AI Shared Responsibility Framework, V1.0

Workstream 2: Preparing Defenders for a Changing Cybersecurity Landscape

Version 1.0

26 May 2026

Contents

AI Shared Responsibility Framework, V1.0

OASIS Open Project : Coalition for Secure AI (CoSAI) Workstream 2: AI Shared Responsibility Framework

Approved by the CoSAI Project Governing Board on 26 May 2026

1. Executive Summary

The evolving landscape of Artificial Intelligence (AI) necessitates a refined approach to accountability within AI systems. Given the criticality of clear understanding of accountability and responsibility across shared systems - including AI systems, we introduce an expanded framework that aligns with and extends established AI Shared Responsibility Frameworks (SRF) [5,16]. This model is designed to identify accountability across the AI ecosystem, with examples addressing industry-specific regulatory requirements and model supply chain risks.

The implementation of non-trivial business-specific AI systems presents organizations with significant accountability gaps [1]. Incidents involving AI system failures, harm caused, or regulation violations often lead to complex scenarios where identifying the responsible parties becomes challenging due to blurred responsibility boundaries [2]. Such ambiguities both delay resolution and impede effective incident response and regulatory compliance [3, 4, 6, 7]. Emerging agentic AI systems capable of semi-autonomous operations on behalf of humans or other agents create new challenges in accountability.

CoSAI uses enterprise architecture layers to address the critical issues and complexities in responsibility posed by modern AI systems. This expands on the architectural view of responsibility by adding additional layers inherent in solutions using AI. CoSAI has added **AI Business** to the **Usage** layer to address Industry-Specific Regulatory Compliance, added **AI Information layer** to include data owners, and the **AI Model Provider** to address the Supply Chain. While organizational structures are manifold, and personas and their roles may vary widely from one organization to another, the dependencies encoded in the enterprise architecture layers remain invariant when provisioning, building or operating AI systems. The CoSAI Shared Responsibility Framework provides a governance manual for decomposing AI system solutions into components with one accountable party.

Key Takeaways

- 5 layers provide clear accountability across AI systems
- Close regulatory compliance gaps with solution decomposition
- Addresses the AI agentic governance challenges resulting from accelerated adoption
- Provides implementation playbooks and evidence requirements

2. Current state and challenges

The advent of AI-enabled systems has ushered in an era characterized by probabilistic behaviors and novel risks emanating from user-generated content, training data, and the model providers themselves. Recognizing these challenges, numerous industry regulators have proactively introduced governance frameworks specifically tailored to address AI-related concerns pertinent to their respective sectors.

In response to this evolving landscape, we present a strategic framework designed for identifying responsible component owners and assigning accountability. This structure depicts dependencies and responsibilities across the AI ecosystem. Our approach aims to ensure alignment with multiple industry regulations while fostering transparency and trust among all stakeholders involved in the deployment and operation of AI systems.

2.1 Current State Pain Points

Ambiguous ownership is a growing liability for AI system deployments. These pain points underscore why a formal shared responsibility model is no longer optional for AI.

- **Vendor Complexity**: When a single application relies on disparate model providers, cloud platforms, and agent frameworks, isolated contracts create operational blind spots. Critical handoffs fail when no one has full-stack oversight.
- **Incident response**: When an adverse event spans multiple layers of an AI system, triage stalls. Without explicitly assigned owners for detection, containment, and remediation, teams default to the finger-pointing cycle.
- **Regulatory compliance**: Demonstrating adherence to AI-specific mandates, such as data governance, bias testing, or model validation, fails instantly when organizations cannot map specific controls to designated owners across the value chain.
- **Shadow AI proliferation:** Unsanctioned AI usage thrives in the gray areas between IT security, business units, and external vendors. Left unassigned, no single group monitors for rogue usage or owns the risk of sensitive data exposure.

CoSAI's five-layer framework addresses these pain points by establishing exactly a clear accountable party for each component across the full AI system.

2.1.1 Vendor Management Complexity

AI system procurement may include the following parties:

- Foundation model providers
- Specialized AI platform providers
- Agentic system providers
- Application developers
- End Organizations with sector-specific constraints and regulations (Healthcare, Finance, Public Sector)

2.1.2 Examples of AI regulatory impact in Industry

- **Healthcare**: FDA AI/ML guidance creates new validation requirements [8] not covered by traditional cloud models, and subsequent requirements cascade down the layers. Without clear responsibility assignment, organizations cannot determine who validates model changes or documents compliance.

- **Financial Services**: SR 11-7 model risk management [9] applies differently to AI than traditional software, where risk also cascades down supporting layers. Ambiguous accountability makes it impossible to assign model risk ownership across the multi-vendor AI stack.

- **Public Sector**: NIST AI RMF requirements [10] span multiple organizational boundaries, between organizations and within each organization.The framework enables government agencies to map NIST controls to specific accountable parties across vendor relationships.

- **Consumer Protection**:
  - **Airline**: In 2022, a national carrier was sued by a customer after the airline failed to honor a claim made by its chatbot that contradicted its actual bereavement policy. A Canadian court held the organization responsible, making it pay $812.02 to the customer [13].

  - **Car Dealership**: By negotiating with the chatbot of a car dealership to agree without seeking proper authorization, a customer manipulated the chatbot to agree to sell a 2024 Chevy Tahoe for $1 [17].

2.2 Use Case Example

Consider a prompt injection attack that bypasses guardrails in a customer-facing chatbot, exposing PII from a connected database. Under the five-layer model, accountability traces cleanly:

The AI Model Provider is accountable for the base model's susceptibility to prompt injection and for documenting known weaknesses in the model card. The Cloud/AI Platform Provider is accountable for infrastructure-level protections such as tenant process isolation that contain blast radius. The Application Developer is accountable for implementing application-level guardrails, input validation, and data access controls that should have prevented the chatbot from reaching PII in the first place. The deploying organization is accountable for having classified that data appropriately, defining which data the chatbot could access, and maintaining incident response procedures.

Without this layered accountability, the typical outcome is the cycle described above: the model provider blames configuration, the cloud provider points to the tenant, and the application team cites model limitations. The framework turns "whose fault is this?" into "which layer's controls failed, and who owns remediation for each?" This is an example that highlights the complexity and the two prompt injection responsibility points including the upstream and downstream data sources. Details of this example can be found in Appendix A.3.6.

3. The CoSAI AI Shared Responsibility Framework

CoSAI AI Responsibility Framework
The SRF is primarily an accountability framework, it answers *who* is responsible for each component across the AI stack. It complements rather than replaces existing control and management frameworks: NIST AI RMF defines *what* governance outcomes to achieve, ISO/IEC 42001 defines

how to manage an AI management system, and EU AI Act defines *which* regulatory obligations apply by risk tier. Compliance controls referenced throughout this document illustrate how accountability at each layer is discharged in practice; they are not requirements of the SRF itself.

There should be exactly one accountable party per component to prevent overlaps. Specific Security & Governance requirements cascade from the Business layer to the supporting layers. The layers illustrate areas of responsibility where personas reflect types of actors who are accountable. For additional detail on the Cloud SRM, consult **Appendix A1**.

Depending on the solution operating model, IaaS, PaaS, or SaaS, the solution layers and their components will be provided by different entities, understanding which is key to assigning responsibility. Enterprise solutions are unlikely to be monolithically operating within one model, so defining the operating model for AI system components may be necessary. Responsibility is shared between customers and providers at the boundary of what is being provided, and the enterprise architecture layers indicate governance activities mandated to personas responsible for that solution layer. The CoSAI Shared Responsibility Framework answers "Who is responsible, internally and externally, for issues with AI systems?"

Principle: *There should be exactly one accountable party per activity to prevent overlaps.*

CoSAI 5-Layer AI Shared Responsibility Matrix

		IaaS (Infrastructure as a Service)	AI & Agent PaaS (Platform as a Service)	AI Enabled SaaS (Software as a Service)
AI Usage & Business Verticals (C-Suite)	Capabilities & Business Strategy	Customer	Customer	Customer
	Processes & Governance	Customer	Customer	Customer
	Business Units & Accountability	Customer	Customer	Shared
AI Information (Data Owners)	Master Data Management	Customer	Customer	Shared
	Privacy Controls & Policies	Customer	Shared	Shared
	AI Training Data	Customer	Shared	Provider
AI Application (Dev Teams)	Agents & Orchestration Models	Customer	Shared	Provider
	APIs & Fine-tuned Models	Customer	Shared	Provider
	Application Platforms	Customer	Provider	Provider
AI Platform (Platform Providers)	Guardrails & Safety Systems	Shared	Shared	Provider
	Compute Infrastructure	Shared	Provider	Provider
	LLM Routers & Gateways	Customer	Provider	Provider
AI Model Provider (Supply Chain)	Model Distribution	Customer	Provider	Provider
	Model Governance	Model Dependent	Provider	Provider
	Foundation Models	Model Dependent	Model Dependent	Provider

Provider | Shared | Customer | Model Dependent

Key Insights for Auditors & Stakeholders:

- **IaaS:** Maximum customer responsibility, especially for governance and application layers
- **AI-PaaS:** Balanced shared responsibility model with provider managing AI platform infrastructure
- **Agent-PaaS:** Balanced shared responsibility model with provider managing agentic platform infrastructure
- **AI-SaaS:** Provider assumes majority of technical responsibilities, customer retains business governance
- **Model Dependent:** Foundation layer responsibilities vary based on model licensing and deployment approach

3.1 Personas

While the Enterprise Architecture layers provide solution dependencies that indicate governance components, the heterogeneity of solutions means that the persona for a given activity may vary from solution to solution. The framework recognizes areas of responsibility, aligned with the eight CoSAI-RM Personas [22]:

3.1.1 Agentic Platform and Framework Providers

- **Description**: Actors that provide the development environments, software frameworks, and orchestration runtimes required to implement agentic reasoning, planning, and tool execution.
- **Standards Mapping**: ISO/IEC 22989:2022 §5.19.5 - AI Partner (tool and framework provider)
- **AI SRF Layers**: AI Application, AI Platform
- **Responsibilities**:
  - Ensuring framework security and sandboxing.
  - Implementing safety controls around tool execution primary responsibility
  - Managing state in multi-turn workflows securely.
  - Integrating APIs securely.
  - Defining cognitive architecture for AI systems.

3.1.2 Application Developer

- **Description**: Actors that integrate AI models (via APIs or embedded models) into applications, products, or services. They may consume models without modifying them, or perform light customization (prompt engineering, RAG, etc.).
- **Standards Mapping**: ISO/IEC 22989:2022 §5.19.2 - AI Provider (deploys AI system for use by end users); also acts as AI Customer (§5.19.4) with respect to upstream model and platform providers
- **AI SRF Layer**: AI Application
- **Responsibilities**:
  - Implementing application-level security controls.
  - Implementing safety controls around tool execution supporting responsibility
  - Ensuring input validation and output filtering.
  - Managing user access control mechanisms.

3.1.3 Data Provider

- **Description**: Actors that supply training data, evaluation datasets, or inference data to model providers or application developers. This includes data aggregators, data marketplaces, and those licensing datasets.

- **Standards Mapping**: ISO/IEC 22989:2022 §5.19.5 - AI Partner (data provider)
- **AI SRF Layer**: AI Information
- **Responsibilities**:
  - Conducting data quality assurance.
  - Tracking data provenance and compliance.
  - Implementing privacy protections in data handling.
  - Manage data classification

3.1.4 AI System Users

- **Description**: Actors that use AI-powered applications or services without developing or deploying the AI components themselves. Users rely on application developers and providers for AI security controls.
- **Standards Mapping**: ISO/IEC 22989:2022 §5.19.4 - AI Customer (end user sub-role)
- **AI SRF Layer**: AI Usage & Business
- **Responsibilities**:
  - Adhering to appropriate use policies.
  - Reporting issues or anomalies detected during use.
  - Following usage guidelines to minimize data inputs.

3.1.5 AI System Governance

- **Description**: Actors responsible for defining security control objectives, measuring implementations, and enforcing compliance for AI systems across the AI system lifecycle. This includes AI risk officers, compliance teams, and governance boards.
- **Standards Mapping**: ISO/IEC 22989:2022 §5.19.4 - AI Customer (acquires and governs AI systems on behalf of the organization); where governance extends to enforcing regulatory obligations, also maps to Relevant Authority (§5.19.7)
- **AI SRF Layer**: AI Usage & Business
- **Responsibilities**:
  - Establish security and governance rules that all AI systems must follow, including acceptable risk levels based on system importance.
  - Evaluate how well security measures work across the AI lifecycle and confirm they're used correctly by everyone involved.
  - Make sure AI deployments comply with necessary standards, laws, and company policies throughout their operation.
  - Identify, prioritize, and manage both security and operational risks in AI systems, keeping a register of these issues along with assigned owners and timelines for fixing

them.

- Ensure there are plans to handle incidents, assign accountability across all AI system layers, review post-incident analysis, and implement fixes for any recurring problems.

3.1.6 Model Provider

- **Description**: Actors that develop, train, evaluate, and tune AI/ML models (foundation models, specialized models, or domain-adapted models). This includes those that develop models from scratch or significantly modify existing models for distribution..

- **Standards Mapping**: ISO/IEC 22989:2022 §5.19.3 - AI Producer (develops, trains, and validates AI models for distribution)

- **AI SRF Layer**: AI Model Provider

- **Responsibilities**:
  - Secure and responsible model architecture design and training.
  - Model security, safety, and performance validation.
  - Provide clear documentation or “model cards” detailing provenance, benchmarks, and intended use.
  - Publish and maintain model version and vulnerability disclosures.

3.1.7 AI Model Serving

- **Description**: The entity responsible for provisioning, managing, and securing the runtime environment that serves AI and ML model predictions at scale. This persona covers all model types, including classical ML, statistical, optimization, and generative AI models, focusing on the secure execution of predictions, ensuring runtime integrity, confidentiality, and availability of data and outputs. It separates its duties from model training, tuning, or registry storage (Model Provider) and physical infrastructure management (AI Platform Provider), focusing on the secure orchestration and delivery of the model serving application layer.

- **Standards Mapping**: ISO/IEC 22989:2022 §5.19.5 - AI Partner (platform and runtime services provider); distinct from the infrastructure provider sub-role in that this persona focuses on model serving orchestration, not physical compute

- **AI SRF Layers**: AI Platform

- **Responsibilities**:
  - Manage secure API endpoints, enforce access policies, and perform rigorous input validation.
  - Execute models in isolated or confidential computing environments to protect sensitive runtime data.
  - Ensure the integrity of models and datasets at load-time and during runtime, preventing execution of malicious or compromised artifacts.
  - Monitor and validate outputs to prevent unintended disclosures and ensure safety and

confidentiality.

- Secure underlying infrastructure and conduct adversarial simulations to test the robustness and security of model serving.

3.1.8 AI Platform Provider

- **Description**: Actors that provide infrastructure, compute resources, APIs, and platform services for AI model hosting, training, or inference. This includes internal infrastructure teams, cloud providers (AWS, Azure, GCP), MLOps platforms, and model API services.

- **Standards Mapping**: ISO/IEC 22989:2022 §5.19.5 - AI Partner (infrastructure provider); also acts as AI Provider (§5.19.2) where the platform is deployed to serve application developers as its customers

- **AI SRF Layers**: AI Platform

- **Responsibilities**:
  - Secure infrastructure against unauthorized access and maintain high availability.
  - Maintaining platform-level compliance certifications (e.g., SOC 2 Type II, ISO 27001, FedRAMP).
  - Providing robust Identity and Access Management (IAM) primitives that upstream tenants use to enforce their own access policies.
  - Providing configurable data residency, encryption at rest and in transit, and region-locking capabilities aligning with sovereignty data protection requirements.
  - Guaranteeing uptime, providing platform-level incident notification, and defining SLAs that bound the platform's contribution to any AI system incident response timeline.

3.2 Cloud Operating Models

The AI Shared Responsibility Framework applies across multiple deployment and service models. Understanding how responsibilities shift across these Cloud operating models (SaaS, PaaS, IaaS) is critical for organizations to properly assign accountability and manage risk. This section maps the framework layers to standard cloud service models, enhanced to address AI-specific considerations.

Each Cloud operating model represents a different distribution of control and responsibility between providers and consumers, impacting layers of the framework: AI Business & Usage, AI Information, AI Applications, AI Platform, and AI Model Providers. Detailed breakdown and definitions of the matrix can be found in **Appendix A.2**.

3.2.1 Cloud Operating Models Responsibility Matrix

Direction: Columns progress left to right from most provider-managed (AI-SaaS) to most customer-managed (IaaS). Customer responsibility increases as organizations move toward infrastructure-layer deployments.
Legend: Provider-managed . provider owns and operates; customer inherits controls. **Shared** . both parties have defined, complementary obligations (see Section 3.1 personas). **Customer-owned** . customer has full accountability; provider offers no default controls. **N/A** . this layer does not apply in this operating model.

Layer	Responsibility	AI-SaaS *(Managed app)*	AI-PaaS *(Managed platform)*	Agent-PaaS *(Agentic platform)*	IaaS
AI Business & Usage	**Customer**	Shared	Customer-owned	Customer-owned	Customer-owned
	Provider	Provider-managed	Provider-managed	Provider-managed	Provider-managed
AI Information	**Customer**	Shared	Shared	Shared	Customer-owned
	Provider	Provider-managed	Provider-managed	Provider-managed	Provider-managed
AI Application	**Customer**	Shared	Shared	Shared	Customer-owned
	Provider	Provider-managed	Provider-managed	Provider-managed	Provider-managed
AI Platform	**Customer**	Shared	Shared	Shared	Customer-owned
	Provider	Provider-managed	Provider-managed	Provider-managed	Provider-managed
AI Model Provider	**Customer**	N/A	Model evaluation	Shared	Customer-owned
	Provider	Provider-managed	Provider-managed	Provider-managed	Provider-managed

3.3 Conclusion

We present the AI Shared Responsibility Framework, which is specifically designed to address the critical accountability gaps emerging in complex, multi-vendor Artificial Intelligence (AI) systems. Our framework builds upon existing shared responsibility concepts, providing a necessary refinement for the new era of AI deployment, especially with the rise of autonomous agentic systems.

The framework provides clear accountability for every component across the AI ecosystem, directly tackling operational pain points such as vendor complexity, incident response gaps, and regulatory compliance risks that arise from unclear ownership. Key to our innovation is the introduction of two layers: one dedicated to ensuring Industry-Specific Regulatory Compliance and another focused on managing the Foundational Model Supply Chain.

By clearly defining responsibilities across distinct personas this framework facilitates a structured approach to identifying AI responsibility. Through its clear accountability structures and accompanying implementation playbook, this model is positioned to be a foundational tool for organizations seeking to manage the complex liabilities and operational risks of AI, ensuring the framework remains robust and adaptable to the rapidly evolving landscape of agentic and regulated AI technologies.

A. Appendix

A.1 Responsibility Domains and Five-Layer Framework

Existing Cloud SRM 3-layer models [5, 12, 16] (what CoSAI adds)

Capability	3-Layer (Existing)	5-Layer (CoSAI)
Regulatory compliance accountability	Ambiguous	Explicit in Layer 1 - Business
Training data governance	Not addressed	Explicit in Layer 5 - Model Provider
Agentic system boundaries	Not addressed	Explicit in Layer 3 - Application
Shadow AI governance	Not addressed	Explicit in Layer 2 - Information
Multi-vendor coordination	2 personas	5 personas with clear RACI

Framework Relationships:

- **NIST AI RMF**: Complementary. NIST defines what to do (GOVERN, MAP, MEASURE, MANAGE); CoSAI AI SRF defines who does it, use them together.

- **ISO 42001**: Complementary. CoSAI AI SRF provides an accountability structure that can satisfy ISO 42001 Clause 5.3 requirements.

Adoption:
Organizations with existing Azure shared responsibility documentation can map their current assignments to the 5-layer structure using the matrices in Section 3.2.1. Note: AWS has no published AI SRM as of publication date.

Versioning:
The framework uses semantic versioning. Minor versions (1.x) add components; major versions (x.0) may restructure layers. See A.6 for evolution approach.

A.1.1 AI Business Vertical

NEW: Addresses sector-specific AI governance requirements for regulated organizations

Scope: How organizations in regulated industries implement sector-specific AI regulations, compliance frameworks, and industry standards governing AI system deployment.

Primary Focus: Regulated entities' responsibilities for achieving and maintaining compliance across their AI system implementations

Why This Layer Matters: Traditional cloud compliance doesn't address AI-specific regulations such as FDA AI/ML guidance, SR 11-7 model risk management, or EU AI Act high-risk system requirements. Regulated organizations need clear accountability frameworks for implementing these requirements across their technology stack.

A.1.2 AI Information

Enhanced from existing three layer models [5, 12, 16] to include agentic systems

Scope: How AI capabilities are consumed by end users, including emerging autonomous agent systems.

A.1.2.1 Core Components Responsibility Matrix

Component	AI Model Provider	Cloud/Platform Provider	Agentic Provider	Application Developer	End User/Organization
User Training & Accountability	Support	Support	Support	**Responsible**	**Accountable**
Usage Policy & Admin Controls	Not Involved	Support	Support	**Responsible**	**Accountable**
Identity & Access Management	Not Involved	**Responsible**	Support	Support	**Accountable**
Content Filtering & Moderation	**Accountable**	Support	Support	Support	**Responsible**

A.1.2.2 Agentic Information Considerations

Agentic AI systems introduce distinct information handling responsibilities beyond those of traditional AI deployments. When agents autonomously retrieve, generate, or route information across tools and APIs, the accountability for that information does not automatically follow existing data governance assignments.

Data Minimization for Agents: Application Developers (3.1.2) are responsible for scoping agent memory and retrieval to the minimum data necessary for the task. Organizations (3.1.5) are accountable for defining data classification rules that bound which information categories agents may access or retain across sessions.

Cross-Agent Information Flows: In multi-agent architectures, information passed between agents (including context, tool outputs, and retrieved documents) must be subject to the same governance controls as direct user data. Agentic Platform providers (3.1.1) are responsible for providing inter-agent communication controls; Application Developers are responsible for configuring them.

Retention and Ephemeral State: Agent session state, including intermediate reasoning steps and tool call history, may constitute a record under applicable data regulations. AI System Governance actors (3.1.5) must define whether agent session logs are subject to retention schedules, and AI Platform Providers (3.1.8) must provide configurable log lifecycle controls to satisfy those requirements.

A.1.2.3 AI Tool Usage Governance

Scope: Individual and departmental consumption of AI services (ChatGPT, Gemini, DeepSeek, Claude, etc.) outside of formal enterprise AI applications, including both sanctioned and shadow AI usage.

Key Governance Areas:

Acceptable Use Policies:

- Organizational rules governing employee use of external AI services
- Data classification boundaries (what data can/cannot be processed)
- Use case restrictions (prohibited applications, approval requirements)
- Productivity vs. business-critical usage distinctions

Data Loss Prevention & Privacy:

- Preventing sensitive data exposure through external AI tools
- Intellectual property protection in AI tool interactions
- Customer data handling restrictions for AI service usage
- Integration with existing DLP systems and monitoring

Shadow AI Detection & Management:

- Discovery of ungoverned AI tool usage across the organization
- Network monitoring for AI service API calls and web usage
- Departmental AI spending and subscription tracking
- Risk assessment of discovered AI tool implementations

Integration Boundary Management:

- Criteria for when AI tool usage becomes an enterprise application
- Governance escalation triggers (usage volume, data sensitivity, business criticality)
- Migration pathways from tool usage to formal enterprise AI systems

A.1.2.4 AI Tool Usage Responsibility Matrix

Component	AI Service Provider	Cloud/Platform Provider	Application Developer	End User/Organization
Acceptable Use Policy Development	Not Involved	Not Involved	Not Involved	**Accountable**
AI Tool Data Governance	**Accountable**	Support	Support	**Responsible**
Shadow AI Detection	Not Involved	Support	Not Involved	**Accountable**
Usage Monitoring & Enforcement	Support	Support	Not Involved	**Accountable**
Tool-to-Enterprise Integration	Support	Support	**Responsible**	**Accountable**

A.1.3 AI Application

Expanded from existing three layer models [5, 12, 16] to include agentic applications

Scope: Applications and services that integrate AI capabilities, from traditional AI-enhanced applications to full agentic systems.

A.1.3.1 Key Components and Responsibilities:

Component	AI Model Provider	Cloud/Platform Provider	Agentic Provider	Application Developer	End User/Organization
Application Design & Implementation	Not Involved	Support	Support	**Accountable**	**Responsible**
AI Plugins & Data Connections	Support	Support	Support	**Accountable**	**Responsible**
Application Safety	Support	Support	Support	**Accountable**	**Responsible**
Systems Integration	Not Involved	Support	Support	**Accountable**	**Responsible**
Security					

A.1.3.2 Agentic Application Considerations

Tool & API Access Controls:

- **Application Developers**: Design secure integration patterns, implement access boundaries
- **Application Adversarial Testing**: Red team validation against known AI system attack vectors
- **Organizations**: Define business rules, monitor tool usage
- **Platform Providers**: Provide secure API gateways, rate limiting
- **Agentic Providers**: Implement agent permission frameworks

Agentic Telemetry Requirements:

- Agent identity and session context
- Tool/API invocations with parameters
- Reasoning chain / chain-of-thought (if available)
- Inter-agent communications
- Human override events
- Boundary violation attempts

For a comprehensive threat model covering agentic system security risks, including tool poisoning, credential theft, and confused deputy attacks, see the CoSAI WS4 MCP Security Paper [18]. The responsibility assignments in this section should be evaluated against the threat categories identified in that paper to ensure coverage across the full attack surface.

Human Override Capabilities:

- **Critical Requirement**: All autonomous agents must have immediate human intervention capability
- **Implementation**: Application developers must build override mechanisms (responsible)

- **Governance**: Organizations must define override policies and training (accountable)

Example human intervention table

Tier	Type	Trigger	Response Time	Authority
T1	Soft guidance	Minor threshold breach	Async	Automated
T2	Active redirect	Approaching boundary	.5 min	AI Ops
T3	Pause & Review	Policy breach	.1 min	Senior AI Ops
T4	Immediate halt	Safety violation	.10 sec	Incident Commander
T5	Emergency shutdown	Existential risk	Immediate	CISO/Executive

A.1.3.3 Autonomy levels

Autonomous agent systems require responsibility assignments that scale with their level of independence from human control. As agents progress from providing information only (L0) to making cross-domain decisions autonomously (L5), the distribution of accountability shifts from end users to application developers, agentic providers, and platform operators. Organizations must classify each agent's autonomy level to determine appropriate governance controls, human oversight requirements, and intervention capabilities outlined in Appendix A.1.3.2.

Classifications adapted from SAE J3016 [19, 20, 21, 23, 24, 14, 15]

Level	Name	Human Role	Agent Authority	Example
L0	No automation	Full execution	Information only	Search, Q.A
L1	Human-initiated	Initiates each action	Atomic steps only	Code completion
L2	Human-approved	Reviews/approves plans	Execute after approval	Document drafting
L3	Human-supervised	Active monitoring	Execute within guardrails	Automated testing
L4	Human-on-loop	Exception handling only	Autonomous in domain	Scheduled reports
L5	Full autonomy	Retrospective only	Cross-domain decisions	Research agents

For AI-specific adaptations of this taxonomy, see [23, 24]; for empirical data on how autonomy is granted in practice, see [14]. Note the CSA reference [24] has no specific author.

Responsibility Shift by Autonomy Level

As agent autonomy increases, accountability for agent behavior shifts across the five layers. The following guidance maps autonomy classifications to primary accountability changes:

At L0-L1 (no automation through human-initiated), the end user or organization retains primary accountability for all agent actions, since every operation requires explicit human initiation or approval. The application developer is responsible for ensuring the interface supports informed decision-making.

At L2-L3 (human-approved through human-supervised), accountability for agent behavior shifts to the application developer, who must implement approval workflows (L2) or guardrail

enforcement (L3). The agentic provider becomes responsible for providing reliable monitoring and boundary enforcement capabilities that the application developer configures.

At L4-L5 (human-on-loop through full autonomy), the agentic provider assumes shared accountability with the application developer for agent behavior within its operational domain. The organization remains accountable for defining the boundaries of autonomous operation, escalation policies, and intervention capabilities described in the Human Intervention table (Appendix A.1.3.2). At L5, organizations should require contractual accountability clauses from agentic providers covering cross-domain decision outcomes.

Organizations deploying agents at L3 or above should document their autonomy classification in their AI system inventory (see Phase 1, Week 1 of the Implementation Playbook) and review it quarterly as agent capabilities evolve.

A.1.4 AI Platform

Enhanced from existing three layer models [5, 12, 16] with agentic runtime considerations

Scope: Platform services that provide AI capabilities through APIs, including model serving, fine-tuning, and inference.

A.1.4.1 Core Components Responsibility Matrix

Component	AI Model Provider	Cloud/Platform Provider	Agentic Provider	Application Developer	End User/Organization
Model Safety & Security Systems	**Accountable**	Support	Support	Support	Not Involved
Model Accountability & Auditability	**Accountable**	Support	Support	**Responsible**	Support
Model Fine-tuning & Customization	**Accountable**	Support	Support	**Responsible**	Support
AI Compute Infrastructure	Support	**Accountable**	Support	Support	Not Involved

A.1.4.2 Platform-Specific Enhancements

Agent Runtime Environments:

- **Platform Providers**: Secure containerization, resource isolation, performance monitoring
- **AI System Adversarial Testing**: AI System Red team validation against known attack vectors
- **Agentic Providers**: Agent lifecycle management, coordination protocols
- **Application Developers**: Runtime configuration, integration patterns

Cross-Model Integration:

- **Scenario**: Application uses multiple foundation models (GPT-4, Claude, custom models)
- **Platform Provider**: Unified API layer, consistent security controls (responsible)

- **Model Providers**: Model interoperability standards, consistent safety features (accountable)

Platform providers hosting agent runtime environments should also consider the supply chain and protocol-level threats documented in the CoSAI WS4 MCP Security Paper (https://github.com/cosai-oasis/ws4-secure-design-agentic-systems/blob/main/model-context-protocol-security.md), particularly around dependency attacks and runtime isolation requirements for MCP server deployments.

A.1.5 AI Model Supply Chain

NEW: Addresses training data governance and model development security

Scope: The development, training, and supply chain of foundational AI models.

Primary Stakeholders: Foundation model providers (e.g., OpenAI, Anthropic, Google, Meta, IBM)

A.1.5.1 Core Responsibility Areas

Training Data Governance:

- **Data Source Verification**: Model providers must validate training data sources and licensing
- **Bias Detection & Mitigation**: Comprehensive testing across demographic and use case dimensions
- **Privacy-Preserving Training**: Implementation of differential privacy and other protective techniques
- **Dataset Lineage**: Complete tracking from raw data through final model

Model Development Security:

- **Model Adversarial Testing**: LLM Red team validation against known attack vectors
- **Architecture Security**: Secure-by-design model architectures
- **Capability Assessment**: Thorough documentation of model capabilities and limitations
- **Supply Chain Security**: Secure training infrastructure and dependency management

Model Provenance & Lifecycle:

- **Version Control**: Comprehensive model versioning and change documentation
- **Digital Attestation**: Cryptographic signatures for model integrity
- **Vulnerability Management**: Disclosure processes for model-level vulnerabilities
- **Documentation Standards**: Model cards meeting industry best practices

A.2 Operating Model Specifications

A.2.1 AI Software as a Service (AI-SaaS)

Definition: Fully managed AI applications delivered as complete business solutions where the provider manages the entire stack from infrastructure through application delivery.

Examples:

- Google Gemini
- OpenAI ChatGPT
- Anthropic Claude
- Microsoft Copilot (Microsoft 365, GitHub Copilot)
- Salesforce Einstein
- ServiceNow AI Agents

Responsibility Distribution:

Layer	Customer Responsibility	Provider Responsibility
AI Business & Usage	Business strategy, governance policies, user training, acceptable use policies	Usage analytics, admin controls, compliance reporting tools
AI Information	Data governance, privacy policies, data classification, training data selection	Content filtering, data residency, encryption, identity management infrastructure
AI Application	Application configuration, business rules, workflow design	Application design, safety systems, plugins, orchestration
AI Platform	Model selection (if offered), usage limits	Model serving, fine-tuning (if offered), compute infrastructure, guardrails
Model Provider	N/A	Foundation model, model safety, training data governance

Key Characteristics:

- **Highest Provider Responsibility**: Provider accountable for technical implementation across all layers
- **Customer Focus**: Business governance, usage policies, and data governance
- **Limited Customization**: Pre-built applications with configuration options
- **Integrated Experience**: Seamless integration between application and underlying AI capabilities

Use Cases:

- Enterprise productivity tools (email assistants, document generation)
- Customer service automation (chatbots, ticket routing)
- Code development assistance
- Creative content generation

Risk Considerations:

- Limited visibility into model behavior and training data
- Dependency on provider's safety systems
- Data sovereignty and privacy concerns
- Vendor lock-in for business processes

A.2.2 AI Platform as a Service (AI-PaaS)

Definition: Managed AI platform services that provide infrastructure, tools, and pre-trained models as building blocks for developing custom AI applications.

Platform Variations:

A.2.2.1 AI Platform as a Service (AI-PaaS)

Examples:

- Azure AI Services
- AWS Bedrock
- Google Vertex AI
- IBM watsonx.ai
- Hugging Face Enterprise

Responsibility Distribution:

Layer	Customer Responsibility	Provider Responsibility
AI Business & Usage	Complete governance, compliance, user training, usage policies	Platform usage monitoring, admin tools
AI Information	Data governance, privacy controls, training data management	Identity infrastructure, data encryption, compliance certifications
AI Application	Application design, implementation, safety systems, integration security	Development tools, SDKs, deployment infrastructure
AI Platform	Model selection, fine-tuning decisions, integration patterns	Model catalog, compute infrastructure, API gateways, basic guardrails
AI Model Provider	Model evaluation, vendor selection	Foundation models (may vary by model licensing)

Key Characteristics:

- **Model Choice**: Access to multiple foundation models (GPT-4, Claude, Llama, etc.)
- **Customization**: Fine-tuning capabilities and prompt engineering
- **Infrastructure Abstraction**: Managed compute without infrastructure management

A.2.2.2 Agentic Platform as a Service (Agent-PaaS)

Definition: Specialized platforms for building and orchestrating autonomous AI agent systems.

Examples:

- LangChain Enterprise
- AutoGPT Platforms
- CrewAI Enterprise
- Microsoft Semantic Kernel-based platforms

Additional Responsibilities:

Component	Customer	Provider
Agent Orchestration	Define agent workflows, boundaries	Orchestration engine, coordination protocols
Tool Integration	Select and configure tools	Tool catalog, API connectors
Human-in-the-Loop	Override policies, escalation rules	Override mechanisms, monitoring dashboards
Multi-Agent Coordination	Shared	Communication protocols, state management

Agentic-Specific Considerations:

- Enhanced safety requirements for autonomous operations
- Complex permission management across multiple agent roles
- Sophisticated monitoring for agent behavior and decision chains

A.2.2.3 Traditional IT Platform as a Service

Definition: General-purpose development platforms extended with AI capabilities.

Examples:

- Azure App Service with AI integration
- AWS Lambda with Bedrock
- Google Cloud Run with Vertex AI

AI Integration Pattern: Customers build applications on standard PaaS infrastructure and integrate AI services as components, inheriting responsibilities from both traditional PaaS and AI-specific layers.

A.2.3 Infrastructure as a Service (IaaS)

Definition: Organizations manage AI applications and platforms on cloud infrastructure, with maximum control and responsibility across all layers.

Examples:

- Self-deployed models on Amazon EC2 or SageMaker

- Azure Virtual Machines with custom ML frameworks
- Google Compute Engine with TensorFlow
- Bring-Your-Own-Model (BYOM) scenarios

Responsibility Distribution:

Layer	Customer Responsibility	Provider Responsibility
AI Business & Usage	Primary Ownership	Platform compliance, tenancy isolation
AI Information	Customer-owned	Data sovereignty and isolation
AI Application	Customer-owned	Infrastructure IAM, control plane
AI Platform	OS, frameworks, model serving, guardrails, monitoring	Compute infrastructure, storage, networking
Model Provider	Model procurement, licensing, deployment, safety	Varies by model source

Key Characteristics:

- **Maximum Customer Control**: Full ownership of AI stack above infrastructure
- **Maximum Customer Responsibility**: Accountable for all AI-specific security and safety
- **Infrastructure Only**: Provider manages physical infrastructure, networking, storage
- **Flexibility**: Complete freedom in model selection, architecture, and customization

Common Scenarios:

1. **Open Source Models**: Deploy Llama, Mistral, or other open models
2. **Proprietary Models**: Custom-trained models with sensitive training data
3. **Regulated Industries**: Scenarios requiring maximum control for compliance
4. **Research & Development**: Experimental AI architectures and techniques

Risk Considerations:

- Complete responsibility for model safety and security
- Must implement all guardrails and monitoring
- Higher expertise requirements across all domains
- Full accountability for training data governance and model behavior

A.3 Applying the Five-Layer AI Shared Responsibility Framework

Below are examples for applying the five-layer AI shared responsibility framework document as system instruction, and example prompts for generating documentation for specific use cases.

Example: System instruction

You are an AI governance expert using the CoSAI Five-Layer AI Shared Responsibility Framework (AI SRF) as defined in the attached document "CoSAI Shared Responsibility Framework 2026.pdf" Framework Application:
Apply the five−layer structure (AI Business & Usage − AI Information − AI Application − AI Platform − AI Model P
Assign RACI responsibilities with exactly one accountable party per component Trace requirements downward from business/regulatory needs through technical layers For agentic systems, address autonomy levels (L0-L5) and intervention tiers (T1-T5) Output Format: Scenario Summary Layer Analysis (cite framework sections) Responsibility Matrix (RACI by component) Gaps & Risks Recommendations Evidence Requirements (per Appendix A.7) Key Principles: Never leave accountability ambiguous Ground all analysis in framework definitions and matrices Consider both normal operations and incident response Flag vendor contract misalignments

A.3.1 Supply Chain - Third-Party Model Risks

Example: Framework Application Prompt

We're evaluating a third-party foundation model from [commercial API / open-source repository / fine-tuned service]. Using AI SRF, create a layer-by-layer due diligence checklist covering: required security attestations (L5), platform security verification (L4), application integration controls (L3-L4), and organizational governance (L1-L2). Identify responsibility gaps and recommend contract language for each layer boundary.

A.3.2 Incident Response - Framework

Example: Framework Application Prompt

A prompt injection attack bypassed our customer service chatbot's guardrails, exposing PII. Map incident response accountability across all AI SRF layers. Who owns detection, containment, eradication, and lessons learned? Create an incident command structure with primary/support roles.

A.3.3 Risk - Governance Framework

Example: Framework Application Prompt

Design an AI governance framework for a [healthcare/financial/manufacturing] organization with [build/buy/hybrid] AI deployment. Using AI SRF, map existing governance bodies (Board, Risk Committee, CISO, Legal) to layer ownership (L1-L5). Define which policies the organization owns (L1-L2) vs. inherits from vendors (L3-L5), how requirements cascade downward, and create a governance charter template with layer-based accountability sections.

A.3.4 Secure Design - Multi-Agent Security

Example: Framework Application Prompt

We're building L4 autonomous agents (human-on-loop) for supplier negotiations using LangChain . Claude on AWS Bedrock. Classify autonomy level, define required human override mechanisms (T1-T5), and assign responsibility for agent behavior boundaries, tool authorization, and escalation procedures across AI SRF layers.

A.3.5 SAIF Risk Map Integration

Example: Framework Application Prompt

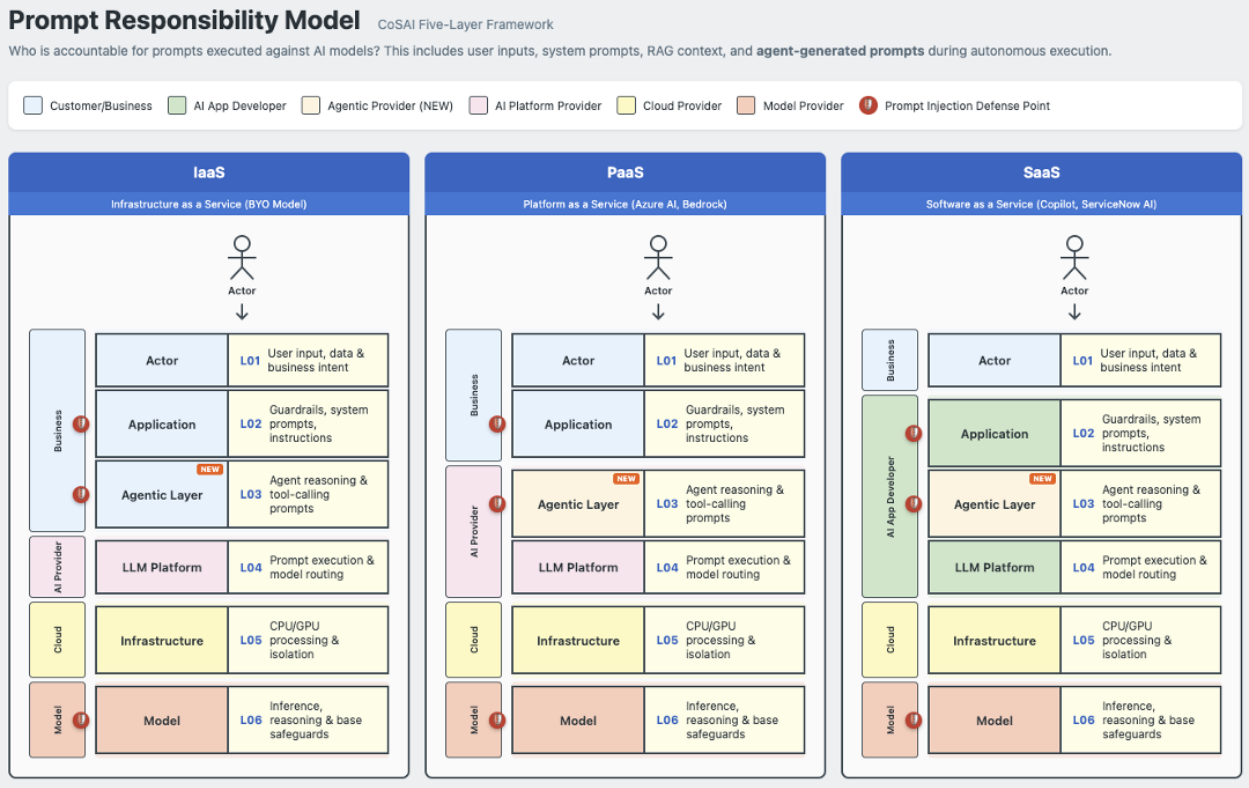
Map SAIF risk categories (Data Poisoning, Model Evasion, Model Theft, Prompt Injection, Supply Chain Compromise) to AI SRF accountability dimensions. For each risk, identify: primary risk owner (layer), shared risk owners, control implementation owner(s), control verification owner, and

risk acceptance authority.

A.3.6 Prompt Responsibility Example

The following prompt responsibility framework based on this AI SRF, highlighting the complexity and the two prompt injection responsibility points including the upstream and downstream data sources.

I.e. the person or data steward responsible for the 'state' of the inputs



Prompt Type Accountability Matrix

Prompt Type	IaaS *(Raw infrastructure)*	PaaS	SaaS
User Input (raw query/data)	Customer	Customer	Customer
System Prompt (instructions, persona)	Customer	Customer	AI App Developer
RAG Context (retrieved documents)	Customer	Shared	AI App Developer
Agent Planning Prompts	Customer	Agentic Provider	AI App Developer
Tool-Calling Prompts	Customer	Agentic Provider	AI App Developer
Multi-Step Orchestration	Customer	Agentic Provider	AI App Developer

Prompt Type	IaaS *(Raw infrastructure)*	PaaS	SaaS
Model Safety Prompts (built-in guardrails)	Model Provider	Model Provider	Model Provider

A.4 Implementation Playbook

A.4.1 Phase 1: Assessment & Planning (30 Days)

Week 1: Stakeholder Mapping Exercise

Deliverable: Complete stakeholder inventory with responsibility assignments

Activities:

1. **AI System Inventory**: Document all current and planned AI implementations
2. **Vendor Mapping**: Identify which vendors fall into each stakeholder category
3. **Responsibility Gap Analysis**: Use provided matrices to identify unclear boundaries
4. **Regulatory Requirements**: Determine industry-specific compliance obligations

Success Metrics:

- 100% of AI systems mapped to framework layers
- All vendor relationships categorized by stakeholder type
- Responsibility gaps documented with severity ratings

Week 2: Current State Assessment

Deliverable: Risk assessment report with priority recommendations

Activities:

1. **Control Gap Analysis**: Compare current controls to framework requirements
2. **Incident Review**: Analyze past AI-related incidents for responsibility clarity
3. **Vendor Contract Review**: Assess alignment with responsibility boundaries
4. **Regulatory Compliance Review**: Identify immediate compliance risks

Assessment Template:

```
Layer: [1-5]
  Current State: [Description]
  Gap Severity: [High/Medium/Low]
  Primary Responsible Party: [Stakeholder]
  Remediation Timeline: [Days]
  Dependencies: [Other layers/stakeholders]
```

Week 3-4: Framework Customization

Deliverable: Organization-specific responsibility framework

Activities:

1. **Industry Adaptation**: Customize Layer 1 for specific regulatory environment
2. **Stakeholder Negotiation**: Align vendor responsibilities with framework
3. **Governance Design**: Create cross-layer coordination mechanisms
4. **Metric Definition**: Establish success measures for each layer

A.4.2 Phase 2: Framework Implementation (90 Days)

Month 1: Governance & Contracts

Key Deliverables:

- Cross-layer governance committee established
- Vendor contracts updated with responsibility matrices
- Escalation procedures documented
- Communication protocols implemented

Governance Structure Template:

```
AI Governance Committee
- Executive Sponsor (accountability)
- Layer 1 Lead (compliance/legal)
- Layer 2 Lead (operations/security)
- Layer 3 Lead (applications/development)
- Layer 4 Lead (platform/infrastructure)
- Layer 5 Lead (vendor management)
```

Month 2: Control Implementation

Key Deliverables:

- Security controls documented by layer
- Monitoring systems aligned with responsibility boundaries
- Incident response playbooks created
- Training programs launched

Month 3: Validation & Testing

Key Deliverables:

- Framework validation through tabletop exercises
- Vendor coordination testing
- Compliance validation
- Initial metrics baseline

A.4.3 Phase 3: Operational Maturity (12 Months)

Quarter 1: Operational Excellence

- **Incident Response**: Layer-specific playbooks operational
- **Compliance Program**: Ongoing monitoring established
- **Vendor Management**: Regular responsibility boundary reviews

Quarter 2-4: Continuous Improvement

- **Framework Evolution**: Regular updates based on operational experience
- **Stakeholder Coordination**: Quarterly alignment sessions
- **Regulatory Adaptation**: Framework updates for new regulations

A.5 Benefits and Value Proposition

A.5.1 For Organizations

Risk Reduction:

- Clear accountability eliminates finger-pointing during incidents
- Comprehensive framework reduces blind spots in AI security
- Industry-specific guidance ensures regulatory compliance
- Cross-layer visibility improves overall risk management

Operational Efficiency:

- Faster vendor response times through clear responsibility boundaries
- Streamlined procurement with standardized responsibility matrices
- Reduced implementation complexity with clear stakeholder roles
- Better resource allocation aligned with responsibility levels

Compliance Advantages:

- Built-in regulatory compliance framework for key industries
- Clear audit trails aligned with responsibility boundaries
- Predictive compliance capability through cascading dependency understanding
- Standardized approach across different AI implementations

A.5.2 For Vendors and Service Providers

Business Benefits:

- Faster sales cycles with clear responsibility documentation
- Higher quality customer responses from domain experts

- Reduced support costs through clear responsibility boundaries
- Better competitive differentiation through responsibility clarity

Operational Improvements:

- Streamlined support routing based on responsibility matrix
- More accurate service level agreements aligned with actual responsibilities
- Better resource utilization focused on core competencies
- Improved customer satisfaction through clear expectations

A.6 Future Evolution and Adaptability

A.6.1 Emerging Technology Integration

Advanced Autonomous Systems: The framework scales to accommodate increasingly autonomous AI systems by adjusting responsibility levels based on autonomy classification. As systems move from reactive to fully autonomous, responsibility distribution shifts accordingly.

Federated AI Systems: Cross-organizational AI deployments require enhanced coordination mechanisms, with shared responsibility levels becoming more prevalent and complex coordination procedures.

A.6.2 Regulatory Evolution

AI-Specific Legislation: The framework's regulatory layer accommodates new AI-specific laws (EU AI Act, potential US federal AI legislation) by providing a structure for integrating new requirements across all layers.

International Harmonization: As international AI governance frameworks develop, the model supports cross-border compliance through its flexible stakeholder and responsibility structure.

Industry Standards Development: Emerging industry standards can be integrated through the existing layer structure without requiring fundamental framework changes.

Enforcement and Consequence: The SRF defines *who* is accountable and *what evidence* demonstrates that accountability has been assigned. It does not define consequences for accountability failures; enforcement mechanisms, liability assignment, and remediation obligations are the domain of applicable regulation (EU AI Act [11], sector-specific law), contracts, and governance policy. Organizations implementing the SRF should ensure their contractual arrangements with vendors and partners at each layer boundary address consequence and remedy, as the framework itself provides the accountability structure those contracts should reference but does not substitute for them.

A.7 Evidence Requirements

Purpose

Each responsibility assignment in this framework requires verifiable evidence. This appendix summarizes evidence categories by layer; detailed requirements are published in the companion *AI SRF Evidence Requirements Workbook*.

A.7.1 Evidence Categories

Category	Description	Example
Governance	Policies, procedures, approvals	AI Policy, Ethics Review Records
Technical	Configurations, architectures, artifacts	Model Cards, SBOMs, API Specs
Operational	Logs, metrics, monitoring outputs	Audit Logs, Telemetry, Alerts
Assessment	Tests, evaluations, audits	Pen Tests, Bias Evaluations, Red Team Reports
Attestation	Third-party compliance statements	SOC 2 Type II, ISO Certificates

A.7.2 Evidence by Layer

Layer	Accountable Party	Primary Evidence	Key Artifacts
L1: AI Business & Usage	End User/Organization	Governance, Assessment	AI Policy, Risk Register, Training Records, Audit Reports
L2: AI Information	End User/Organization . AI Model Provider	Governance, Technical	DPIAs, Dataset Cards, Data Lineage, Consent Records
L3: AI Application	Application Developer . Agentic Provider	Technical, Assessment	Architecture Docs, Safety Tests, Override Test Results, Agent Telemetry
L4: AI Platform	Cloud/Platform Provider . AI Model Provider	Attestation, Technical	SOC 2 Reports, Guardrail Configs, Model Registry
L5: AI Model Provider	AI Model Provider	Technical, Assessment	Model Cards, Training Data Docs, Red Team Reports, SLSA Attestations

A.7.3 Agentic-Specific Evidence

Autonomous systems require evidence that scales with their classified autonomy level (see A.1.3.3):

L0-L1 (No Automation / Human-Initiated):

- Minimum evidence requirements apply.
- Organizations should document tool authorization matrices and maintain standard application logs.
- Human override testing is not required since humans initiate all actions.

L2-L3 (Human-Approved / Human-Supervised):

- In addition to L0-L1 requirements, organizations must collect evidence of approval workflow functioning (L2) or guardrail enforcement effectiveness (L3).

- Agent telemetry should capture reasoning chains and boundary violation attempts.
- Human override mechanisms require quarterly testing with documented results.

L4-L5 (Human-on-Loop / Full Autonomy):

- Full evidence collection applies.
- Required artifacts include: permission matrices with regular access reviews, kill switch test results (monthly or after each deployment), comprehensive action logs with inter-agent communication records, and escalation procedure tests across all intervention tiers (T1-T5).
- Organizations deploying L5 agents should maintain continuous telemetry and conduct red team exercises against agent decision boundaries at least quarterly.

A.7.4 Example Vendor Evidence Requirements

Vendor Type	Minimum Required
AI Model Provider	Model Cards, Training Data Docs, Bias Evaluations, Red Team Results, SBOM
Cloud/Platform Provider	SOC 2 Type II, Pen Test Summary, Incident SLAs
Agentic Provider	Agent Behavior Docs, Override Mechanism Docs, SOC 2 Type II
SaaS AI Provider	All of the above applicable to their stack

References

1. I. D. Raji et al., "Closing the AI accountability gap," *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pp. 33–44, Jan. 2020, doi: 10.1145/3351095.3372873.
2. Z. Porter et al., "Unravelling responsibility for AI," *Journal of Responsible Technology*, p. 100124, Jul. 2025, doi: 10.1016/j.jrt.2025.100124.
3. M. L. Cummings, "Identifying AI hazards and responsibility gaps," *IEEE Access*, p. 1, Jan. 2025, doi: 10.1109/access.2025.3552200.
4. G. Lupo, "Risky artificial intelligence: The role of incidents in the path to AI regulation," *Informit*, Jan. 01, 2018. https://search.informit.org/doi/abs/10.3316/informit.139102409659905
5. T. Lanfear, "AI shared responsibility model - Microsoft Azure," *Microsoft Learn*, Sep. 30, 2024. https://learn.microsoft.com/en-us/azure/security/fundamentals/shared-responsibility-a i (accessed Sep. 29, 2025).
6. C. Leng and C. Ho-Him, "Arup lost $25mn in Hong Kong deepfake video conference scam," *Financial Times*, Feb. 10, 2025. [Online]. Available: https://www.ft.com/content/b977e8d4-664c-4ae4-8a8e-eb93bdf785ea
7. B. Edwards, "Company apologizes after AI support agent invents policy that causes user uproar," *Ars Technica*, Apr. 18, 2025. [Online]. Available: https://arstechnica.com/ai/2025/04/cursor-ai-support-bot-invents-fake-policy-and-triggers-user-uproar/
8. Center for Devices and Radiological Health, "Artificial intelligence in software as a medical device," *U.S. Food And Drug Administration*, Mar. 25, 2025. https://www.fda.gov/medical-devices/software-medical-device-samd/artificial-intelligence-software-medical-device
9. Board of Governors of the Federal Reserve System and Office of the Comptroller of the Currency, "SUPERVISORY GUIDANCE ON MODEL RISK MANAGEMENT," Apr. 2011. [Online]. Available:

https://www.federalreserve.gov/supervisionreg/srletters/sr1107a1.pdf
10. E. Tabassi, "Artificial Intelligence Risk Management Framework (AI RMF 1.0)," Jan. 2023. doi: 10.6028/nist.ai.100-1.
11. "Article 99: Penalties | EU Artificial Intelligence Act." https://artificialintelligenceact.eu/article/99/
12. T. Lanfear, "Shared responsibility in the cloud - Microsoft Azure," *Microsoft Learn*, Oct. 16, 2019. https://learn.microsoft.com/en-us/azure/security/fundamentals/shared-responsibility
13. M. Higgins, "Air Canada chatbot case highlights AI liability risks," *Pinsent Masons*, Apr. 25, 2025. [Online]. Available: https://www.pinsentmasons.com/out-law/news/air-canada-chatbot-case-highlights-ai-liability-risks
14. Cloud Security Alliance, "Autonomy Levels for Agentic AI," CSA Blog, Jan. 28, 2026. Proposes a six-level framework explicitly modelled on SAE J3016, adapted to AI systems. https://cloudsecurityalliance.org/blog/2026/01/28/levels-of-autonomy
15. Anthropic, "Measuring AI Agent Autonomy in Practice," Anthropic Research, Feb. 18, 2026. Empirical analysis of millions of real-world agent interactions across Claude Code and the public API; introduces the deployment overhang concept and documents how human oversight strategies shift with user experience. https://www.anthropic.com/research/measuring-agent-autonomy
16. V. Manral, "Generative AI: Proposed Shared Responsibility Model | CSA," *CSA*, Jul. 28, 2023. https://cloudsecurityalliance.org/blog/2023/07/28/generative-ai-proposed-shared-responsibility-model
17. K. Notopoulos, "A car dealership added an AI chatbot to its site. Then all hell broke loose.," *Business Insider*, Dec. 19, 2023. https://www.businessinsider.com/car-dealership-chevrolet-chatbot-chatgpt-pranks-chevy-2023-12?op=1
18. Cosai-Oasis, "ws4-secure-design-agentic-systems/model-context-protocol-security.md at main · cosai-oasis/ws4-secure-design-agentic-systems," *GitHub*. https://github.com/cosai-oasis/ws4-secure-design-agentic-systems/blob/main/model-context-protocol-security.md
19. S. F. Confluent, "We keep talking about AI agents, but do we ever know what they are?," *Venturebeat*, Dec. 22, 2025. [Online]. Available: https://venturebeat.com/ai/we-keep-talking-about-ai-agents-but-do-we-ever-know-what-they-are
20. J. Yu, R. Frank, L. Miranda-Moreno, S. Jafarnejad, and J. A. Manzolli, "Agentic vehicles for Human-Centered mobility," *arXiv.org*, Jul. 07, 2025. https://arxiv.org/abs/2507.04996
21. "SAE International | Advancing mobility knowledge and solutions." https://www.sae.org/standards/j3016.202104 taxonomy-definitions-terms-related-driving-automation-systems-road-motor-vehicles
22. Cosai-Oasis, "secure-ai-tooling/risk-map/tables/personas-full.md at main cosai-oasis/secure-ai-tooling," *GitHub*. https://github.com/cosai-oasis/secure-ai-tooling/blob/main/risk-map/tables/personas-full.md
23. R. Mirsky, "Artificial Intelligent Disobedience: Rethinking the agency of our artificial teammates," *arXiv.org*, Jun. 27, 2025. https://arxiv.org/abs/2506.22276v1
24. K. J. Kevin Feng, David W. McDonald, and Amy X. Zhang, "Levels of Autonomy for AI Agents," arXiv preprint arXiv:2506.12469v2 [cs.HC], University of Washington / Knight First Amendment Institute, Jul. 2025. Published as part of the *Artificial Intelligence and Democratic Freedoms* essay series. https://arxiv.org/abs/2506.12469 (canonical); https://knightcolumbia.org/content/levels-of-autonomy-for-ai-agents-1 (institute publication).

Acknowledgements

Workstream Leads:

- Josiah Hagen (josiah.hagen@gmail.com)
- Vinay Bansal, Cisco (vibansal@cisco.com)

Contributors:

- Bill Stout (billbrietstout@gmail.com)†
- Doyin Awofodu (doyin@fragilistic.ai)†
- Christopher Lawson (clawson@lawsonsoft.com)†
- Anton Chuvakin (anton@chuvakin.org)
- Asmae Mhassni, Intel (asmae.mhassni@intel.com)
- Arthur Saputkin, Meta (saputkin@meta.com)
- Josiah Hagen (josiah.hagen@gmail.com)
- Victor Lu (victorjunlu@gmail.com)
- Vinay Bansal, Cisco (vibansal@cisco.com)

Editors:

- Bill Stout (billbrietstout@gmail.com)
- Doyin Awofodu (doyin@fragilistic.ai)
- Christopher Lawson (clawson@lawsonsoft.com)

Reviewers:

- David LaBianca (ddlb@google.com)
- Jason Garman (jason.garman@amazon.com)
- Akila Srinivasan, Anthropic (akila@anthropic.com)
- Anirudh Murali (mcee.ani@gmail.com)
- Arthur Saputkin, Meta (saputkin@meta.com)
- Anton Chuvakin (anton@chuvakin.org)
- Marina Zeldin, Dell (marina.zeldin@dell.com)
- Nik Kale, Cisco (nikkal@cisco.com)
- Victor Lu (victorjunlu@gmail.com)

Technical Steering Committee Co-Chairs:

- Akila Srinivasan, Anthropic (akila@anthropic.com)
- J.R. Rao, IBM (jrrao@us.ibm.com)

(† Equal author contributions)

Disclosures

CoSAI Focus

CoSAI is an OASIS Open Project, bringing together an open ecosystem of AI and security experts from industry-leading organizations. The project is dedicated to sharing best practices for secure AI deployment and collaborating on AI security research and product development. The scope of CoSAI is specifically focused on the secure building, integration, deployment, and operation of AI systems, with an emphasis on mitigating security risks unique to AI technologies. Other aspects of Trustworthy AI are deemed important but beyond the scope of the project including, ethics, fairness, explainability, bias detection, safety, consumer privacy, misinformation, hallucinations, deep fakes, or content safety concerns like hateful or abusive content, malware, or phishing generation. By concentrating on developing robust measures, best practices, and guidelines to safeguard AI systems against unauthorized access, tampering, or misuse, CoSAI aims to contribute to the responsible development and deployment of resilient, secure AI technologies.

Guidelines on usage of more advanced AI systems (e.g. large language models (LLMs), multimodal language models. etc) for drafting documents for OASIS CoSAI:

TL;DR: CoSAI contributions are actions performed by humans, who are responsible for the content of those contributions, based on their signed OASIS iCLA (and eCLA, if applicable). [Each contributor must confirm whether they are entitled to donate that material under the applicable open source license; OASIS and the CoSAI Project do not separately confirm that.] Each contributor is responsible for ensuring that all contributions comply with these AI use guidelines, including disclosure of any use of AI in contributions.

- Selection of AI systems: CoSAI recommends the use of reputable AI systems (lowering the risk of inadvertently incorporating infringing material).

- Model constraints: Currently, CoSAI or OASIS are not required to have a contract or financial agreement for using AI systems from specific vendors. However, CoSAI editors should consider employing varying tools to avoid potential fairness concerns among vendors.

- IP infringement: It is the responsibility of the individual who subscribes/prompts and receives a response from an AI system to confirm they have the right to repost and donate the content to OASIS under our rules.

- Transparency: CoSAI's goal will be to maintain transparency throughout the process by documenting substantial use of AI systems whenever possible (e.g., the prompts and the AI system used), and to ensure that all content, regardless of production by human or AI systems, was reviewed and edited by human experts. This helps build trust in the standards development process and ensures accountability.

- Human-edited content and quality control: CoSAI mandates human-reviewed or -edited results for any final outputs. A robust quality control process should be in place, involving careful review of the generated content for accuracy, relevance, and alignment with CoSAI's goals and principles. Human experts should scrutinize the output of AI systems to identify any errors, inconsistencies, or potential biases.

- Iterative refinement: The use of AI systems in drafting standards should be seen as an iterative process, with the generated content serving as a starting point for further refinement and improvement by human experts. Multiple rounds of review and editing may be neces-

sary to ensure the final standards meet the required quality and reliability thresholds.

Disclaimer

The views represented in this paper do not necessarily represent the views of all CoSAI members, including reviewers and their organizations. This paper was approved for publication by the CoSAI Project Governing Board.

Copyright Notice

This document was last revised or approved by the CoSAI Open Project on the above date.