What We Have Learned in Four Years of AI Security

A Brief

Patterns from four years of AI security consulting, distilled into a working definition you can apply to your own organization.

July 2026 · Prepared by Argus AI

www.argus-ai.at

DISTRIBUTION: AUTHORIZED FOR PUBLIC RELEASE

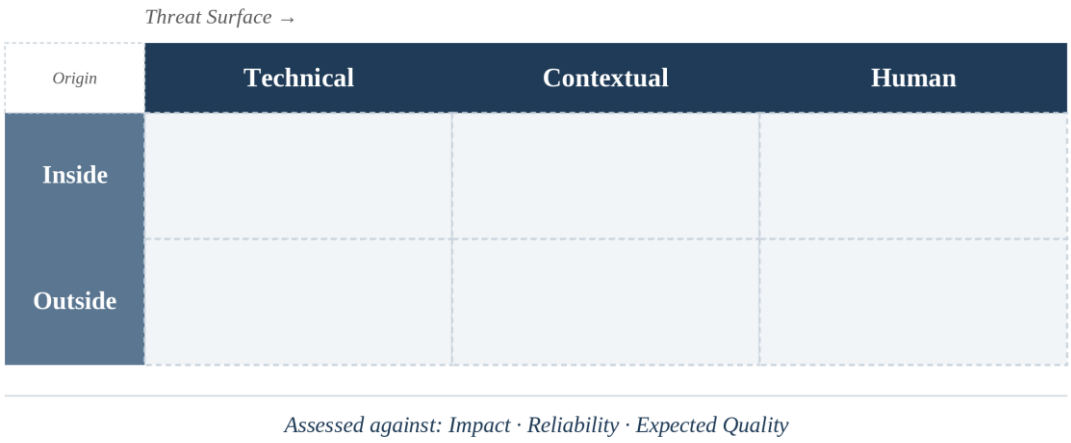
This paper distills patterns we keep encountering across AI security engagements, industries, and technology stacks, over four years of consulting work. It is not a framework you can buy off the shelf, and it is not a compliance checklist. It is the working definition of AI security we have arrived at after having to explain, defend, and operationalize the term for clients who each meant something slightly different by it.

AI security in general is a topic mostly flying under the radar. Even for organizations that do have it on their radar, it is a term that is not standardized and not attributed to a fixed set of methods, principles, or frameworks. Does a faulty implementation of an AI system on a technological level fall under “AI security”? Or is it the mere use of an LLM that counts? AI security needs to be specified before it can be enacted.

Our working understanding is manifold:

> *AI Security encompasses all activities of an organization that have an impact on the reliability and expected quality of an AI application, encompassing all technical, contextual, and human threats, from the inside as well as the outside of an organization.*

With this definition, we can break down the factors and attack vectors that influence the use of AI systems.



Impact, Reliability, Expected Quality

Impact can be measurable (a drop in output quality, slower response times, reduced availability) or unmeasurable: failed adoption because users stop trusting the outputs, or the simple fact that nobody in the organization would notice if something had already gone wrong. The unmeasurable side is usually the more dangerous one, precisely because it never shows up on a dashboard.

Reliability, for an AI system, cannot mean what it means for traditional software. We are not dealing with a guarantee that input X always produces output Y; that level of determinism belongs to a

different category of technology. Reliability here means defining, in advance, the band of acceptable variance an application is allowed to operate in, and being able to tell when it has left that band.

Expected Quality asks a simpler but frequently unanswered question: does the model do what it is supposed to do, at the quality level that was actually requested? That sounds trivial until you try to find out who defined "the quality that was requested" and against what benchmark. Most organizations discover, when asked, that no one did.

Technical, Contextual, Human: It Cannot Stop There

Technological factors are only one of three dimensions, and human and contextual factors carry at least as much weight.

Technical: adversarial AI, cyberthreats, misconfiguration, hardware faults, integration issues.

Contextual: is the system being attacked through its context, for example by flooding a deliberately slow, methodical, high-confidence system with a volume of decisions it was never built to handle at speed?

Human: misuse, misunderstanding, social engineering, and the many ways people adapt their behavior around a system once they know, or think they know, how it works.

From the Inside and From the Outside

Inside: insider threats, and the everyday erosion of controls that happens when the people who understand a system best also have the most reason and opportunity to work around it.

Outside: malicious external actors, but just as often vendor issues: a dependency on a third-party model, API, or platform whose security posture you did not choose and cannot fully audit.

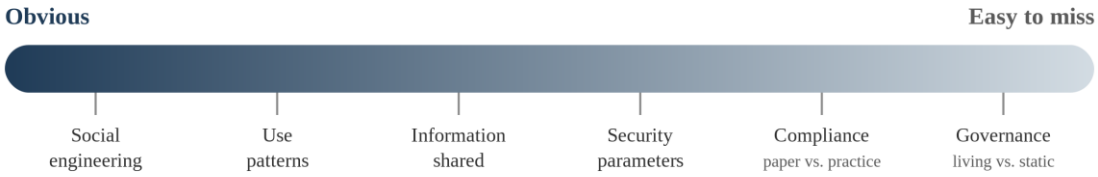
Our Insight from Practice

Across these dimensions, the single biggest observable issue is that almost no one has looked at their AI application through this full set of variables at once. Technical audits, or adherence to an industry standard, in a domain this complex and this volatile, do not protect you; they lull you into a false sense of confidence. AI systems are not, on a technical level, software in the sense that gives you an assurance that input X always produces output Y. If you require that assurance, use traditional software; don't use AI simply because it is the hot commodity of the moment. AI comes with wiggle room, and to an extent, an open-endedness of outputs. The effort to map technical depth onto an industry standard is well spent, but be warned of dogmatic adherence. A dynamic domain needs a standard that allows for adaptation. The same logic applies to regulation: if you operate in the EU, you comply with the AI Act, and you should. But a compliance certificate tells you that you have met a legal floor, not that you have covered technical, contextual, and human threats from the inside and the outside. Treat it as a floor, not a ceiling.

And it cannot stop there. Technological factors are one of three. Human and context are just as important, and can have at least as much impact on a system as the technical layer.

On the contextual side, we consistently see that the use case for AI is under-refined. The question “is AI actually needed here, and what are we trying to do?” too often falls victim to the drive for innovation. There is a useful discipline borrowed from product design here: real innovation is not saying yes to everything, it is saying no to all but the most crucial features. Without a clear answer to what the problem is and why AI is the right solution to it, you open the gate to contextual threats. An internal Q&A chatbot built to answer like a subject-matter expert, for instance, can end up over-communicating sensitive organizational knowledge to internal stakeholders who don’t need it, or are explicitly barred from having it.

Human threat runs from the obvious to the easy to miss. The obvious end is social engineering aimed at the model itself, or at the people operating it. The easy-to-miss end is everything that shifts once an AI system is in the loop: how people actually interact with it day to day, what use patterns emerge once the novelty wears off, how much information they feed into it (often more than they would ever tell a colleague, because “it’s just a tool”), whether the security parameters around its use survive contact with a deadline, whether compliance is met on paper or in practice, and whether AI governance is a living process or a document that was signed off once. In our experience, most organizations can answer the first two questions. Very few have seriously worked through the rest.



Most organizations have answered the left two. Few have worked through the rest.

Putting the Framework Together

Take the internal Q&A chatbot from above and run it through every axis at once. The root cause is contextual: the use case, give employees fast answers, was approved without anyone asking who should not get an answer. The vector is human: staff ask it things they would never ask a colleague, because a chat window carries none of the social friction a hallway conversation does. The origin is inside the organization: no external attacker is required, the system is doing exactly what it was built to do, for people formally authorized to use it. The impact splits in two: measurable, if it surfaces in an HR case or a compliance finding, and unmeasurable, in the slow shift of who inside the company knows what, which nobody is tracking. Reliability and expected quality both fail quietly here, not because the model malfunctioned, but because nobody defined, in advance, what it was allowed to be right about. That is the exercise behind the first item below: can you do this, on demand, for your own AI application, without phoning a colleague?

What You Can Do Right Away

- Look at your AI implementation through the lens of our definition above. If you can attribute a concrete answer to every aspect (technical, contextual, human, inside, outside) without having to ask ten different people, you are already in the top 10 percent of organizations we've seen.
- Is the technical, contextual, and human aspect of your AI application clearly separated and understood? For each category, what are the potential threats, and how likely and how impactful is each one?
- How important is AI to your business today? How much of your organization's day-to-day operations would actually be affected if that AI became unavailable tomorrow?
- Name one person, by name, who owns AI security end to end today. If the honest answer is "IT" or "nobody," that is your most urgent finding, not a footnote.

Where This Leaves You

None of this is a reason to slow down on AI. It is a reason to stop treating "AI security" as a checkbox next to your existing cybersecurity program, and start treating it as its own discipline, with its own definition, before an incident forces that definition on you. If you want to work through your own AI implementation against this framework, get in touch with us. This is exactly the kind of conversation we have with clients before things go wrong, not after.

About Argus AI

Argus AI is an Austrian artificial-intelligence company working at the intersection of AI safety and defense. We help national and international organizations implement AI safety standards and stress-test their systems against real-world failure modes, ensuring that high-stakes AI is deployed reliably, securely, and in line with emerging regulatory requirements. Alongside our advisory work, we develop applied AI systems at the intersection of machine learning and unmanned aerial platforms. Founded in Burgenland, Austria, we are a focused team of specialists combining deep technical expertise with a rigorous, standards-driven approach to AI utilization.

contact@argus-ai.at