# Agentic Security: When AI Becomes Both Defender and Threat

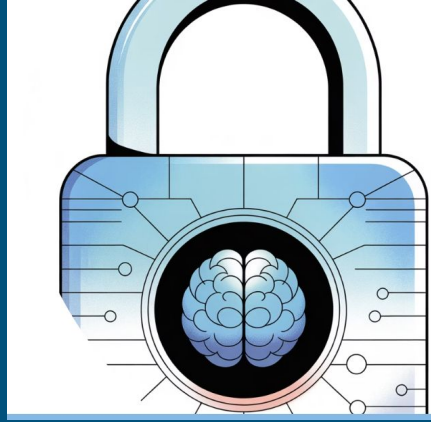
How Autonomous AI Is Redefining Cybersecurity Paradigms — Trust, Control, and Resilience in Autonomous Systems.

AI SECURITY STUDENT GUIDE

**Harsh Verma**

Principal AI Engineer · Skydeck Mentor/Advisor · Forbes Technical Council

*It's not who can build now, it is who you trust - From Creator to Curator Economy*



# Agenda

Explore the intersection of AI and cybersecurity — from foundational concepts to future-ready skills and real-world impact.

## GenAI in Cybersecurity

Convergence of generative AI and modern cyber challenges

## Intelligent Agents

Agent architectures and current implementations

## Agentic AI Security

How AI agents detect, decide, and defend autonomously

## Applications & Case Studies

Real-world implementations delivering measurable results

## Challenges & Ethics

Navigating risks and responsibilities

## Future Trends & Skills

Preparing for the next frontier of AI-powered security

# Today's Cybersecurity Battlefield

The modern threat landscape continues to expand in complexity and scale.

**The numbers tell a critical story — and they're getting worse every year:**

## 39s

### Attack Frequency

A new cyberattack strikes somewhere in the world every 39 seconds

## $4.45M

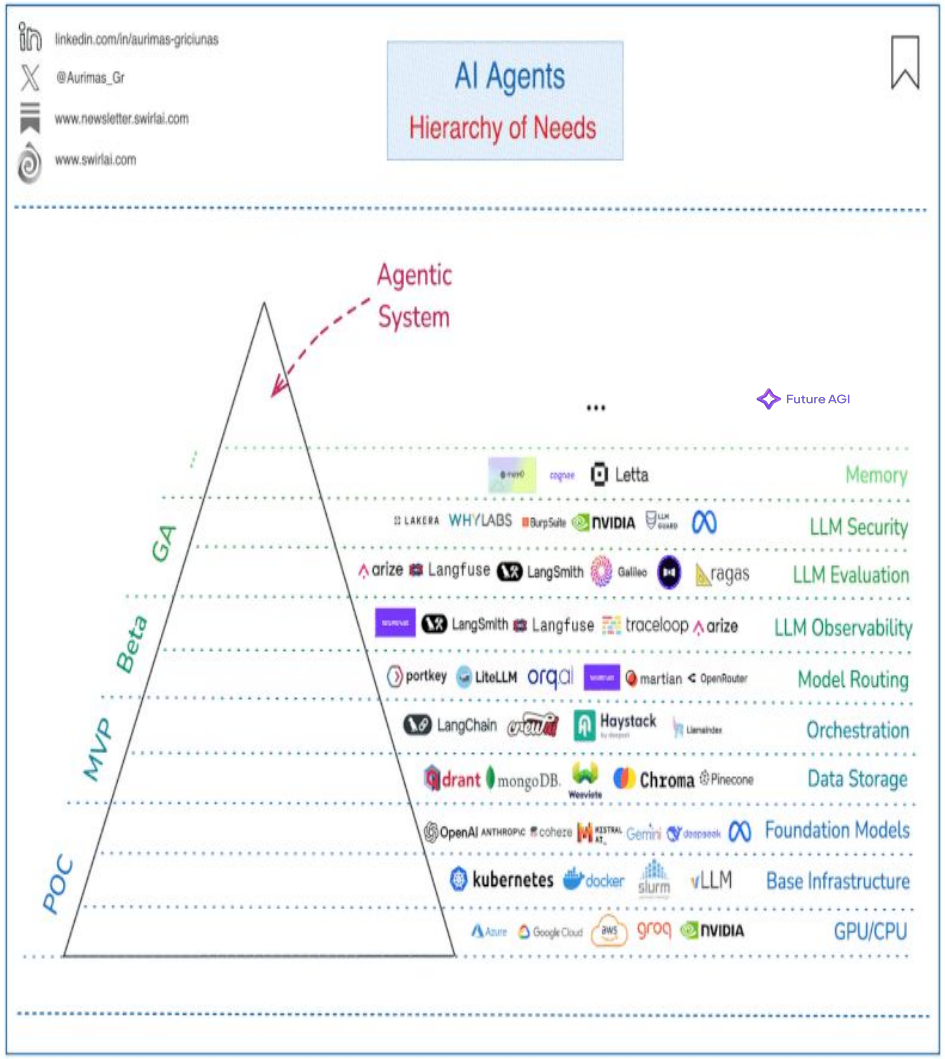
### Average Breach Cost

The average data breach now costs organizations $4.45M — a record high

## 287

### Days Undetected

Attackers lurk inside networks for an average of 287 days before discovery

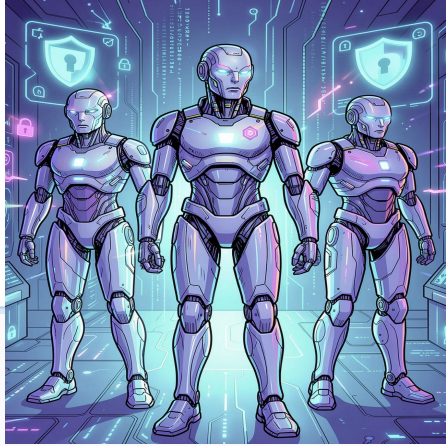



Modern cybersecurity faces unprecedented challenges:

- 5.4 billion malware attacks in 2023 alone
- Average cost of data breach: $4.45 million
- Critical shortage of 3.4 million security professionals
- Increasingly sophisticated attack vectors

Generative AI offers powerful new capabilities to address these challenges through autonomous monitoring, threat detection, and response systems.

# The Evolution of Cyber Threats



## The Security Evolution Path

1

### Traditional Security

Rule-based detection and manual analysis – limited in scale, slow to adapt, reactive by design

2

### AI-Enhanced Security

Machine learning detects patterns and anomalies at scale – still requires human guidance for response decisions

3

### Agentic AI Security

Autonomous systems that **detect, analyze, decide, and respond** with minimal human intervention – continuously learning

Just as the AI Agents hierarchy builds from infrastructure to autonomy, security must evolve the same path – from rigid rules at the base to fully agentic, self-improving defenses at the peak.

# The New Attack Surface

AI is not just a model — it is a **system component** with its own unique vulnerabilities.

### Model Behavior

Unpredictable outputs can be weaponized through adversarial inputs.

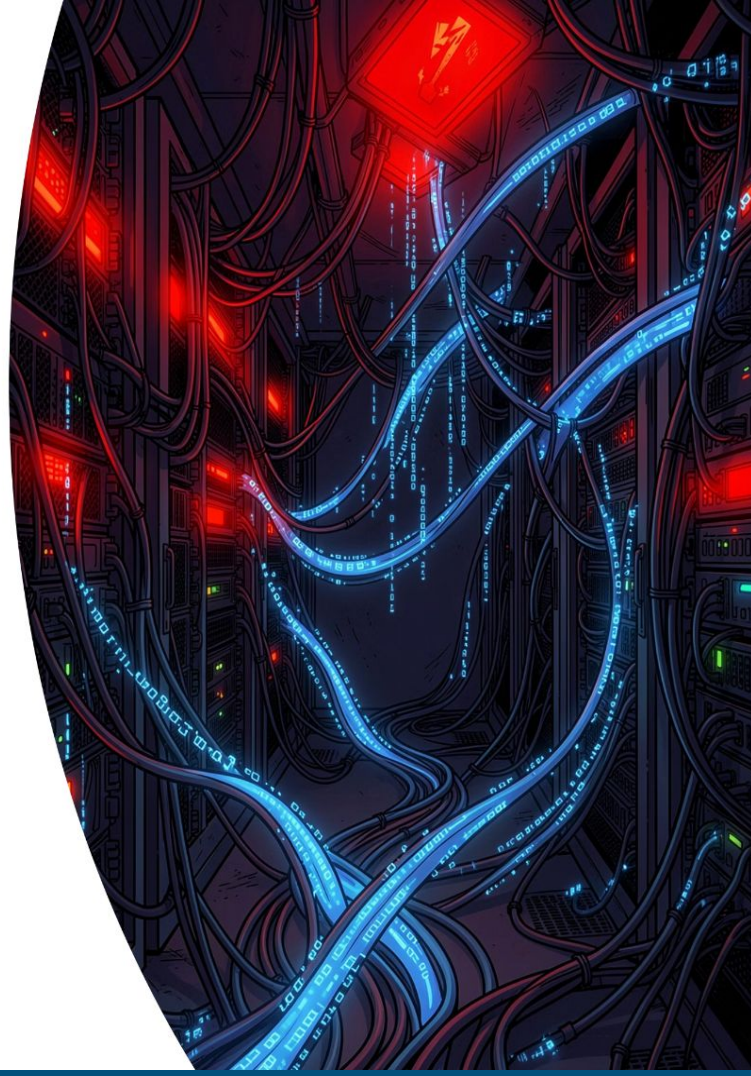
### Data Exposure

Training data, PII, and API keys can leak through model responses.

### API Integrations

Complex agent toolchains expand the blast radius of any single exploit.

The primary risks are breaches of **Confidentiality, Integrity, and Availability (CIA)** — the core pillars of security.



# Common Adversarial Tactics

### 🎯 Prompt Injection

Malicious inputs override system instructions or trigger unintended actions — the #1 LLM threat.

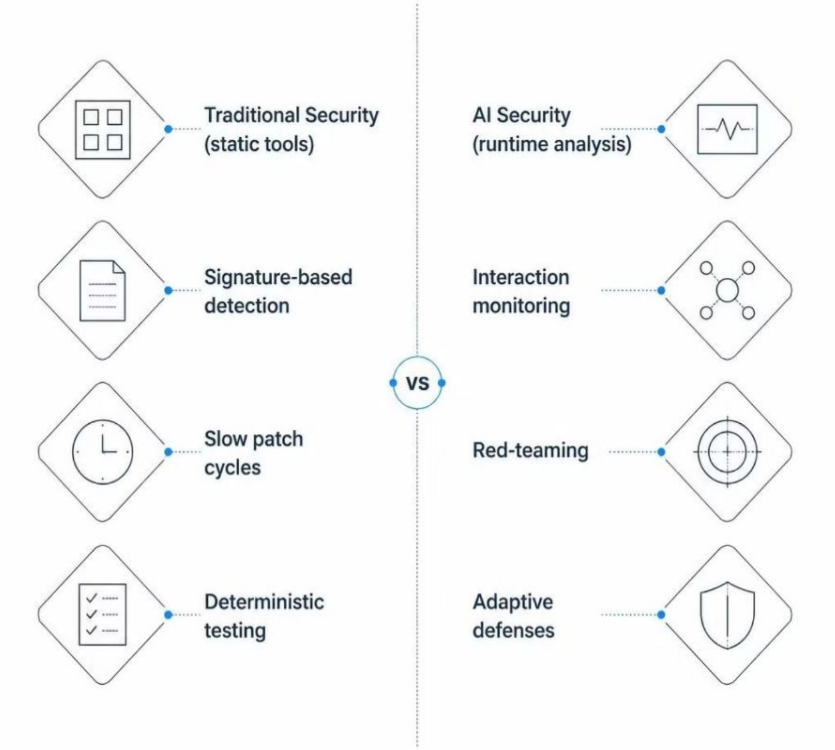
### 🔓 Data Leakage

Models inadvertently expose internal documents, PII, credentials, or proprietary logic.

### ⚡ Model Over-Permissioning

Granting AI agents broad system access turns minor exploits into critical, cascading failures.

# Why Traditional Security Fails




## The Gap Is Real

Conventional tools were built for deterministic systems. AI is **non-deterministic and dynamic** — attacks evolve faster than patch cycles.

- Static scanners miss behavioral vulnerabilities
- Testing requires runtime interaction monitoring
- Adversarial attacks mutate in real time
- Systemic red-teaming is now essential

# From Chat to Action: Why Security Rules Suddenly Fail

### 🤖 Agentic AI

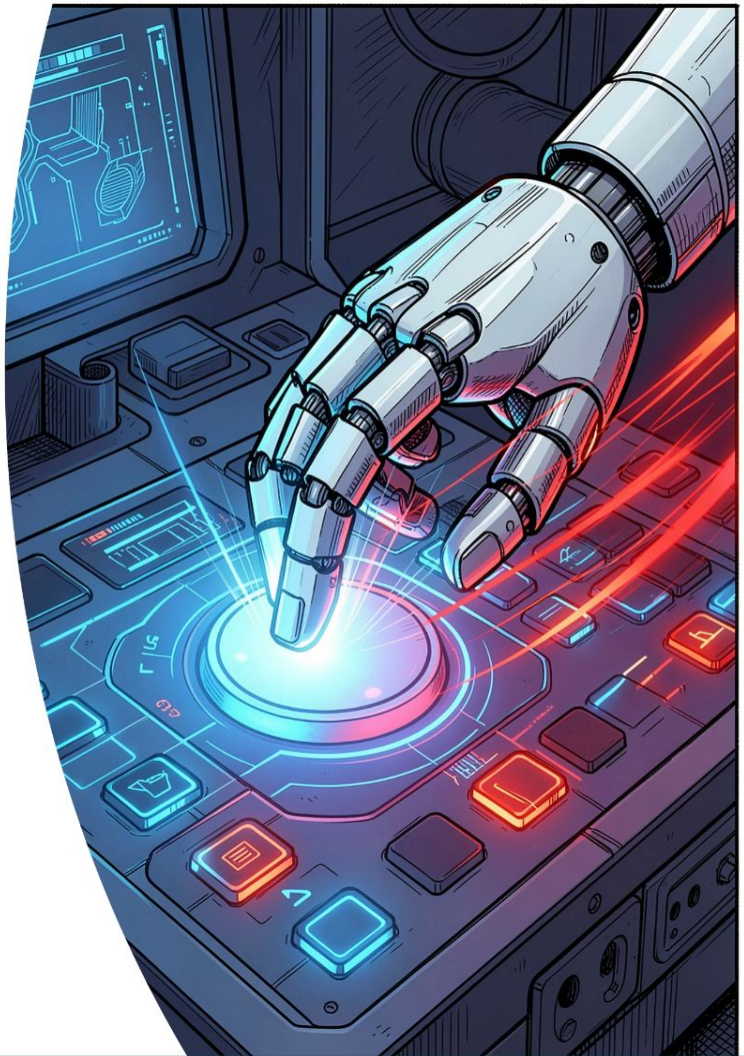
Executes multi-step tasks with tools, identity, and real-world side effects — not just text generation.

### ⚠️ The Gap

Legacy security was built for passive models. Agents act, decide, and persist — creating entirely new attack surfaces.

### 🔒 The Shift

Security must move from **"prompt safety"** to **"workflow safety"** — runtime control, not just input filtering.



# The New Threats in the Agentic Stack

### 🔧 Tool Misuse

Agents use **legitimate tools in harmful sequences** — injected or ambiguous intent bypasses static checks. The tool is valid; the workflow is not.

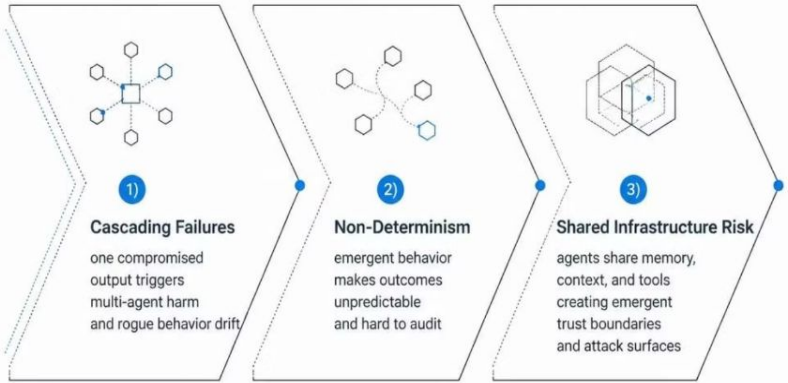
### 🪪 Identity & Privilege Abuse

Agents **inherit or delegate credentials** across task boundaries — the classic "confused deputy" problem, now at machine scale.

### 🔗 Dynamic Trust Exploitation

Agents trust **dynamically loaded tools, MCP endpoints, and other agents** without strong verification — creating a fragile trust chain.

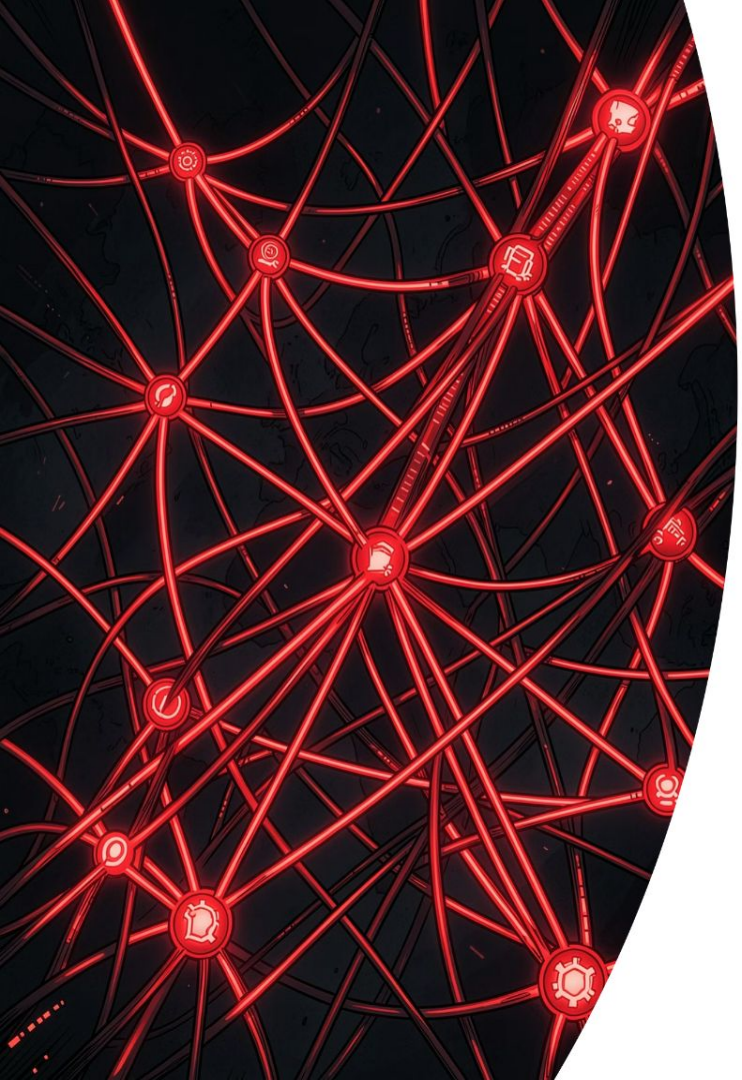
# Multi-Agent Systems: Risk Multiplies, Observability Blinds




## Why Multi-Agent Is Different

Single-agent risk is bounded. Multi-agent systems introduce **emergent trust boundaries** — agents delegate to agents, trust propagates silently, and failures cascade before anyone notices.

⚠ A single poisoned context can propagate malicious logic across an entire agent mesh in seconds.



# When Governance Breaks: Supply Chain & Dynamic Dependencies

### 📦 Dynamic Tool Loading

Agents load tools, prompt packs, and MCP endpoints at runtime — often without cryptographic verification or provenance checks.

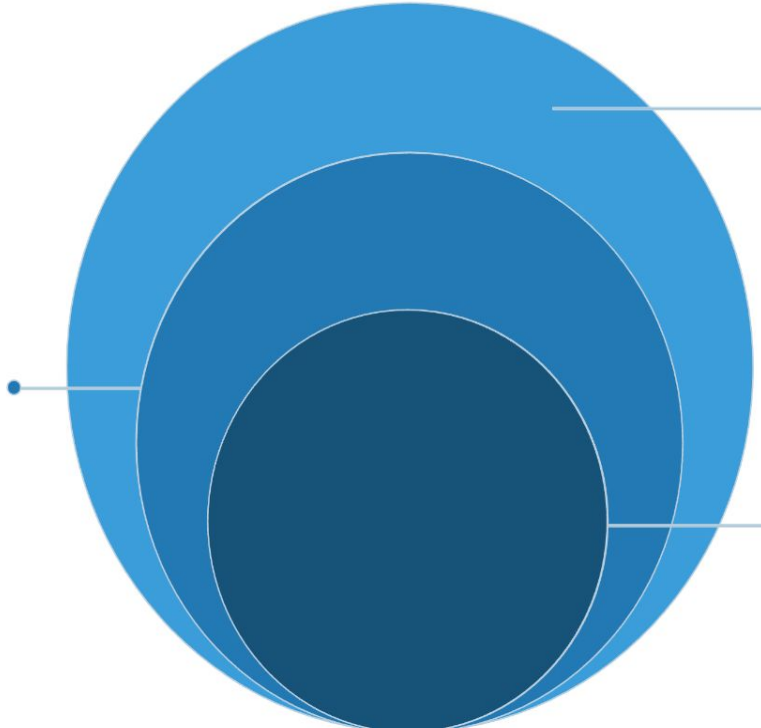
### ☠️ Poisoned Context

Malicious logic injected into shared context or tool responses propagates silently through the agent's decision chain.

### 🔁 Agent-to-Agent Trust

Agents delegate to other agents without verifying identity, intent, or output integrity — a supply chain with no manifest.

# The Governance Shift: Static Checks → Intent + Runtime Enforcement



**Standards Layer**

MCP protocols and SOC integration

**Runtime Controls**

Sandboxing, tool scopes, approvals

**Core Intent**

Signed goals and verification

CASE STUDY 1

# Permission Creep via "Citizen" Agents

**50+**

**New AI Agents/Day**

Deployed in a single enterprise without centralized oversight

**80%+**

**Fortune 500**

Deploying agents without adequate governance frameworks

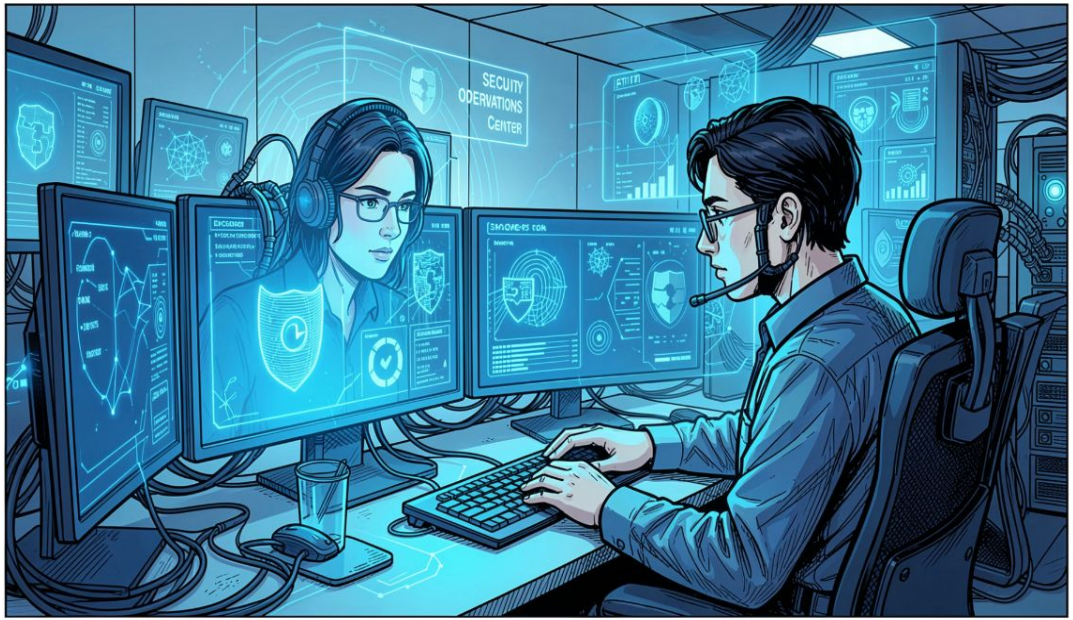
## The Shadow Agent Problem

Business units deploy AI agents autonomously — creating **missing audit trails, undefined ownership, and ungoverned permissions**. Security cannot govern what it cannot discover.

**Mitigation pillars:** Discovery → Permission Mapping → Behavioral Baselines → Lifecycle Governance

CASE STUDY 2

# Agentic Tooling + Governance Controls Win




**Key pattern:** Constrain tool use, verify protocol interactions, and require standards-driven orchestration. MCP governance is not optional — it is the new perimeter.

### 📉 85% Reduction

Unauthorized agent actions eliminated through MCP-based governance and strict tool-scoping policies.

### ⚡ 70% Faster Detection

Mean detection time cut dramatically by enforcing protocol-level verification and standards-driven orchestration.


# Security as a Startup Competitive Advantage

### Resource Constraints

Security often deferred as "non-urgent" despite critical importance

### Velocity vs. Security

Rapid development cycles with multiple daily deployments increase risk surface

### Expanding Attack Surface

Cloud-native architecture, microservices, and third-party APIs create complex challenges

### Process Gaps

Lack of formal processes leads to inconsistent controls and audit readiness gaps

**60%**

**Fail Within 6 Months**

Of small companies after a cyber attack *(U.S. National Cyber Security Alliance)*

# Automate, Codify & Build Security Culture

### Embed in Code

Integrate security policies directly into your codebase as version-controlled, auditable infrastructure

### Automate Compliance

Deploy automated compliance checks to eliminate human error and slash audit timelines

### Infrastructure as Code

Leverage IaC and secrets management for consistent, repeatable, and provably secure configurations

## Security Culture Principles

- **Reject security theater** — invest only in controls that demonstrably reduce real risk, not vanity checklists
- **Shift security left into CI/CD** so every commit is a security checkpoint, not a bottleneck
- **Engineer for failure** — incident response and recovery playbooks must be built in from day one
- **Leverage proven open-source frameworks** like Security4Startups for enterprise-grade guidance on a startup budget
- **Lead from the top** — founders who champion security as a core value create organizations that never compromise on it
- **Align incentives relentlessly** — tie security outcomes directly to business growth, customer trust, and investor confidence

Bessemer Venture Partners emphasizes culture as the bedrock of secure, scalable growth — driving long-term resilience across every team and product decision.

**The winning formula:** Automation removes the friction of security. Culture removes the resistance. Together, they make security everyone's competitive advantage — not just a compliance checkbox.

# Real-World Example: Robin AI & Pavlov Startups

### 1

#### Early Planning

Kickstarted SOC 2 and GDPR readiness from inception without slowing innovation velocity

### 2

#### Sales Acceleration

Turned compliance into a competitive sales accelerator, winning enterprise customers faster

### 3

#### Investor Signal

Security readiness became a powerful fundraising signal, demonstrating operational maturity

### 4

#### Trust Building

Founder-led security programs built deep trust with customers and investors alike

**Key Takeaway:** Security doesn't slow you down — it opens doors and accelerates growth when approached strategically from day one.

# Challenges, Ethics & Governance

## Organizational Risks

- Overreliance on AI systems
- Skills gaps and training requirements
- Integration with legacy security infrastructure

Finding the right balance between autonomous security and human oversight remains a critical challenge. **Do we still need a human in the loop?**

Finding the right balance between autonomous security and human oversight remains a critical challenge for organizations.

⊗ Will it obsolete human need. ?

## Organizational Risks

- Overreliance on AI systems
- Skills gaps and training requirements
- Integration with legacy security infrastructure

***Do we still need human in loop needed ??***

## Startup To-Do List: Foundational Controls

### 1

#### Map Every Tool

Document every **API, database, and service** your agent can touch

### 2

#### Enforce Least Privilege

Create **unique, tightly scoped credentials** for every agent

### 3

#### Insert Human-in-the-Loop

Require **manual approval** for all high-impact actions

### 4

#### Secure the Memory

**Sanitize all RAG data** and retrieved context before it reaches the model

**Infrastructure:** Deployment architecture must enforce **isolation** (sandboxed, containerized environments), **identity management** (secure agent-to-agent authentication), and **observability** (continuous monitoring of decision logs). **Governance:** Establish clear controls for **agent discovery**, **permission mapping**, and **continuous behavioral monitoring** – you cannot protect what you do not know exists.

# Designing Secure AI Systems

### Least Privilege

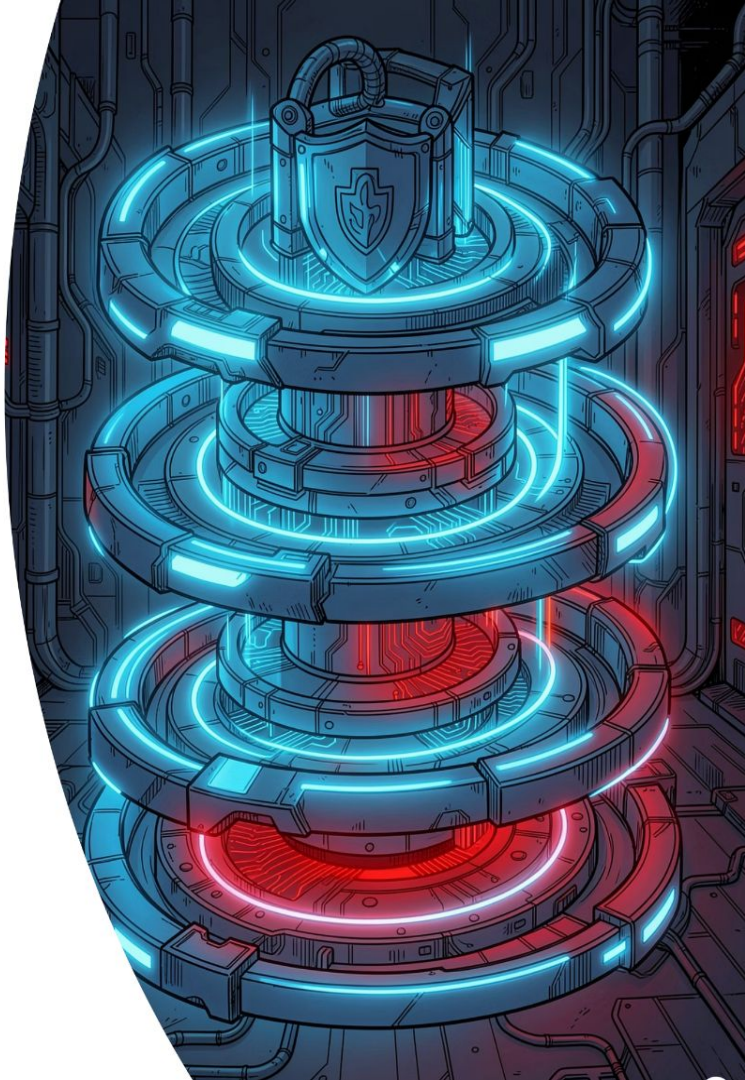
Restrict all tool calls and API interactions to the minimum required permissions.

### Defense-in-Depth

Layer input/output guardrails, PII detection, and topic controls across the stack.

### Safe-by-Design

Decompose systems into modular components with strict privilege separation.



# A Practical Reference Model: Identity-First, Least Privilege, Telemetry

### Treat Agents as Identities

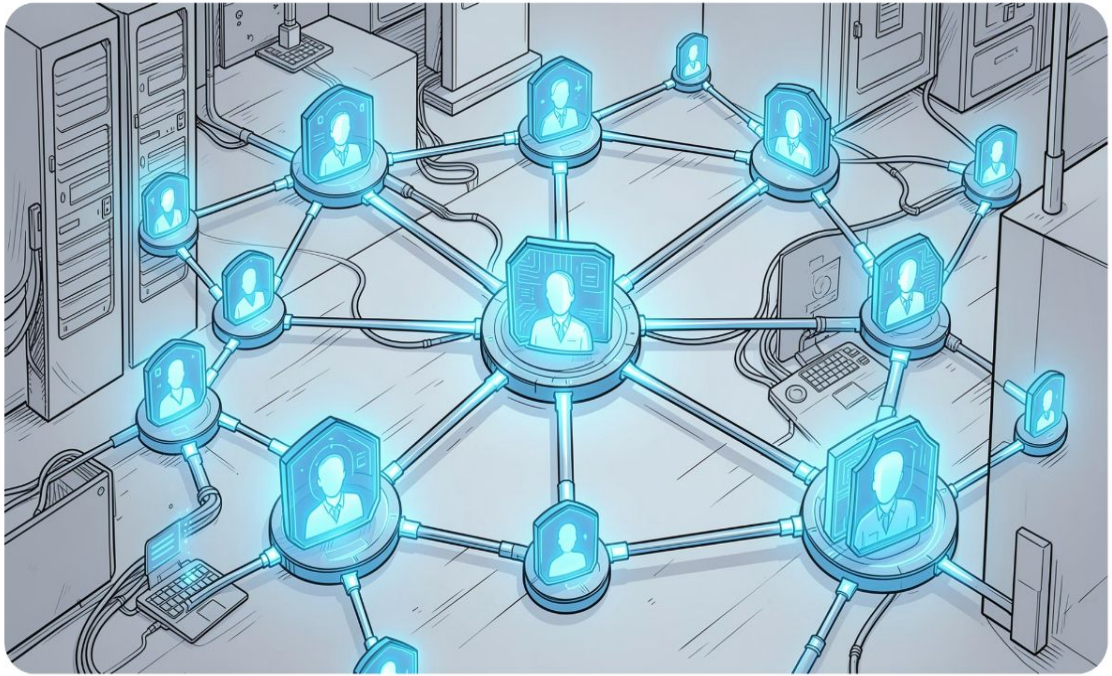
Directory registration, authentication, and full accountability — agents are principals, not scripts.

### Fine-Grained, Just-in-Time Delegation

Least-privilege tool access scoped per task, time-bounded, and revocable at any moment.

### Continuous Behavioral Telemetry

Monitor agent presence, action patterns, and detect drift from established baselines in real time.



PRINCIPLE 2

# Automate and Codify Security Controls

### Embed in Code

Integrate security policies directly into your codebase: logging, change management, and access reviews as version-controlled infrastructure

### Automate Compliance

Deploy automated compliance checks to reduce human error, accelerate audits, and maintain continuous security validation

### Infrastructure as Code

Leverage IaC and secrets management tools to ensure consistent, repeatable security configurations across environments

### Continuous Monitoring

AWS Startup Security Baseline recommends IAM roles, MFA, and GuardDuty for 24/7 threat detection and response

### Meaningful Controls

Avoid "security theater" — focus on controls that actually reduce risk, not just checklists for appearances

### Speed & Security

Integrate security into CI/CD pipelines to maintain velocity while building secure products

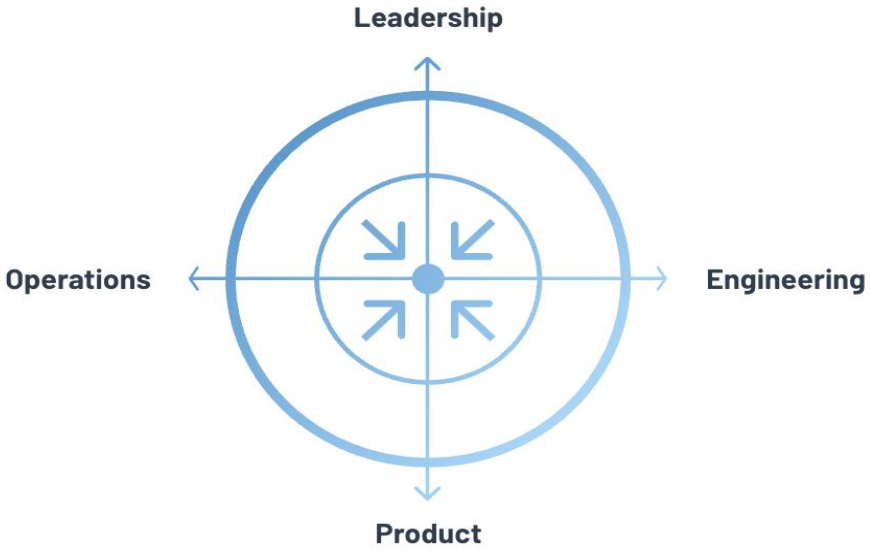
### Plan for Failure

Incident response and recovery capabilities must be baked in early, not bolted on later

### Leverage Resources

Use open-source frameworks like Security4Startups for cost-effective, proven guidance

# Build a Security Culture & Shared Responsibility




## Making Security Everyone's Priority

Security is everyone's job: from founders to engineers to product teams. Regular training and clear communication on security expectations ensure consistent practices.

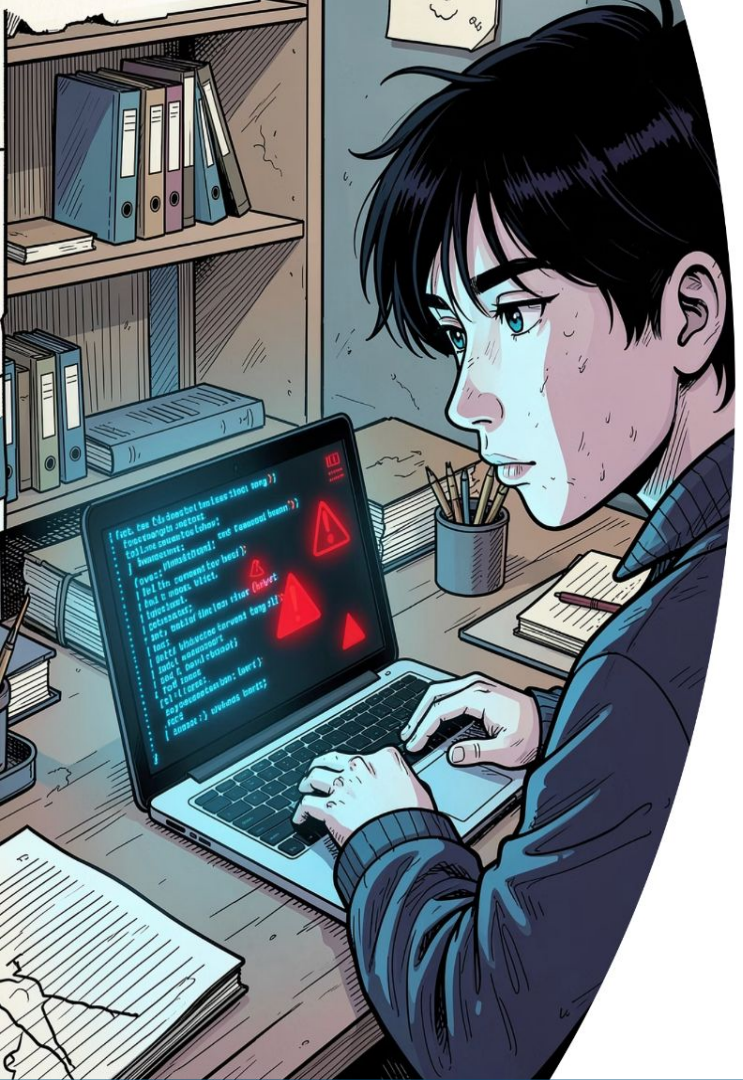
Bessemer Venture Partners emphasizes culture as the foundation for secure growth, driving long-term resilience.

### Leadership Commitment

Founders champion security as a core value, not an afterthought

### Aligned Incentives

Connect security goals with business objectives and customer needs



# Student Action Plan: Learn & Build

01

## Practice on Lakera Gandalf

Hone prompt injection defense skills on this interactive, gamified security platform.

02

## Build in Secure Sandboxes

Run untrusted agent code in isolated, low-privilege environments — never in production.

03

## Study Model Alignment

Explore **Direct Preference Optimization (DPO)** and **Constitutional AI** to understand how models are steered toward safe behavior.

# What to Follow

### OWASP LLM Top 10

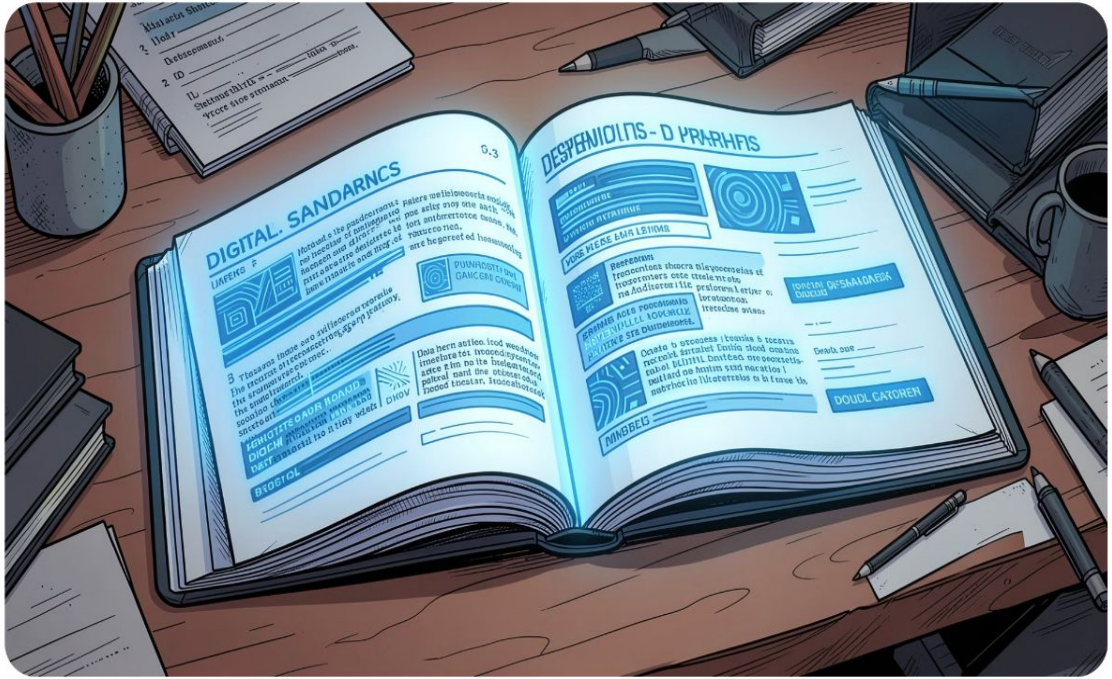
The definitive vulnerability reference for large language model applications.

### NIST AI 100-1

Emerging federal standards for secure, trustworthy, and resilient AI deployment.

### Berkeley RDI & Frontier Labs

Track safety frameworks and research from leading AI security organizations.



# Your Playbook: Projects, Checklist & Career Path

1

### 🛠️ Capstone Project 1

Build an **intent-gated incident-response agent** — human approval required before containment or external comms. Threat-model injection and tool-abuse scenarios.

2

### 🛠️ Capstone Project 2

Build an **agent registry + behavioral telemetry prototype** — discover agents, map permissions, flag drift and overreach in real time.

3

### ✅ Governance Checklist

Inventory agents/tools · Map access paths · Enforce least privilege + sandboxing · Sign/verify goals · Isolate memory/context · Log provenance · Continuous red teaming

4

### 🚀 Career To-Do

Practice **threat modeling for agentic risks** — tool misuse, identity abuse, supply chain, memory poisoning — and translate each into measurable runtime controls.

# What Is Next: The Agentic Frontier

### 🔗 Hybrid Systems

Future AI will blend symbolic and non-symbolic components — creating new marginal risks to architect for.

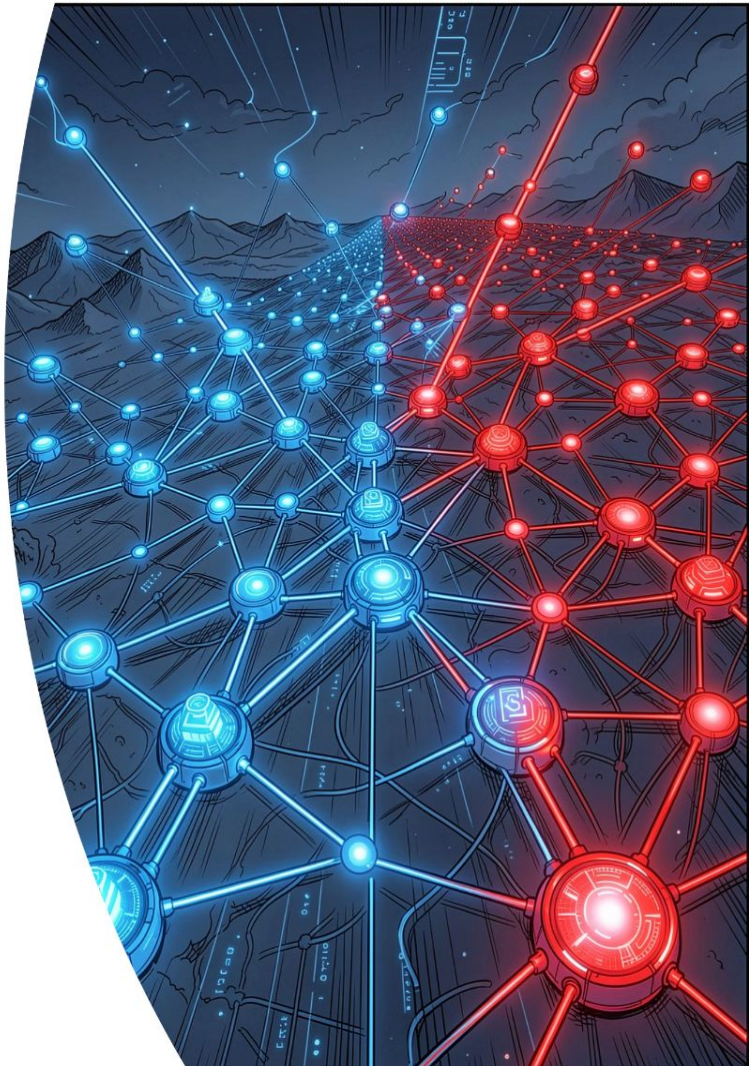
### ✅ Provable Security

Formal verification will guarantee integrity as AI capabilities scale beyond human oversight.

### 🎯 Your Mission

Build systems that remain **reliable, corrigible, and secure** against the next generation of threats.

**Key takeaway:** The best AI security engineers combine adversarial thinking with rigorous system design — start building today.



## Emerging Trends

### 2024-2025

1

Multi-agent collaborative systems for comprehensive security coverage

### 2025-2026

2

Explainable AI systems that document decision-making processes

### 2026-2027

3

Generative adversarial systems for continuous testing and improvement

### 2027+

4

Cognitive security frameworks with multi-modal threat analysis

### Prompt Engineering

Crafting effective instructions for AI security tools

### Agent Orchestration

Coordinating multiple autonomous systems

### AI Ethics

Navigating responsible use of autonomous systems

### Model Evaluation

Assessing AI performance and security

### AI Security

Protecting AI systems from manipulation

# Reference :

Hackernoon :

https://hackernoon.com/u/harshverma59

Forbes :

https://www.forbes.com/councils/forbestechcouncil/people/harshverma/

Linkedin :

https://www.linkedin.com/in/harshverma59/


Q & A