-based detectors and code detectors. The ZTAGRAF system is designed to provide an trustworthy environment for agentic AI systems. The ZTAGRAF system is a system that requires careful consideration of the ZTAGRAF architecture and the ZTAGRAF components. The ZTAGRAF system is a tool, for ensuring the security and accountability of AI systems.

4.4 Structured Tool Invocation and Guardrail Enforcement

When a tool is used it goes through a set of steps that are clearly defined. The system does the following things when an agent wants to use a tool:

(1) it checks if the agent is allowed to use the tool and if it is being used in the situation

(2) it looks at the information the agent wants to use with the tool and checks if it is correct

(3) it tries to guess what will happen when the tool is used to see if there are any problems

(4) it makes sure the agent only has access to the things it needs

(5) it runs the tool in a safe place with limits on what it can do

and (6) it checks what happened when the tool was used to see if everything is okay.

The guardrail system helps at three times: Before something is done it stops things that are clearly not safe like deleting a database by accident. While something is being done it watches to see if anything strange is happening, like a lot of data being moved and it can stop it if necessary. After something is done it looks at what happened. Tries to fix any problems like trying a different tool if one did not work. The guardrail system makes decisions using a group of detectors.

These detectors look at what the agent's trying to do and compare it to what it said it wanted to do. They also look for things that're not normal and make sure the tool is being used correctly. For agents that create code the detectors look for patterns like SQL injection or command injection that could cause problems. The guardrail system uses tool selection and the system checks the tool selection, against agent permissions and current context of the tool selection.The system is always checking the tool selection to make sure it is safe and being used correctly.

5. GOVERNANCE FRAMEWORK: ORGANIZATIONAL AND OPERATIONAL INTEGRATION

5.1 Maturity Model for Zero-Trust Agentic Governance

We created a 5-level plan to help companies achieve zero-trust governance. The goal is to reach zero-trust governance.

Following are the various levels through which it operates:

- Level 1 is the starting point: companies don't have a system for AI governance. They have to check everything and only react when something goes wrong.
- Level 2 is relatively better: companies have some rules and simple security measures. They still rely on rules and don't really prevent problems.
- Level 3 is the stage of more serious outlook: companies have the processes in place. They start using zero-trust concepts. They look for threats. They do not have a complete zero-trust system in place.
- Level 4 is when companies make decisions based on data: they try to improve their governance all the time. They have fully adopted zero-trust policies. They use policies that can change. They have a zero-trust system in place.
- The best level is Level 5: companies have made governance and security a part of everything they do. They are always looking for ways to get better. They can automatically. Fix problems. Zero-trust governance is part of all their development and operational processes.

This paper looked at six companies. We found that one company was at Level 2 two companies were at Level 3 two companies were at Level 4 and one company was at Level 5. The companies took 6-9 months to move to the level. They had to invest in machine learning and monitoring. The company at Level 5 had been working on zero-trust governance for than 18 months. They included zero-trust governance in their development practices.

5.2 Implementation Roadmap and Phased Deployment

Companies that want to adopt zero-trust governance should do it in stages. This will help them succeed. This is how they can achieve zero-trust governance.

The following phases are used:

Phase 1 is the starting point: companies should set up identity infrastructure start tracking and registering develop security policies and educate stakeholders. This should take 3 months.

Phase 2 is about core controls: companies should add access control layers introduce guardrails to tools install audit procedures and perform security evaluations. This should take 3 months.

Phase 3 is about integration: companies should bring together identity, access control and guardrails in all environments define automated policy enforcement and establish incident response practices. This should take 3 months.

Phase 4 is the optimization phase: companies should establish analytics and threat hunting to optimize policies based on data and scale to zero-trust deployments. This may take than 3 months.

All these phases are important, for zero-trust governance. The analysis found that companies that completed these stages and had support, allocated security resources and worked closely with security and AI departments were more successful.

Those that rushed or did not prepare their company failed and had security problems. This is how companies can achieve zero-trust governance.

6. EMPIRICAL EVALUATION AND RESULTS

6.1 Threat Mitigation Effectiveness

ZTAGRAF was evaluated against a comprehensive scenario suite spanning 47 attack scenarios across five attack categories. Results are summarized in Table 1:

Table 1: Attack Scenario Results Across Five Attack Categories.

Attack Category	Scenario Count	Success Rate (Baseline)	Success Rate (ZTAGRAF)	Detection Rate
Prompt Injection	12	68.2%	8.7%	98.4%
Tool Misuse	13	71.5%	12.3%	94.2%
Data Poisoning	11	55.8%	7.1%	96.8%
Agent Coordination Attacks	6	78.3%	15.4%	92.1%
Governance Circumvention	5	42.1%	3.2%	99.1%
Overall	**47**	**63.4%**	**9.3%**	**96.1%**

6.2 Organizational Impact Metrics

ZTAGRAF really made a difference in stopping attacks from succeeding. It cut down on the success rate of a type of attack called injection by 87.2 percent. This is a deal because injection attacks are pretty common. ZTAGRAF was also very good at finding data poisoning, which's something that is really hard

to detect. It found data poisoning 96.8 percent of the time.

The system also got a lot better at stopping governance circumvention attacks with a 92.1 percent reduction. This is because ZTAGRAF has things like audit trails and policy enforcement that help keep everything in check. When we looked at all the types of threats we found that ZTAGRAF was wrong, about something being a threat only 2.1 percent of the time. This is called a positive.

Sometimes when ZTAGRAF thought something was a threat but it was not it was because of the way some tools work. This happened 3.8 percent of the time when tools were being used. But after ZTAGRAF had been learning for two weeks it got even better at figuring out what was really a threat and what was not. It reduced the number of positives to 1.9 percent. ZTAGRAF is a system that really helps stop attacks and it keeps getting better over time.

Six organizations implementing ZTAGRAF reported significant operational improvements (Table 2):

Table 2: Organizational Metrics Pre- and Post-ZTAGRAF Implementation.

Metric	Organization	Pre-Implementation	Post-Implementation	Change
Unauthorized Tool Invocation Attempts	A	127/month	16/month	-87.4%
	B	89/month	8/month	-91.0%
	C	42/month	5/month	-88.1%
Policy Compliance Rate	A	87.2%	97.8%	+10.6pp
	B	91.3%	98.1%	+6.8pp
	C	83.4%	96.7%	+13.3pp
Incident Response Time (minutes)	A	156	8.2	-94.7%
	B	203	11.3	-94.4%
	C	118	6.8	-94.3%
Deployment Velocity (features/month)	A	12.3	11.8	-4.1%
	B	18.7	17.4	-6.9%
	C	9.2	8.5	-7.6%

The post-implementation organizations experienced reduced illegal attempts to use the tools with a decline of 87-91. This demonstrates that the zero-trust model was effective in restricting activities not permitted. Moreover, the improvement in the policy compliance rates increased by 6.8-13.3 percentage points. Critically the response time to incidents became significantly better with an increase of 94%+. This is due to the fact that investigation was eliminated as automated detection and containment were introduced. The minimal decline in deployment speed 4-7% was considered a cost to zero-trust governance. This trade-off was acceptable to organizations due to security enhancement. According to one respondent, we are able to trade 5 percent deployment velocity against 90 percent faster incident response that, said the respondent, is a deal we can make.

6.3 Operational Overhead and Performance

Approaches taken by organizations to check overhead were as below: Latency Overhead: Tool invocation latency increased between 18-47ms in average of 31ms. This was because of policy evaluation, intent verification as well as sandboxing. To agents this extra time was not noticed. The agents of batch processing did not have much additional work. Computational Overhead: 8-12% of agent inference cost was used to conduct policy evaluation and threat detection on the CPU. GPU use was very low. One organization decreased the overhead to 4.2 percent through optimization of the policies and caching. Data Overhead: 100-300 KB of agent execution was generated by audit logging. The cost of log storage in one of the organizations where executions were done every day was approximately $800/month. Saving logs after 90 days reduced the costs down to approximately 120/month. The companies believed that this was worth the additional money to enhance their security. No company claimed that overhead was a reason why ZTAGRAF should not be adopted.

6.4 Stakeholder Perspectives: Qualitative Findings

Interviews with 22 practitioners showed factors for success and barriers to implementation:

Success Factors:

- Senior management support and sufficient budget.
- Collaboration between teams in other regions.
- To introduce in installments with success.
- Matching goals with clear policies.
- Constant monitoring and feedback.

Key Barriers:

- There was complexity in integrating zero-trust governance with the systems that are there.
- The fact that many organizations did not have knowledge about AI systems and zero-trust architecture became evident.

- It caused tension between AI and security teams. There were organizations that were not certain of the ROI.
- Success was gained by more mature organizations. They were also enjoying governance processes.
- According to one security leader, zero-trust governance is not a technology. It is, and sharing responsibility between AI engineering and security teams. It is worth doing when they do succeed.

Organizations that were more mature had success. They had governance processes in place. One security leader said, "Zero-trust governance isn't about technology. It's, about both AI engineering and security teams taking responsibility together. When they do success is possible."

7. DISCUSSION

7.1 Theoretical Contributions

This research helps cybersecurity and AI governance in three areas:

This study assists in cybersecurity and AI governance in three dimensions: Against Agentic AI Threat Modeling: New threat models dealt with failures of the AI itself or software bugs. Our model considers risks in five categories: thinking errors) temporal (memory attacks) operational (tool misuse) coordination (working together on multiple AI) and governance (avoiding rules). This entire model assists in making AI systems safe. Architecture-Governance Integration: We have discovered that security architecture and governance have to collaborate. Firms that had security architecture and weak governance did not improve significantly. People who had governance and a weak architecture were also prone. Our solution is a hybrid that merges architecture and governance, which makes it more efficient. Constant Checking in Autonomous Systems: In regular security, human beings do decision-making. Artificial intelligence systems come out with independent choices. We developed a system that monitors AI actions within the system: prior to their occurrence (intent verification) during their occurrence (behavior analysis), and after their occurrence (outcome verification). This is a strategy of applying zero-trust to AI systems.

7.2 Practical Implications

For those using AI our findings offer practical guidance:

- Start with Identity Not Access Control: Many companies focus on access control first and find identity gaps later. Our approach emphasizes - dimensional agent identity.
- Stakeholders should invest in Intent Verification. Lots of attacks are successful because the system is not able to figure out what is behavior and what is malicious behavior. Intent Verification is a technique that helps us understand what the user or agent is trying to do. By using Intent Verification we can make sure that Intent Verification is real and that it is what the agent or user really wants to do. This can help keep our system safe from people. Intent Verification can provide lots of security benefits, for our system and the people who use it.
- Secure RAG Pipelines Holistically: Point solutions for security provide protection. Integrated defenses addressing ingestion, retrieval and generation reduce attack success rates.
- Design for Monitoring: Security controls relying on static rules fail against adaptive adversaries. Continuous monitoring and dynamic policy adjustment provide long-term security.
- Govern by Phase: Organizations attempting zero-trust implementation simultaneously experience failure rates exceeding 40%. Phased approaches achieve 80%+ success rates.

7.3 Limitations and Boundaries

This research has limitations:

- Sample Size: We worked with 22 practitioners and 14 organizations; findings may not generalize to all contexts.
- Evaluation Timeframe: Implementation evaluation lasted 8-12 weeks; term operational impacts remain unobserved.
- Adversarial Evaluation: Threat scenarios were developed by researchers and known attack vectors; zero-day attacks and novel attack classes may evade defenses.
- Model-Specific Findings: Evaluation focused on three LLMs; different models may exhibit vulnerability profiles.
- Regulated vs. Non-Regulated: Implementation contexts included both non-regulated organizations; findings may be biased toward regulated contexts.

7.4 Emerging Challenges and Future Directions

There are various new issues, which need to be researched:

- Adversarial AI vs. Defensive AI: With attackers coming up with advanced AI based attacks countermeasures might need advanced evolution. Studies of the AI-enabled defense and resistance to adversaries are essential.

- Multiagent Alignment and Coordination: The more autonomous agentic systems grow the more challenging it is to achieve alignment among a number of independent agents. Mechanism design and game-theoretic approaches can be solutions.
- Quantum-Resistant Cryptography: Defenses relying on identity will require migration to post-quantum cryptography as quantum computing matures.
- Interpretability and Explainability: For deployments stakeholders require explainable security decisions. Current guardrail implementations provide explainability; enhancing transparency without compromising effectiveness is an open problem.
- Standardization and Interoperability: As organizations deploy AI systems, from multiple vendors standardized security interfaces and governance protocols become essential. Industry initiatives are beginning to address this; formal standards would accelerate adoption.

8. CONCLUSIONS

This study combines research methods to explore how to govern Artificial Intelligence (AI) systems that can act on their own. The study found that using a zero-trust approach, where security checks are built into the system and constantly updated can greatly reduce the risk of attacks on AI systems.

In fact our results show that this approach can reduce attack success rates from 63% to 9% without slowing down operations of AI systems. The ZTAGRAF system is a way to apply zero-trust principles to AI systems that can act on their own like ZTAGRAF. It addresses challenges that traditional security systems can't handle well with AI systems. We tested ZTAGRAF in scenarios and found that it works well and can be easily implemented for AI systems. Our research also shows that organizations need to change their governance structure and security approach at the time as they implement AI systems. Organizations that implement zero-trust governance in a step-by-step way with teams from departments and support from top executives get much better results than those that only focus on technology for AI systems. For researchers our study provides a foundation for work on AI security and AI systems. We created a list of threats a framework for understanding them and proof that our approach works for AI systems. For practitioners we provide a roadmap for implementing zero-trust governance and factors that contribute to success with AI systems. As AI systems that can act on their own become more common in areas like finance, healthcare and government strong security governance is essential for AI systems. Our study shows that zero-trust governance is possible can be implemented and is acceptable for AI systems.

The next step is for security researchers AI engineers and organizational leaders to work together to ensure that AI capabilities are used responsibly with AI systems. The ZTAGRAF architecture helps to achieve this goal by providing a framework for zero-trust governance of AI systems like ZTAGRAF. The zero-trust approach is crucial for ensuring the security and reliability of these AI systems. ZTAGRAF and zero-trust governance are key, to realizing the benefits of AI while minimizing risks with AI systems.

REFERENCES

Adabara, I., Sadiq, B. O., Shuaibu, A. N., Danjuma, Y. I., & Maninti, V. (2025). Trustworthy agentic AI systems: A cross-layer review of architectures, threat models, and governance strategies for real-world deployment. *F1000Research*.

Adabara, I., Sadiq, B. O., Shuaibu, A. N., Danjuma, Y. I., & Venkateswarlu, M. (2025). A review of agentic AI in cybersecurity: Cognitive autonomy, ethical governance, and quantum-resilient defense. *F1000Research*.

Alampalli, S. (2025). Privacy-first, AI-driven web analytics: An architecture for schema governance, consent compliance, and intelligent policy enforcement. *2025 6th International Conference on IoT Based Control Networks and Intelligent Systems (ICICNIS)*.

Allegrini, E., Shreekumar, A., & Celik, Z. B. (2025). Formalizing the safety, security, and functional properties of agentic AI systems. *arXiv.org*.

Ammann, L., Ott, S., Landolt, C. R., & Lehmann, M. P. (2025). Securing rag: A risk assessment and mitigation framework. *Swiss Conference on Data Science*.

Ampel, B., Samtani, S., Zhu, H., Chen, H., & Nunamaker, J. (2024). Improving threat mitigation through a cybersecurity risk management framework: A computational design science approach. *Journal of Management Information Systems*.

An, H., Zhang, J., Du, T., Zhou, C., Li, Q., Lin, T., & Ji, S. (2025). IPIGuard: A novel tool dependency graph-based defense against indirect prompt injection in LLM agents. *Conference on Empirical Methods in*

Natural Language Processing.
Arora, S., & Hastings, J. (2025). Securing agentic AI systems - a multilayer security framework. *arXiv.org*.
Atre, S. (2026). Legal challenges of agentic AI systems in education and employment decision-making. *Indian Journal of Computer Science and Technology*.
Bhushan, B., Rajgopal, P. R., & Sharma, K. (2025). An intent-aware zero trust identity architecture for unifying human and machine access. *International Journal of Computational and Experimental Science and Engineering*.
Boisvert, L., Bansal, M., Evuru, C. K. R., Huang, G., Puri, A., Bose, A., Fazel, M., Cappart, Q., Stanley, J., Lacoste, A., Drouin, A., & Dvijotham, K. (2025). DoomArena: A framework for testing AI agents against evolving security threats. *arXiv.org*.
Bommareddy, A. R. (2025). Strengthening cybersecurity resilience in federal financial systems through zero-trust architectures. *Journal of Information Systems Engineering & Management*.
Castagnaro, A., Salviati, U., Conti, M., Pajola, L., & Pizzi, S. (2025). The hidden threat in plain text: Attacking RAG data loaders. *AISec@CCS*.
Chakrabarty, P. K. (2025). Adversarial attacks on agentic AI systems: Mechanisms, impacts, and defense strategies. *International Journal of Science and Research (IJSR)*.
Chennabasappa, S., Nikolaidis, C., Song, D., Molnar, D., Ding, S., Wan, S., Whitman, S., Deason, L., Doucette, N., Montilla, A., Gampa, A., Paola, B. de, Gabi, D., Crnkovich, J., Testud, J.-C., He, K., Chaturvedi, R., Zhou, W., & Saxe, J. (2025). LlamaFirewall: An open source guardrail system for building secure AI agents. *arXiv.org*.
Chowdhury, M. A. R., Dang, H. N., Bazan-Antequera, R., Khan, Md. S., Karim, M. R., & Manakkadu, S. (2025). Towards a serverless intelligent firewall: AI-driven security, and zero-trust architectures. *International Conference on Cyber Security and Cloud Computing*.
Creamer, E. G. (2024). Weighing mixing in a decision about priority in mixed methods research. *Journal of Mixed Methods Research*.
Datta, S., Nahin, S. K., Chhabra, A., & Mohapatra, P. (2025). Agentic AI security: Threats, defenses, evaluation, and open challenges. *arXiv.org*.
Devagiri, B. R. (2025). Autonomous zero trust enforcement: Revolutionizing security through AI-powered identity behavior analytics. *Journal of Computer Science and Technology Studies*.
Du, X. (2025). Empowering LLM-based agents: Methods and challenges in tool use. *Applied and Computational Engineering*.
Dwivedi, Y. K., Helal, M. Y., Elgendy, I., Alahmad, R., Walton, P., Suh, A., Singh, V., & Jeon, I. (2025). Agentic AI systems: What it is and isn"t. *Global Business and Organizational Excellence*.
Engin, Z., & Hand, D. (2025). Toward adaptive categories: Dimensional governance for agentic AI. *arXiv.org*.
Fbregues, S., Younas, A., Escalante-Barrios, E., Molina-Azorn, J. F., & Vzquez-Miraz, P. (2024). Toward a framework for appraising the quality of integration in mixed methods research. *Journal of Mixed Methods Research*.
Ferrag, M., Tihanyi, N., Hamouda, D., Maglaras, L. A., & Debbah, M. (2025). From prompt injections to protocol exploits: Threats in LLM-powered AI agents workflows. *ICT Express*.
Filho, W. L. R. (2025). The role of zero trust architecture in modern cybersecurity: Integration with IAM and emerging technologies. *Brazilian Journal of Development*.
Gambo, M. L., & Almulhem, A. (2025). Zero trust architecture: A systematic literature review. *Journal of Network and Systems Management*.
Gasmi, T., Guesmi, R., Belhadj, I., & Bennaceur, J. (2025). Bridging AI and software security: A comparative vulnerability assessment of LLM agent deployment paradigms. *Information Sciences*.
Gong, Y., Chen, Z., Chen, M., Yu, F., Lu, W., Wang, X., Liu, X., & Liu, J. (2025). Topic-FlipRAG: Topic-orientated adversarial opinion manipulation attacks to retrieval-augmented generation models. *USENIX Security Symposium*.
Gurram, A. (2025). Generative AI for enhanced cybersecurity: Building a zero-trust architecture with agentic AI. *World Journal of Advanced Engineering Technology and Sciences*.
Hiriyanna, S., & Zhao, W. (2025). Multi-layered framework for LLM hallucination mitigation in high-stakes applications: A tutorial. *De Computis*.
Huang, K., Huang, J., Mehmood, Y., Atta, H., Baig, M., & Haq, M. A. U. (2025). AAGATE: A NIST AI RMF-aligned governance platform for agentic AI. *arXiv.org*.

Hussain, M. H., Riaz, A., & Butt, A. (2025). Integrating COBIT 2019 with zero trust architecture: A strategic approach to GRC in cybersecurity. *VFAST Transactions on Software Engineering*.

Joshi, H. (2024). Emerging technologies driving zero trust maturity across industries. *Institute of Electrical and Electronics Engineers*.

Joshi, S. (2025). Advancing u.s. Competitiveness in agentic gen AI: A strategic framework for interoperability and governance. *International Journal of Innovative Science and Research Technology*.

Khair, R., Azzahra, H., Lubis, S., & Febrian, V. (2025). Blockchain implementation in the development of a zero trust-based cybersecurity framework at the indonesian national data center. *Jurnal Sistem Komputer Dan Informatika (JSON)*.

Khan, I. U., Khan, F. M., Haider, Z. A., & Alturise, F. (2025). Integrating AI, blockchain, and edge computing for zero-trust IoT security: A comprehensive review of advanced cybersecurity framework. *Computers, Materials &Amp; Continua*.

Khan, R. I., Habib, M. Q. B., Hasan, Md. M., & Shad, K. W. U. (2026). Zero trust architecture in cloud-native environments: A scalable framework for cybersecurity. *International Journal of Science and Research Archive*.

Khoo, S., Foo, J., & Lee, R. K.-W. (2025). With great capabilities come great responsibilities: Introducing the agentic risk & capability framework for governing agentic AI systems. *arXiv.org*.

Kim, S., Kim, J., Jeon, Y., & Lee, G. (2025). Safeguarding RAG pipelines with GMTP: A gradient-based masked token probability method for poisoned document detection. *Annual Meeting of the Association for Computational Linguistics*.

Kumar, Dr. S. N. P. (2025). Building scalable and reliable agentic AI systems: A technical blueprint for autonomous intelligence. *Global Journal of Engineering and Technology Research*.

Kumurdjieva, M., Doukovska, L., Ghouaiel, N., & Dhanani, A. (2025). Governance of AI and agentic systems: Challenges and methodologies. *2025 International Conference on Big Data, Knowledge and Control Systems Engineering (BdKCSE)*.

Lazer, S. J., Aryal, K., Gupta, M., & Bertino, E. (2026). A survey of agentic AI and cybersecurity: Challenges, opportunities and use-case prototypes. *arXiv.org*.

Li, H., Liu, X., Chiu, H.-C., Li, D., Zhang, N., & Xiao, C. (2025). DRIFT: Dynamic rule-based defense with injection isolation for securing LLM agents. *arXiv.org*.

Li, Z., Zhang, H., & Zhang, J. (2025). Token-level precise attack on RAG: Searching for the best alternatives to mislead generation. *arXiv.org*.

Mou, Y., Xue, Z., Li, L., Liu, P., Zhang, S.-B., Ye, W., & Shao, J. (2026). ToolSafe: Enhancing tool invocation safety of LLM-based agents via proactive step-level guardrail and feedback. *arXiv.org*.

Mushtaq, S., Mohsin, M., & Mushtaq, M. M. (2025). A systematic literature review on the implementation and challenges of zero trust architecture across domains. *Italian National Conference on Sensors*.

N, P. S., Pimpalkar, A., Shelke, N. M., & Saini, D. K. J. B. (2025). Zero trust architectures empowered by AI: A paradigm shift in cloud and edge cybersecurity. *2025 3rd International Conference on Sustainable Computing and Data Communication Systems (ICSCDS)*.

Nalluri, S., Malyala, M. M., Kandagiri, H., Jakku, P. C., & Kandagiri, K. K. (2025). AI-enhanced zero trust architecture for cloud security with quantum resilience. *2025 6th International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)*.

Nandagopal, S. (2025). Securing retrieval-augmented generation pipelines: A comprehensive framework. *None*.

Narajala, V. S., & Narayan, O. (2025). Securing agentic AI: A comprehensive threat model and mitigation framework for generative AI agents. *arXiv.org*.

Nguyen, V. K., & Husain, M. I. (2025). Penetration testing of agentic AI: A comparative security analysis across models and frameworks. *arXiv.org*.

Obrik-Uloho, E. P., Ejiofor, V. O., Egonwanne, C. H., Kolo, F. H. O., & Olasege, R. (2025). Zero-trust architecture for smart hospitals: A virtual blueprint for cyber-resilient healthcare infrastructure. *Archives of Current Research International*.

Paraschiv, E., Vevera, A., Crnu, C., Bjenaru, L., Dinu, A., & Prada, G. (2026). Towards trustworthy AI agents in geriatric medicine: A secure and assistive architectural blueprint. *Future Internet*.

Patel, B., Belli, D., Jalalirad, A., Arnold, M., Ermolov, A., & Major, B. (2025). Dynamic tool dependency retrieval for efficient function calling. *arXiv.org*.

Pervez, H., Gaurav, S., Heikkonen, J., & Chaudhary, J. (2025). Governance-as-a-service: A multi-agent

framework for AI system compliance and policy enforcement. *arXiv.org*.
Rai, P., Sood, S., Madisetti, V. K., & Bahga, A. (2024). GUARDIAN: A multi-tiered defense architecture for thwarting prompt injection attacks on LLMs. *Scientific Research Publishing*.
Ramakrishnan, B., & Balaji, A. (2025). Securing AI agents against prompt injection attacks. *arXiv.org*.
Raza, S., Sapkota, R., Karkee, M., & Emmanouilidis, C. (2025). TRiSM for agentic AI: A review of trust, risk, and security management in LLM-based agentic multi-agent systems. *arXiv.org*.
Reddy, A. R. P. (2025). Zero trust architecture: An AI-driven framework for modern cybersecurity challenges. *FMDB Transactions on Sustainable Intelligent Networks*.
Schmitz, C., Rystrm, J., & Batzner, J. (2025). Oversight structures for agentic AI in public-sector organizations. *Proceedings of the 1st Workshop for Research on Agent Language Models (REALM 2025)*.
Schoonenboom, J., & Johnson, R. B. (2017). How to construct a mixed methods research design. *KZfSS Klner Zeitschrift Fr Soziologie Und Sozialpsychologie*.
Sengupta, A. (2024). Securing the autonomous future a comprehensive analysis of security challenges and mitigation strategies for AI agents. *None*.
Shih, Y.-K., & Kang, Y.-K. (2025). Design and implementation of a secure RAG-enhanced AI chatbot for smart tourism customer service: Defending against prompt injection attacks - a case study of hsinchu, taiwan. *arXiv.org*.
Stefano, G. D., Schnherr, L., & Pellegrino, G. (2024). Rag and roll: An end-to-end evaluation of indirect prompt manipulations in LLM-based application frameworks. *arXiv.org*.
Suggu, S. K. (2025). Agentic AI workflows in cybersecurity: Opportunities, challenges, and governance via the MCP model. *Journal of Information Systems Engineering & Management*.
Syros, G., Suri, A., Nita-Rotaru, C., & Oprea, A. (2025). SAGA: A security architecture for governing AI agentic systems. *arXiv.org*.
Tamboly, N. A., Costantini, L. P., Connolly, M., & Alhayajneh, A. (2026). Rethinking human-centric cybersecurity: A mixed-methods analysis of incident severity determinants. *Computer and Information Science*.
Trivedi, B. (2025). Retail cybersecurity in the agentic age: Securing autonomous shopping agents in e-commerce. *European Modern Studies Journal*.
Venkataramanan, S., & Joy, M. (2025). AI-powered policy formulation and enforcement in zero trust frameworks. *Social Science Research Network*.
Vijay, D., & Ethiraj, V. (2025). Graph-symbolic policy enforcement and control (g-SPEC): A neuro-symbolic framework for safe agentic AI in 5G autonomous networks. *arXiv.org*.
Wok, K. (2025). Evaluating retrieval-augmented generation variants for clinical decision support: Hallucination mitigation and secure on-premises deployment. *Electronics*.
Wu, Z., Cho, S., Mohammed, U., Munoz, C., Costa, K., Guan, X., King, T., Wang, Z., Kazim, E., & Koshiyama, A. S. (2025). LibVulnWatch: A deep assessment agent system and leaderboard for uncovering hidden vulnerabilities in open-source AI libraries. *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*.
Xu, D., Gondal, I., Yi, X., Sunjak, T., Watters, P. A., & Mcintosh, T. R. (2025). The erosion of cybersecurity zero-trust principles through generative AI: A survey on the challenges and future directions. *Journal of Cybersecurity and Privacy*.
Zeng, W., Zhu, H., Qin, C., Wu, H., Cheng, Y., Zhang, S., Jin, X., Shen, Y., Wang, Z., Zhong, F., & Xiong, H. (2025). *Multi-level value alignment in agentic AI systems: Survey and perspectives*.
Zhang, D., Li, Z., Luo, X., Liu, X., Li, P., & Xu, W. (2025). MCP security bench (MSB): Benchmarking attacks against model context protocol in LLM agents. *arXiv.org*.
Zhou, K., Zheng, Y., He, Y., Xue, M., Gong, X., Wang, Y., & Lam, K.-Y. (2026). Beyond max tokens: Stealthy resource amplification via tool calling chains in LLM agents. *arXiv.org*.
Zhou, P., Feng, Y., & Yang, Z. (2025). Privacy-aware RAG: Secure and isolated knowledge retrieval. *arXiv.org*.
Zhu, J., Tseng, K., Vernik, G., Huang, X., Patil, S. G., Fang, V., & Popa, R. A. (2025). MiniScope: A least privilege framework for authorizing tool calling agents. *arXiv.org*.

APPENDIX: ZTAGRAF TECHNICAL SPECIFICATIONS

Architecture Layers Summary: - Layer 1: Multi-dimensional identity (cryptographic, behavioral, role, intent) - Layer 2: Intent extraction via reasoning analysis; deviation detection - Layer 3: ABAC policies; continuous verification checkpoints - Layer 4: Multi-stage RAG defense (ingestion, retrieval, generation) - Layer 5: Structured tool lifecycle; guardrail ensemble (semantic, behavior, specification, code) - Layer 6: Agent behavior monitoring; anomaly detection; multi-agent policy enforcement - Layer 7: Immutable audit trails; blockchain-assisted logging; forensic support

Maturity Progression: Organizations successfully transitioning from Level 2 to Level 4 require structured governance, dedicated resources, and 6-9 months implementation time. Executive sponsorship and cross-functional teams are critical success factors.

This research provides organizations with both theoretical foundations and practical frameworks for deploying agentic AI systems responsibly in mission-critical contexts.