Safe to Experiment

How Secure AI Sandboxes Give Teams Space to Innovate, Fail, and Learn

*This document is written for a general professional audience, with particular attention to legal and compliance practitioners. Technical terms are defined in the Glossary of Terms beginning on page 5. Terms that appear in the glossary are shown in **bold** on first use.*

Every organization wants to harness the power of artificial intelligence, but the path from curiosity to capability is lined with risk. Teams eager to experiment with **large language models (LLMs)**, **AI agents**, and automation workflows often resort to unsanctioned tools, inadvertently exposing sensitive data. Secure **AI sandboxes** offer a way forward: governed, isolated environments where teams can explore freely, learn from failure, and build confidence in new technology without putting **production data**, compliance standing, or customer trust at risk.

The Innovation vs. Security Tension

The pressure to adopt AI is intense. Across industries, leadership teams are asking how AI can reduce costs, accelerate decision-making, and unlock new revenue streams. But when enthusiasm outpaces governance, the results can be damaging.

When employees experiment with AI tools outside sanctioned channels, a growing phenomenon known as **shadow AI**,[1] they can inadvertently feed proprietary data, source code, and client information into models that may retain or expose that data. The IBM and Ponemon Institute *2025 Cost of a Data Breach Report* found that one in five organizations has experienced a data breach tied to unsanctioned AI use, with associated costs averaging roughly $670,000 above standard breach remediation.[2] Beyond the direct financial impact, these incidents erode customer trust, trigger regulatory scrutiny, and create a chilling effect that can slow legitimate AI adoption across the entire organization.

The dilemma is clear: teams need room to experiment, but uncontrolled experimentation creates unacceptable exposure. Banning AI outright is not a viable strategy either, as it simply drives usage underground. What organizations need is a structured environment where experimentation is both encouraged and contained.

What Are Secure AI Sandboxes?

A secure AI sandbox is an isolated computing environment purpose-built for experimentation. Think of it as a controlled testing room with its own walls, locks, and cameras. It creates a strong **trust boundary** between experimental AI workloads and the organization's live ("production") systems, ensuring that models, prompts, and agent workflows can be tested freely without touching real data or impacting business operations.

Well-designed sandboxes share several core characteristics. They use **synthetic data** or **anonymized data** so teams never work with real customer information. They enforce **role-based**

access controls (RBAC) tied to the organization's **identity provider (IdP)**. They maintain immutable **audit trails** that capture every file read, **API** call, and agent action with timestamps and user identity. And they apply resource limits on computing power, network access, and data storage to prevent both misuse and runaway costs.

This approach aligns with the **NIST AI Risk Management Framework (AI RMF 1.0)**, a voluntary federal standard released in January 2023.[3] The framework provides four core functions that map naturally onto a sandbox lifecycle: **Govern** (establish policies for what may be tested), **Map** (identify risks before deployment), **Measure** (test for validity and safety inside the sandbox), and **Manage** (decide what is ready to move into production). By embedding this cycle into an isolated environment, organizations create a repeatable, auditable path from initial idea to production-ready AI.

Building Your Sandbox: A Practical Roadmap

Standing up a secure AI sandbox does not require building from scratch. A range of managed platforms and **open-source** tools make it possible to move from concept to a working environment in days, not months. The following five steps outline a practical approach that balances speed with rigor.

1. Choose your isolation layer. The foundation of any sandbox is the technology that separates experimental workloads from live systems. Modern approaches fall into two categories. The first is **microVMs** (micro virtual machines), such as AWS Firecracker or Kata Containers, which create lightweight, self-contained virtual computers that provide hardware-level separation. The second is user-space application kernels like **gVisor**, which intercept an application's requests before they reach the host computer's core operating system, adding a protective buffer. For teams that prefer a managed, turnkey experience, platforms like E2B, Northflank, and Daytona offer ready-made sandbox infrastructure with **bring-your-own-cloud (BYOC)** options, meaning the sandbox runs on your organization's own cloud account rather than a third party's servers. Enterprise cloud users can also leverage built-in capabilities from AWS SageMaker or Azure AI Studio. For maximum control, open-source projects like Agent Sandbox deploy on **Kubernetes** for declarative, policy-driven isolation.

2. Provision safe data. A sandbox is only as safe as the data inside it. **Production data** should never enter the environment. Instead, use platforms like MOSTLY AI, Neosync, or Tonic.ai, which generate **synthetic data** (artificially created datasets that mimic the structure of real data) and perform **anonymization** (stripping out **personally identifiable information**). Before any data enters the sandbox, automated scanning tools should verify that no real PII slipped through. This step is critical: even well-intentioned teams can accidentally include sensitive information in test datasets, creating the very exposure the sandbox was designed to prevent.

3. Lock down access and networking. Integrate the sandbox with your organization's existing **identity provider** via **single sign-on (SSO)**, so that user authentication is handled

centrally and consistently. Apply least-privilege **role-based access controls (RBAC)** with short-lived credentials that expire after each session, ensuring no one retains standing access beyond what they need. On the network side, isolate the sandbox within a dedicated **virtual private cloud (VPC)** using private subnets, and disable outbound internet access unless explicitly required for a specific experiment. An important caveat: isolation that relies only on **DNS** filtering is insufficient, as it can be bypassed through a technique called DNS tunneling, which encodes data inside routine network lookups to move it past security controls undetected.[4] VPC-level network controls are the more reliable safeguard.

4. Instrument everything. Deploy immutable, append-only **audit trails** that record every meaningful action: file reads, **API** calls, commands executed, and AI agent decisions. Feed these records into your organization's **SIEM (Security Information and Event Management)** platform and configure anomaly detection for unauthorized access patterns or **data exfiltration** attempts. These logs serve a dual purpose. For compliance, they provide a defensible, time-stamped record of who did what and when. For organizational learning, they allow teams to replay experiments, understand what worked, and diagnose what went wrong.

5. Define governance and graduation criteria upfront. Before the sandbox goes live, establish approval workflows, experiment scoping rules, and spending caps. Require every experiment to declare a hypothesis and a rollback procedure so that teams learn deliberately rather than exploring without direction. Most importantly, define clear **graduation criteria**: an experiment moves to production only when audit logs show zero policy violations, access controls functioned as designed, data handling met compliance requirements, and security and compliance teams have formally signed off.

Quick Reference: Sandbox Building Blocks

Layer	Purpose	Example Tools
Isolation	Separate experimental workloads from live systems at the hardware or container level	Firecracker, Kata Containers, gVisor, E2B, Northflank
Data	Generate synthetic or anonymized datasets; scan for sensitive data before use	MOSTLY AI, Neosync, Tonic.ai
Access	Authenticate users and enforce least-privilege permissions tied to organizational roles	SSO + RBAC via existing IdP; short-lived session credentials
Network	Prevent unauthorized data movement into or out of the sandbox environment	Dedicated VPCs, private subnets, DNS firewalls
Monitoring	Record all actions; detect anomalies; maintain defensible compliance records	Append-only audit logs, SIEM integration, anomaly detection

Layer	Purpose	Example Tools
Governance	Scope experiments, cap spending, and define criteria for moving to production	Approval workflows, hypothesis templates, sign-off checklists

Operational Best Practices

Even a well-designed sandbox delivers limited value without the right operational habits. Leading organizations complement their technical infrastructure with practices that keep experimentation focused and productive.

- **Apply risk-based governance.** Reserve strict controls for high-risk AI applications (those processing client data or making consequential decisions), while permitting lower-friction experimentation in governed sandbox spaces for lower-risk use cases such as internal productivity tools or document summarization.
- **Run weekly experiment stand-ups.** Short, regular check-ins where teams confirm hypotheses, review preliminary results, and manage scope creep prevent experiments from drifting into unfocused exploration.
- **Embed cross-functional accountability.** Every AI initiative should include security, technology, legal, and business stakeholders from the outset. This prevents the common failure mode where legal or security is consulted only after an experiment is already being prepared for production.
- **Treat failure as reusable knowledge.** Document what did not work as carefully as what did. A well-documented failed experiment prevents other teams from repeating the same mistakes and builds institutional understanding of where AI adds real value and where it does not.

The Bottom Line

Secure AI sandboxes resolve the tension between speed and safety. They transform experimentation from a risk to be managed into a capability to be cultivated. Organizations that invest in governed, isolated experimentation environments do not just protect their data. They build the institutional capacity to adopt AI responsibly, turning curiosity into competitive advantage while keeping their most sensitive assets firmly out of harm's way.

The technology and tooling to build these environments already exist, and the frameworks for governing them are well established. What remains is organizational commitment: the decision to give teams a safe place to experiment, and the discipline to back that commitment with the right infrastructure, data practices, access controls, and governance. For organizations ready to take that step, the reward is an AI adoption strategy that is both ambitious and secure.

Glossary of Terms

The following definitions are provided to ensure accessibility for readers of all technical backgrounds. Terms are listed alphabetically and correspond to bolded terms on first use in the body of this document.

AI Agent – A software program powered by artificial intelligence that can perform tasks autonomously, such as answering questions, drafting documents, or executing multi-step workflows, often with minimal human intervention.

AI Sandbox – An isolated, controlled computing environment where teams can test and experiment with AI tools and models without any connection to live business systems or real data. Analogous to a "clean room" in litigation, where sensitive materials can be reviewed under controlled conditions.

Anonymized Data – Data that has been processed to remove or obscure all personally identifiable information (PII), making it impossible to trace back to any individual. Distinguished from pseudonymized data, which replaces identifiers but may still be re-linked.

API (Application Programming Interface) – A standardized set of rules that allows different software systems to communicate with each other. When this document refers to "API calls," it means requests sent from one system to another to retrieve or transmit information.

Audit Trail – A chronological, tamper-resistant record of all actions taken within a system, including who performed each action, what was done, and when. Similar in function to a chain-of-custody log in evidentiary proceedings.

BYOC (Bring Your Own Cloud) – A deployment model in which an organization runs a vendor's software within its own cloud infrastructure, rather than on the vendor's servers. This allows the organization to retain control over data residency and security policies.

Data Exfiltration – The unauthorized transfer of data out of an organization's systems, whether by a malicious actor, a careless employee, or an automated process. DNS tunneling (see below) is one method by which data can be exfiltrated.

DNS (Domain Name System) – The system that translates human-readable website addresses (e.g., www.example.com) into the numeric addresses that computers use to locate each other. DNS tunneling is a technique that encodes data inside DNS queries to move information past security controls undetected.

Graduation Criteria – The predefined conditions that must be satisfied before an AI experiment is approved for use in a live, production environment. These typically include passing security reviews, demonstrating compliance, and obtaining formal sign-off from designated stakeholders.

gVisor – An open-source security tool developed by Google that creates an additional protective layer between an application and the host computer's operating system. It intercepts the application's requests before they reach the core operating system, reducing the risk of a security breach.

Identity Provider (IdP) – A service that manages and verifies user identities for an organization, such as Microsoft Entra ID (formerly Azure AD) or Okta. When employees log in to internal tools, the identity provider confirms who they are.

Kubernetes – An open-source platform for managing and orchestrating containerized applications (software packaged with everything it needs to run) across multiple servers. Widely used in enterprise IT to deploy and scale software reliably.

Large Language Model (LLM) – A type of AI system trained on vast amounts of text data, capable of generating, summarizing, and analyzing human language. Examples include OpenAI's GPT-4 and Anthropic's Claude. These models power many of the AI tools currently entering the enterprise.

MicroVM (Micro Virtual Machine) – A lightweight, isolated virtual computer that starts in milliseconds and provides strong security boundaries. Each microVM runs as if it were a separate physical machine, preventing one workload from accessing another's data. AWS Firecracker and Kata Containers are leading implementations.

NIST AI RMF (AI Risk Management Framework) – A voluntary framework published by the U.S. National Institute of Standards and Technology in January 2023 (document NIST AI 100-1). It provides organizations with a structured approach to identifying, assessing, and managing risks associated with AI systems throughout their lifecycle. Its four core functions are Govern, Map, Measure, and Manage.

Open-Source Software – Software whose source code is made publicly available, allowing anyone to inspect, modify, and distribute it. Open-source tools mentioned in this document (e.g., gVisor, Agent Sandbox) are freely available but may still require technical expertise to deploy.

PII (Personally Identifiable Information) – Any data that could be used to identify a specific individual, such as names, Social Security numbers, email addresses, or biometric records. The handling of PII is governed by regulations including GDPR, CCPA, and HIPAA, among others.

Production Environment / Production Data – The live systems and real data that an organization uses in its day-to-day operations. "Production data" is actual client, customer, or business data, as opposed to synthetic or test data. A sandbox is specifically designed to be separated from production.

RBAC (Role-Based Access Control) – A method of restricting system access based on a user's assigned role within the organization. For example, a junior analyst may be permitted to run experiments but not to export data, while a security officer may have broader permissions. This follows the principle of least privilege: each user receives only the minimum access necessary to perform their function.

Shadow AI – The use of AI tools, platforms, or services by employees without the knowledge, approval, or oversight of the organization's IT or security teams. Analogous to "shadow IT," but specific to artificial intelligence tools. Shadow AI creates unmanaged risk because the organization cannot monitor what data is being shared or how it is being processed.

SIEM (Security Information and Event Management) – A centralized platform that collects, correlates, and analyzes security-related log data from across an organization's systems. SIEM tools help security teams detect suspicious activity, investigate incidents, and maintain compliance records.

SSO (Single Sign-On) – An authentication method that allows users to log in once and gain access to multiple related systems without re-entering credentials. SSO simplifies access management and makes it easier to enforce consistent security policies across tools.

Synthetic Data – Artificially generated data that mimics the statistical properties and structure of real data but contains no actual personal or sensitive information. Synthetic data allows teams to build and test AI models realistically without privacy risk.

Trust Boundary – A conceptual line in a system's architecture beyond which data or processes are considered untrusted. In a sandbox, the trust boundary ensures that nothing happening inside the experimental environment can affect or access systems outside of it.

VPC (Virtual Private Cloud) – A logically isolated section of a cloud provider's network that an organization can configure and control. A VPC acts like a private, walled-off network within a larger shared infrastructure,

providing network-level isolation for workloads running inside it.

Endnotes

1. IBM Security and Ponemon Institute, "Cost of a Data Breach Report 2025," IBM, 2025. Available at https://www.ibm.com/reports/data-breach

2. IBM, "What Is Shadow AI?," IBM Think, 2025. Available at https://www.ibm.com/think/topics/shadow-ai

3. National Institute of Standards and Technology, "Artificial Intelligence Risk Management Framework (AI RMF 1.0)," NIST AI 100-1, January 2023. Available at https://www.nist.gov/itl/ai-risk-management-framework

4. Unit 42 / Palo Alto Networks, "Bypass of AWS Sandbox Network Isolation Mode via DNS Tunneling," 2025. Available at https://unit42.paloaltonetworks.com/bypass-of-aws-sandbox-network-isolation-mode/