Meeting Notes/Running Notes

March 3, 2026

- Welcome
- Co-Chair Volunteers
    - Ty Smith - Hashgraph/ Hedera
- Logistics
    - Meeting Timing and Frequency
        - Asynchronous communications?
            - Florencio: +1
            - Florencio: -1 discord; +1 the rest
- Discuss Next Steps for the WG
    - Finalize WG mission and scope areas
    - Decide what the final contribution of the WG will be (whitepaper, etc.)
    - Develop a WG Charter and submit to Governing Board for review/approval
- Continue discussion topics from last meeting (I tried to group these into themesy

- Mission out:
    - “The y advances the Agentic AI Foundation's mission by creating open standardization for secure execution and management of data, and privacy of information used by agentic systems.”
        - Look up Ty’s sentence at 49 min in from recording
-

Theme A: Agent Identity, Authentication & Authorization

- Attestation of agent and model in use (incl. zero-knowledge approaches)
- Agent Identity and Delegated Authorization
    - Capability-based auth model (beyond OAuth)
    - Agent ↔ MCP server authentication
    - Agent ↔ infrastructure representation
    - Agent ↔ agent authentication & authorization
- User/agent permission boundaries — restricting and auditing what each can do independently and together
- Tool-level access control — ensuring agents only invoke authorized tools

Theme B: Threat Modeling & Attack Surface

- Agent Threat Model Framework (attack taxonomy, red-team scenarios, testing harness)
- Agent + Model attack surface (model weaknesses → agent vulnerabilities)
- Attack Surface Management / Continuous Threat Exposure Management
- Advising on multi-agent architectures within a single application

Theme C: Prompt Injection & Input/Output Security

- Prompt injection mitigation — need for standardization
    - Florencio: some options to research Dual-LLM, CaMeL, prompt sandwiching
- Social engineering resistance
- Testing corpus for prompt injection
- Agent middleware as guardrails (inter-step validation, not just I/O)
    - Ref: pydantic-ai-middleware
- Agentic security critic / verifiers as workflow plugins

Theme D: Data Protection & Memory Security / Privacy

- Secure Memory specification (data classification, crypto standards)
- Agentic AI memory layer security & standards
- PII obfuscation
    - Agent memory?
    - LLM provider inputs & outputs
    - Summarize historical lineage of agent execution with PII obfuscation built in as agent executes.
        - Ex: initial interaction includes age, etc. That gets summarized into something like “is user >21?” and then that gets passed through instead of the exact age.
- Memory access logs *(cross-ref Theme E)*
- Environment variables / API Keys / Encryption Keys
    - EX: Don’t leak api keys, etc
- Secret / sensitive data management
    - EX: Don’t leak model training data
    - EX: Don’t leak company secret data
- Identify different types of protected data in the context of AI agents
    - Produce recommendations for handling of different types
- Define data boundaries
- How to detect and protect against agent sabotage?
- How can we / should we protect data from a rogue agent?
    - What would be the blast radius of a rogue agent?

Theme E: Auditability, Observability & Forensics

- Decision traceability
- Tool invocation & audit logs
- Conversation traceability (associating tools + messages into a single context)
- Observability tooling for security layers
- Memory access logs *(shared with Theme D)*

Theme F: Evaluation, Compliance & Certification

- Eval framework for compliance/security of an agent
- Measuring the security posture of an AI agent
- Certification criteria for different levels of AI systems

Theme G: Operational Practices

- Best operational practices for agentic AI security
- Ref: Block GenAI Security Principles — Wes offered to refine with the group

Theme H: Commerce, Payments & Tooling

- Commerce & payments (open standards, DeFi/blockchain integration)
- Tooling & plugins — agent tool manifest
- Enforcing security guardrails on commerce

- Use cases
  -
- Scope (in/out)
  - OUT - Ethics/Morality/Responsible/Other - Delegate to the Governance and Policy WG
  - ~~OUT - Hallucinations - Delegate to Accuracy and Reliability WG~~
  - OUT - Recovery from Model Poisoning - Accuracy and Reliability WG (PENDING WG Mission)
- Potential Deliverables
  - Work on initial taxonomy for our working group
    - Once solidified, work on strategy for getting a general AAIF taxonomy page for all working groups

- Prior work/other efforts
  -
- Taxonomy
  - Secret -
  - Security -
  - Privacy -
  - Autonomy limits -

February 17, 2026

- Welcome
- Review the starting point for this WG

    - Why are we here? Who are we? What problems are we solving? How will we solve those problems?
    - What do we want to get out of this WG? What do we expect to contribute here?
    - What are the expected outcomes? What will we deliver to the community?
- Discussion topics
    - Attestation of agent and model in use
        - Zero knowledge models
    - Agent Threat Model Framework
        - Attack taxonomy
        - Red-team scenarios
        - Testing harness definitions
    - Agent Identity and Delegated Authorization
        - We need a “capability” based auth model
        - Oauth doesn’t cut it
        - How does Agent authenticate to MCP server?
        - How does an Agent represent itself to infrastructure?
        - How does an Agent authenticate and authorize itself to another agent?
        - Restrict and audit what a user and agent can do - together, and separately
            - App can talk to database and now with agents/AI this app may talk to this database under some context
        - What tools can a user run/execute? How to ensure an agent can only invoke a tool it should?
    - Should we advise against multiple agents within a single application?
    - What about operational practices? Are there good sources for best operational practices for agentic AI security? Does that area need more work?
        - Wes: We have published this: https://engineering.block.xyz/blog/genai-security-principles
            - Wes: Happy to refine this with the group
    - Prompt Injection Mitigation
        - Needs standardization
        - Social engineering resistance
        - Testing corpus
    - Personally Identifiable Information (PII) obfuscation
    - Secure Memory specification
        - Data classification
        - Crypto standards
    - Auditability and Forensics
        - Decision traceability
        - Tools invocation and audit logs
        - Memory access logs - See Secure Memory Specification
        - Observability
        - “Conversation” traceability - i.e. associating tools + messages into a single context [conversation]

- Attack Surface Management/Continuous Threat Exposure Management
- Commerce and Payments
  - Open standards
  - Defi integration/Blockchain
- Tooling and plugins
  - Agent tool manifest
- Agent middleware as a solution for guardrails, context management, etc.
  - We created a library extension to the PydanticAI framework to enable operations such as: before_run, after_run, before_model_request, after_tool_call etc.
    - https://github.com/vstorm-co/pydantic-ai-middleware
  - Most “guardrails” solutions work by validating the agent’s input or output. But what about the steps in between - when the agent calls the tool?
    - Middleware could help create custom hooks which would allow for dynamic addition of context, e.g. before starting the tool “X” we fetch additional instructions/search in long-term memory
  - This functionality is inspired by the Langchain library.
-
- Eval framework for compliance/security of an agent
- Agent + Model Attack surface (i.e. model weaknesses that translate into agent vulnerabilities)
- **How to measure the security of an AI Agent**. Are there any observability tools (features) for the security layers
- **Agentic AI memory layer and security**. *What are the security standards for working with memory layers. (to expand the context, we’ve created a memory layer and want to tackle the security layer now to make it more enterprise-ready)*
- Certification criteria for different levels of AI systems
- Agentic security critic or verifiers as plugin to workflows

—

*Suggest distilling and consolidating notes into below for agreed upon items

Work Product

Security & Privacy Working Group | Agentic AI Foundation

Antitrust Compliance Notice

Meetings of the Agentic AI Foundation (AAIF) involve participation by industry competitors, and it is the intention of the Project to conduct all of its activities in accordance with applicable antitrust and competition laws. It is therefore extremely important that attendees adhere to meeting agendas, and be aware of and not participate in any activities that are prohibited under applicable U.S. state, federal or foreign antitrust and competition laws. Examples of types of actions that are prohibited at AAIF meetings and in connection with AAIF activities are described in the Linux Foundation Antitrust Policy. If you have questions about these matters, please contact your company counsel or Andrew Updegrove of the firm Gesmer Updegrove LLP, which provides legal counsel to the Linux Foundation.

Linux Foundation Antitrust Policy: https://www.linuxfoundation.org/antitrust-policy

Agentic AI Foundation

2

Zoom link:
https://zoom-lfx.platform.linuxfoundation.org/meeting/91539659750?password=59701aab-9082-4ab4-8115-a18c11039c4b

- **Please note* Working Groups are only open to AAIF members at this time. Participants must be invited to join.*
- *If someone from your organization would like to join, please have them sign up here.*

Cadence: Every other week; Tuesday 10:00am PT

Recordings available via: https://openprofile.dev/

- *Create a Linux Foundation (LFX) account to see all meetings, AI summaries, recordings, etc. all in one place.*

Google Drive: WG Security & Privacy

Private Discord Channel: https://discord.gg/Wbck3N2CsC
Please do not share publicly: this link is for Working Group members only

Private GitHub Repo: https://github.com/aaif/wg-security-and-privacy

Final Charter: AAIF WG Charter: Security & Privacy [FINAL]

Mailing list: wg-security-privacy@lists.aaif.io

Need help? Email: support@aaif.io

The Challenge

- **Current State:** Traditional security models assume human operators or static APIs. Autonomous agents introduce a novel, dynamic attack surface that current tools aren't built to defend.
- **The Gap:** We lack a unified approach to securing agentic environments. How do we secure agents and environments when they are acting autonomously across different domains?

Potential Mission *(suggestion; to be finalized by WG)*

- **Defining the Standard:** Our goal could be to establish the industry benchmark for secure agentic operations, moving from reactive measures to proactive, security-by-design principles.
- **Unified Framework:** An opportunity to answer the call: Could we develop a security and privacy framework that guides the entire ecosystem?

Some Key Scope Areas *(suggestion; to be finalized by WG)*

- **Standardizing Best Practices:** Can we build consensus on a common set of security protocols? Can we standardize security best practices for agent developers?
- **Adversarial Testing & Tooling:** Should this group take the lead on building security test harnesses and defining methodologies for the red-teaming of agents?
- **Strategic Ecosystem Alignment:** A key question for the group to consider: Are there organizations we should partner with to align our work with existing security standards bodies?

Working Group To-Dos:

- ☐ Decide on meeting cadence (date, time, frequency)
- ☐ Finalize WG mission and scope areas
- ☐ Decide what the final contribution of the WG will be (whitepaper, etc.)
- ☐ Develop a WG Charter and submit to Governing Board for review/approval
- ☐ Determine WG chair and co-chair(s) *(only open to Platinum & Gold Members)*
- ☐

Mission Statement

Proposal: The Security and Privacy Working Group advances the Agentic AI Foundation's mission by creating open standardization for secure execution and management of data, and privacy of information used by agentic systems.

Problems/Challenges:

Agent Identity, Authentication & Authorization

- How do we attest to the identity of an agent and the model it is using (including zero-knowledge approaches)?
- How do we handle delegated authorization from users to agents?
- Current auth models (e.g., OAuth) are insufficient for capability-based agent authorization
- No standard exists for agent ↔ MCP server authentication
- Agent ↔ infrastructure and agent ↔ agent authentication and authorization are undefined
- User/agent permission boundaries are unclear — how do we restrict and audit what each can do independently and together?
- No standard mechanism for tool-level access control to ensure agents only invoke authorized tools

Threat Modeling & Attack Surface

- No common agent threat model framework exists (attack taxonomy, red-team scenarios, testing harness)
- Model weaknesses propagate into agent-level vulnerabilities, and the combined attack surface is poorly understood
- Continuous threat exposure management for agents is immature
- Multi-agent architectures within a single application introduce complex, under-explored security challenges

Prompt Injection & Input/Output Security

- Prompt injection lacks standardized mitigation approaches (potential research areas: Dual-LLM, CaMeL, prompt sandwiching)
- Agents are susceptible to social engineering attacks
- No standardized testing corpus exists for prompt injection evaluation
- Inter-step validation (agent middleware as guardrails) is not well-defined beyond simple I/O filtering (ref: pydantic-ai-middleware)
- The role of agentic security critics / verifiers as workflow plugins is undefined

Data Protection & Memory Security / Privacy

- No secure memory specification exists (data classification, crypto standards)
- Agent memory layer security and standards are undefined
- PII obfuscation is needed throughout the agent execution lifecycle (e.g., summarizing "user is >21" rather than passing exact age)
- LLM provider inputs and outputs may leak sensitive data
- Environment variables, API keys, and encryption keys are at risk of exposure
- Secret and sensitive data management is unaddressed (model training data, company secrets)
- Different types of protected data in the context of AI agents have not been identified or categorized
- No defined data boundaries for agents
- Detection and protection against agent sabotage and rogue agents is undefined, including blast radius assessment

Auditability, Observability & Forensics

- Decision traceability for agent actions is lacking
- Tool invocation and audit logging have no standard
- Conversation traceability (associating tools + messages into a single context) is undefined
- Observability tooling for security layers is immature
- Memory access logging overlaps with data protection concerns and needs coordination

Evaluation, Compliance & Certification

- No evaluation framework exists for assessing the compliance/security of an agent
- No standard way to measure the security posture of an AI agent
- Certification criteria for different levels of AI systems are undefined

Operational Practices

- Best operational practices for agentic AI security are not codified (ref: Block GenAI Security Principles)

Commerce, Payments & Tooling

- Open standards for agent-driven commerce and payments (including DeFi/blockchain integration) do not exist
- No standard agent tool manifest for tooling and plugins
- No mechanism for enforcing security guardrails on commerce actions

Use Cases:

- Agent Identity, Authentication & Authorization — Establishing trust in who/what an agent is, what model it runs, and what it is permitted to do on behalf of users and other agents.
- Threat Modeling & Attack Surface — Understanding and managing the combined attack surface of agents and their underlying models, including adversarial testing and multi-agent risk.
- Prompt Injection & Input/Output Security — Preventing and detecting manipulation of agent behavior through malicious inputs, and validating outputs at each step of execution.
- Data Protection & Memory Security / Privacy — Safeguarding sensitive data across the agent lifecycle, including memory, context, secrets, and PII, with clear data boundaries and obfuscation strategies.
- Auditability, Observability & Forensics — Enabling traceability of agent decisions, tool use, and conversations for security monitoring, incident response, and compliance.
- Evaluation, Compliance & Certification — Defining frameworks and criteria for assessing and certifying the security posture of agentic AI systems.
- Operational Practices — Codifying best practices for the secure deployment and operation of agentic AI in production environments.
- Commerce, Payments & Tooling — Securing agent-driven commercial transactions, tool integrations, and plugin ecosystems through open standards.

Scope (in/out) topics:

In-Scope:

-

Out-of-Scope:

- OUT - Ethics/Morality/Responsible/Other - Delegate to the Governance and Policy WG
- OUT - Recovery from Model Poisoning - Accuracy and Reliability WG (PENDING WG Mission)

Potential Deliverables:

-

Other Initiatives:

- OWASP - Agentic Team
- AIUC-1
- Block GenAI Security Principles (Wes offered to refine with the group)

Taxonomy:

Agent Behavior

Agent derailment — An unintended deviation in an AI agent's behavior that causes it to pursue goals or take actions outside its intended scope, without malicious external cause. Note: Raised during discussion of non-malicious agent misbehavior (2026-03-03). Distinct from adversarial attacks.

Agent sabotage — Deliberate manipulation of an AI agent's behavior by an external party, causing it to act against its intended purpose. Note: Discussed alongside derailment as an edge case between security and reliability (2026-03-03).

Blast radius — The scope and extent of damage or data exposure that can result from a compromised, misconfigured, or rogue agent. Note: Samantha Coyle raised this in the context of protecting data from rogue agents (2026-03-03).

Typo Squatting - A supply chain attack where an adversary publishes a malicious server package with a name that rembles a legitimate one, hoping users attack

Human in the loop (HITL) — A design pattern in which a human must review, approve, or intervene in an AI agent's actions at defined checkpoints before the agent may proceed. Note: Identified as a key consideration during the initial brainstorm (2026-02-17).

Multi-agent persuasion — The risk that one AI agent manipulates or unduly influences another agent's decision-making through adversarial or misleading communication. Note: Brainstormed as a mitigation area (2026-02-17).

Rogue agent — An AI agent that operates outside its authorized boundaries, whether due to compromise, misconfiguration, or emergent behavior. Note: Used in discussion of blast radius and data protection (2026-03-03).

Data and Privacy

PII obfuscation — Techniques for masking, redacting, or transforming personally identifiable information so that it cannot be recovered or linked to an individual during agent processing. Note: Listed as a brainstorm topic (2026-02-17) and folded into Theme D (2026-03-03).

Privacy-preserving execution — Patterns in which an AI agent can operate on encrypted or protected data without exposing the underlying content, using techniques such as zero-knowledge proofs or trusted execution environments. Note: Ty described ZK-proof patterns; Tony Douglas cited TEEs and encrypted vector databases (2026-03-03).

Protected data — Information subject to regulatory or contractual protections (e.g., HIPAA, GDPR), which may include health records, financial data, or other categories that AI agents must handle with specific safeguards. Note: Tom Sheffler raised whether AI agents introduce new categories of protected data (2026-03-03).

Secrets — Traditional credentials and cryptographic material — such as API keys, tokens, passwords, and environment variables — that grant access to systems or services. Note: Distinguished from the broader "sensitive data" during terminology discussion. Ofek Dadush raised secrets management; David Deng proposed the narrower scope (2026-03-03).

Sensitive data — An umbrella term encompassing secrets, PII, proprietary information, and any data whose exposure would cause harm. Preferred over the narrower term "secrets" when the

intent is to cover all confidential data categories. Note: The group debated "secrets" vs. "sensitive data" and converged on "sensitive data" as the broader, less ambiguous term (2026-03-03).

Identity and Authorization

Agent identity — A verifiable identifier assigned to an AI agent that distinguishes it from other agents and human users within a system. Note: Samantha Coyle raised agent-to-agent and agent-to-MCP-server authentication (2026-02-17).

Delegated authorization — A mechanism by which a human or system grants an AI agent a scoped set of permissions to act on its behalf, typically with constraints on what actions the agent may perform. Note: Discussed alongside capability-based auth and tool invocation restrictions (2026-02-17).

Security Practices

Hooks and checkpoints — Standardized interception points in an agentic AI platform's execution pipeline where security controls, logging, or policy enforcement can be applied. Note: Bar Kaduri proposed hooks and checkpointing standards (2026-02-17).

Prompt injection — An attack in which an adversary crafts input that causes an AI agent to override its instructions, bypass safeguards, or execute unintended actions. Note: Identified as a mitigation area during ideation (2026-02-17).

Red teaming — Structured adversarial testing of AI systems in which testers simulate attacks to identify vulnerabilities, failure modes, and security gaps. Note: Listed as a brainstorm topic under pen testing and red teaming best practices (2026-02-17).

Infrastructure and Architecture

Agentic AI — AI systems composed of one or more autonomous agents that can perceive their environment, make decisions, invoke tools, and take actions with limited or no human intervention. Note: Foundational term for the working group; used throughout all meetings.

Trusted execution environment (TEE) — A secure, hardware-isolated area of a processor that ensures code and data loaded inside are protected from the rest of the system, including the

operating system. Note: Tony Douglas cited TEE work in the context of privacy-preserving agent workflows (2026-03-03).

Zero-knowledge proof (ZKP) — A cryptographic method by which one party can prove to another that a statement is true without revealing any information beyond the validity of the statement itself. Note: Ty described ZKP patterns where agents return only boolean results to prevent data leakage (2026-03-03).

Mission Statement

Option E - Trust + Builder hybrid. Merges the trust-centered goal with builder-practical language. Incorporates Cosmin's "enabling builders to deliver trusted systems" framing, Madhura's "design patterns" and "best practices" keywords, and Ty's "shared taxonomy."

> The Security & Privacy Working Group advances the Agentic AI Foundation's mission by developing open frameworks, a shared taxonomy, and practical guidance - including design patterns and best practices - that enable builders to deliver agentic AI systems people can trust with their data, their tools, and their decisions.

Option F - Trust + Builder + Shared Responsibility. Builds on Option E but explicitly calls out Govindaraj's shared responsibility model between providers and consumers.

> The Security & Privacy Working Group advances the Agentic AI Foundation's mission by developing open frameworks, a shared taxonomy, and practical guidance - including design patterns and best practices - that enable providers and consumers of agentic AI systems to share responsibility for protecting users, data, and infrastructure in ways the industry can trust.

Option G - Ecosystem + Shared Responsibility. Keeps the cross-cutting ecosystem lens from Option C but layers in Govindaraj's shared responsibility framing and Ty's taxonomy language.

> The Security & Privacy Working Group advances the Agentic AI Foundation's mission by establishing security and privacy as a shared discipline across the agentic AI ecosystem - producing open threat models, a shared taxonomy, and design patterns that providers, consumers, and peer working groups can adopt to share responsibility for trustworthy agentic AI.

Option H - Concise hybrid. Stays short and direct like Option D but weaves in the trust and builder themes from Cosmin's framing.

> The Security & Privacy Working Group advances the Agentic AI Foundation's mission by giving builders open frameworks, design patterns, and practical guidance for protecting data, managing trust boundaries, and resisting attacks - so the agentic AI systems they ship are ones people can trust.

Original Suggestion:
The Security and Privacy Working Group advances the Agentic AI Foundation's mission by creating open standardization for secure execution and management of data, and privacy of information used by agentic systems.

Option A - Trust-centered. Emphasizes the end goal of building confidence in agentic AI through open, practical work.

> The Security & Privacy Working Group advances the Agentic AI Foundation's mission by developing open frameworks, shared vocabulary, and practical guidance that help the industry build agentic AI systems people can trust with their data, their tools, and their decisions.

Option B - Builder-centered. Emphasizes pre-competitive outputs aimed at practitioners who are building these systems today.

> The Security & Privacy Working Group advances the Agentic AI Foundation's mission by producing pre-competitive security and privacy standards, design patterns, and best practices that give builders of agentic AI systems a shared foundation for protecting users, data, and infrastructure.

Option C - Ecosystem lens. Leans into Ty's proposal from 2026-03-03 that security and privacy is a cross-cutting discipline that touches every other working group.

> The Security & Privacy Working Group advances the Agentic AI Foundation's mission by establishing security and privacy as a shared discipline across the agentic AI ecosystem, producing open threat models, design patterns, and guidance that any builder or working group can adopt.

Option D - Concise and direct. Gets to the point in fewer words while still covering the core scope areas.

> The Security & Privacy Working Group advances the Agentic AI Foundation's mission by defining how agentic AI systems should protect data, manage trust boundaries, and resist attacks, through open standards and practical guidance the entire industry can use.

Scope

In Scope

A. Threat Modeling and Attack Analysis

- Attack surface mapping (agent + model + context chain)
- Attack classification and modeling
- Agentic attack vectors and mitigation strategies (e.g., prompt injection, social engineering, multi-agent persuasion, tool misuse, agent derailment, resource exhaustion)
- Supply chain security (system prompt integrity, model provenance, tools manifest verification)
- Agentic-specific indicators of compromise and threat signatures

B. Security Architecture and Design Patterns

- Hooks and checkpointing standards
- Security by design principles for agentic AI
- Human-in-the-loop patterns and guardrails
- Skills, tooling, middleware, and eval frameworks for agent compliance

C. Agent Control and Containment

- Bounding and constraining agent actions (scope limits, permission boundaries)
- Off-task detection, runtime intervention, and graceful degradation (kill switches, halting, rollback, permission downgrade, HITL fallback)
- Post-incident forensics and investigation of agent actions

D. Data Protection and Privacy

- PII obfuscation and sensitive data classification
- Secrets management (credentials and broader sensitive data)
- Privacy-preserving execution patterns (ZK proofs, TEEs, encrypted vector DBs)
- Protected data handling — technical controls, data minimization, and isolation
- Secure memory and agentic AI memory layer security
  Memory lifecycle management (derived data lineage, retention, deletion, and data sovereignty)
- Rogue agent data exposure and blast radius containment

E. Cross-WG Collaboration

- Identity and Trust WG — agent identity, delegated authorization, and authentication

- Observability and Tracing WG — decision traceability and audit logging
- Agentic Commerce WG — commerce and payments security guardrails (fund draining, account leaks, payment agent scope limits)
- Accuracy and Reliability WG — general agent reliability and accuracy
- Governance, Risk and Regulatory Alignment WG — regulatory compliance interpretation; this group monitors standards and understands regulatory requirements at a technical level to inform its own controls
- All WGs — reviewing deliverables through a security and privacy lens

---

Out of Scope

- Ethics, bias, and value-judgment questions — Deferred to the Governance and Policy WG.
- Interpreting or prescribing regulatory compliance — Determining what specific regulations (e.g., HIPAA, GDPR) require belongs to the Governance, Risk and Regulatory Alignment WG. This group may monitor standards and understand regulatory requirements at a technical level to inform its own technical controls (see Scope D), but will not duplicate regulatory or compliance work.
- Certification of agentic systems or tools — Certification criteria and programs are outside the scope of this group.

Deliverables

1. Taxonomy of Terms and Definitions (already in progress)

- Status: WIP, 18 terms defined
- Next step: Group review and expansion, with eventual proposal for AAIF-wide adoption
- Format: Living reference document

2. Agentic AI Threat Modeling: Gap Analysis and Framework Design

- An evaluation of existing AI threat model frameworks (e.g., CSA Maestro, OWASP, NIST AI RMF) to determine whether any adequately address agentic-AI-specific threats — such as multi-agent persuasion, tool invocation risks, delegated authorization boundaries, and supply chain integrity (system prompt integrity, model provenance, tools manifest verification)
- Format: Gap analysis and guidance document
- Rationale: Before the group can produce threat modeling guidance, it needs to establish whether a sufficient model already exists or whether the agentic-specific gaps are large enough to warrant new work

3. Security and Privacy Design Patterns Catalog

- Reusable patterns covering hooks and checkpoints, HITL, enforcing logical/semantical constraints, privacy-preserving execution, blast radius containment, secure memory and memory lifecycle management (retention, deletion, data sovereignty), agentic skills and tool invocation controls, agent containment and runtime intervention (kill switches, rollback, permission downgrade, off-task detection), and agent authorization models
- Format: Pattern catalog (problem, context, solution, trade-offs)
- Rationale: The group has identified many topics about how to build securely that fit a patterns approach; also absorbs the runtime monitoring and containment topics (Scope C) that had no deliverable home

4. Agentic AI Security Best Practices Guide

- Practical guidance for builders on secrets management, secure tool invocation, guardrails, eval frameworks, and post-incident forensics
- Includes a curated reference section pointing builders to recommended external frameworks and resources
- Format: Best practices and recommendations document
- Rationale: Consolidates practical security guidance into a single starting point for builders; agent identity and authorization are contributed to the Identity and Trust WG via the Cross-WG Checklist rather than specified here

5. Cross-WG Security and Privacy Review Checklist

- A lightweight checklist and process for reviewing other WGs' deliverables through a security and privacy lens

- Format: Checklist and process document
- Rationale: This WG serves as an intersecting layer between other WGs; initial targets are Identity and Trust (agent identity, authorization), Observability and Tracing (behavioral anomaly detection), and Agentic Commerce (payments security guardrails)

Security and Privacy Design Patterns Catalog

DRAFT - Security and Privacy Design Patterns Catalog

Version: Draft 0.1

Status: Working Draft

Working Group: Security & Privacy Working Group

---

1. Introduction

1.1 Purpose

The Security and Privacy Design Patterns Catalog defines reusable architectural patterns for addressing recurring security and privacy challenges in agentic AI systems.

The catalog provides implementation-independent guidance that can be applied across platforms, frameworks, deployment models, and execution environments. Each pattern captures a commonly encountered problem, the architectural mechanism used to address it, and the trade-offs associated with its adoption.

This catalog serves as the architectural foundation for the accompanying *Agentic AI Security Best Practices Guide*, which provides implementation and operational guidance for realizing these patterns in practice.

1.2 Scope

This catalog focuses on architectural patterns related to:

- Runtime security and governance
- Agent control and containment
- Secure tool invocation
- Agent memory protection and lifecycle management
- Privacy-preserving execution
- Secure multi-agent collaboration
- Security monitoring and intervention
- Auditability and recovery

This catalog assumes that identity establishment, authentication, trust establishment, and credential management are defined by the Identity & Trust Working Group and treats those capabilities as inputs to security policy and runtime enforcement.

1.3 Intended Audience

This document is intended for:

- AI platform architects
- Framework developers

- Infrastructure providers
- Security architects
- Runtime and orchestration platform developers
- Standards contributors

1.4 Relationship to Agentic AI Security Best Practices Guide

The **Security and Privacy Design Patterns catalog** and the **Agentic AI Security Best Practices Guide** are complementary deliverables that address different levels of abstraction within the security and privacy architecture for agentic AI systems.

The relationship between the two deliverables is intentionally hierarchical:

- The **Security and Privacy Design Patterns Catalog** answers: **What reusable architectural mechanisms address recurring security and privacy problems in agentic AI systems?**
- The **Agentic AI Security Best Practices Guide** answers: **How should those architectural mechanisms be implemented, configured, operated, and validated in practice?**

The Best Practices Guide does not define new architectural patterns. Rather, it builds upon the Pattern Catalog by providing practical recommendations for implementing and operating the patterns in real-world environments.

---

2. Design Principles

The patterns contained in this catalog are guided by the following principles.

2.1 Least Privilege - Grant only the minimum permissions and capabilities necessary for an entity to perform its intended function.

2.2 Defense in Depth - Protect systems through multiple independent controls such that the failure of a single control does not result in system compromise.

2.3 Explicit Authorization - Security-sensitive actions should be performed only after explicit authorization based on applicable policy and context.

2.4 Secure Failure - When security-relevant decisions cannot be completed or verified, systems should default to the most secure practical outcome.

2.5 Data Minimization - Collect, retain, process, and disclose only the information necessary for the intended purpose.

2.6 Impact Containment - Limit the scope and impact of failures, compromise, and unintended agent behavior through isolation and constrained execution.

2.7 Accountability - Security-relevant actions and decisions should be attributable, auditable, and explainable to support oversight, investigation, and continuous improvement.

2.8 Recoverability - Systems should support restoration to a trusted state following failures or security incidents while preserving system integrity and audit evidence.

---

3. Pattern Domains

Patterns are organized into the following domains.

Patterns in this catalog are organized according to the primary security or privacy concern they address. While individual patterns may support multiple objectives, each pattern is classified according to its primary architectural concern.

3.1 Execution Control and Enforcement

Patterns that regulate and constrain agent execution through security controls, policy enforcement, approval mechanisms, and execution constraints.

3.2 Authorization and Capability Control

Patterns that govern what actions agents are permitted to perform and how permissions are enforced when invoking tools, skills, services, or other protected resources.

3.3 Containment and Runtime Intervention

Patterns that limit the impact of failures, compromise, or unintended agent behavior and support controlled intervention during execution.

3.4 Memory Protection and Lifecycle Management

Patterns that protect agent memory and govern the secure management of information throughout its lifecycle, including storage, retention, deletion, integrity, and jurisdictional considerations.

3.5 Privacy Protection

Patterns that protect sensitive information by minimizing unnecessary collection, processing, retention, disclosure, and propagation throughout agent execution and collaboration.

3.6 Agent Collaboration Security

Patterns that secure interactions between cooperating agents, including delegation, information exchange, coordination, and boundary enforcement.

3.7 Monitoring, Audit, and Recovery

Patterns that support detection of security-relevant events, auditability of agent behavior and decisions, and recovery from security incidents or unintended agent actions.

---

4. Pattern Template

Each pattern in this catalog follows a common structure.

Pattern Metadata

Attribute	Value
Pattern Identifier	SP-XXX
Pattern Category	Runtime Governance
Status	Draft
Security Objectives	Confidentiality, Integrity, Availability
Privacy Objectives	Data Minimization
Applicable Lifecycle	Design, Runtime
Related Patterns	SP-005

Pattern Name

Intent

Problem

Context

Solution

Consequences

Benefits

Drawbacks

Residual Risks

Related Patterns

Implementation Considerations

Implementation guidance for this pattern is intentionally omitted from this catalog.

Reference to the *Agentic AI Security Best Practices Guide* for details

---

5. Inter Pattern Relationships

---

6. References

References to relevant specifications, standards, research publications, and external guidance that informed the catalog.

Examples:

- NIST AI Risk Management Framework
- OWASP Agentic AI Project
- MITRE ATLAS
- Zero Trust Architecture
- Privacy by Design
- Relevant AAIF Working Group deliverables

Appendix A – Pattern Status

Patterns progress through the following maturity levels.

Status	Description
Candidate	Proposed pattern under discussion
Draft	Pattern under active development
Stable	Pattern approved by the Working Group
Deprecated	Pattern superseded by a newer pattern
