The world is how we shape it*

sopra steria

White paper

AI LLM and cybersecurity

World such that we Shape*

Why secure the choice of your LLM?

One **Large Language Model (LLM)** is an artificial intelligence system based on neural networks of the transformer architecture, capable of understanding and generating natural language. Trained on billions of tokens, these models are now capable of summarizing, translating, classifying, dialoguing or assisting business processes. Their power is based on their contextual capacity, but their behavior remains **Non-deterministic**, **probabilistic** and **Highly manipulable**.

By 2025, more than **150 LLM templates**[1] are actively referenced, including one **Thirty major business or sovereign models**. Adoption is exploding[2] : **more than 90%** of organizations report testing or deploying GenAI use cases, but **only 5% say they are ready** in terms of cybersecurity.

In this context, the choice of an LLM can no longer be based solely on **raw performance criteria, cost or latency**. It must also include a **structural reading** : What is the degree of control required?

What are the regulatory constraints? And above all, **What are the cyber risks associated with the use of the model?** Data leaks, semantic drifts, technical dependencies, unforeseen behavior?

The **attacks on LLMs** do not resemble traditional threats. They exploit linguistic, visual or semi-structured vectors, with techniques such as:

- **Prompt injection** or Jailbreaking,
- **Data poisoning** in training corpora or RAG databases,
- **Handling of standalone agents or tooled plugins**,
- **Information leakage by controlled hallucinations**.

Unlike traditional information systems, **LLMs were not designed to operate in a hostile environment**. They must therefore be integrated with increased vigilance, by combining **Risk analysis, choice of secure architecture, continuous monitoring**, and strong control over data.

1: Source: HuggingFace Open LLM Leaderboard
2: Source: Lakera GenAI Security Readiness Postponement, 2024.

Why LLMs are transforming the cyber posture of defense

Large language models (LLMs) are radically transforming cybersecurity dynamics. They are no longer just **tools used in cyber defense postures** (enrichment of malicious information, etc.) : they are also used **to automate attacks**.

Increase in security breaches and vulnerabilities

of companies have experienced at least one AI-related incident, and 60% say they are ill-prepared for this new risk.
(Metomic, Lakera)

LLMs blur the lines between defense and threat:

Attackers exploit models like WormGPT or FraudGPT to generate phishing, exploits, or malicious code.

of professionals believe that AI makes phishing more credible and harder to detect.
(Lakera, Arctic Wolf)

of companies declare themselves " **Mature** " in cybersecurity LLM.
(Techradar, Cobalt.io)

The generalization of assistants connected to critical systems (RAG, Chatbots internal, business automation) **Expands the attack surface** : These interfaces become entry points for manipulations such as prompt injection or contextual exfiltration.

Adoption precedes securitization: LLMs are becoming established faster than defense mechanisms are adapting. This requires a complete overhaul of threat models.

LLMs disrupt cybersecurity because they **Accelerate the automation and sophistication of attacks** while at the same time offering new defense tools. This situation requires companies to be more vigilant: adopt LLMs in innovative ways, while rethinking security practices (processes, training, tools) to counter the new attack vectors they introduce. Ignoring these developments would expose you to unforeseen or hard-to-anticipate attacks, while adopting LLMs in a controlled manner can provide a strategic cybersecurity advantage for the most proactive organizations.

LLM Technology Panorama

The landscape of LLMs in 2025 is not limited to their architecture or license (open source vs. proprietary), it also reflects **and the evolution of capabilities.**

LLMs are still based on Architecture **Transformer** (introduced in 2017), which allows for efficient scaling. With a constant growth in the number of parameters (GPT-4, LLaMA 3, DeepSeek R1) and the emergence of more efficient architectures such as Mixture of Experts (Moe).

Multimodal and GAR

LLMs become **Multimodal**, capable of processing text, images or audio. Models like *DeepSeek Janus-Pro-7B* combine generation and visual interpretation in a single interface.
At the same time, LLMs are equipped via **External tools** (APIs, plugins, code execution, basic querying) to go beyond their native limits. This approach, known as the **RAG** (Retrieval-Augmented Generation) or **Tool Use** is now common in companies.

Tuning and alignment

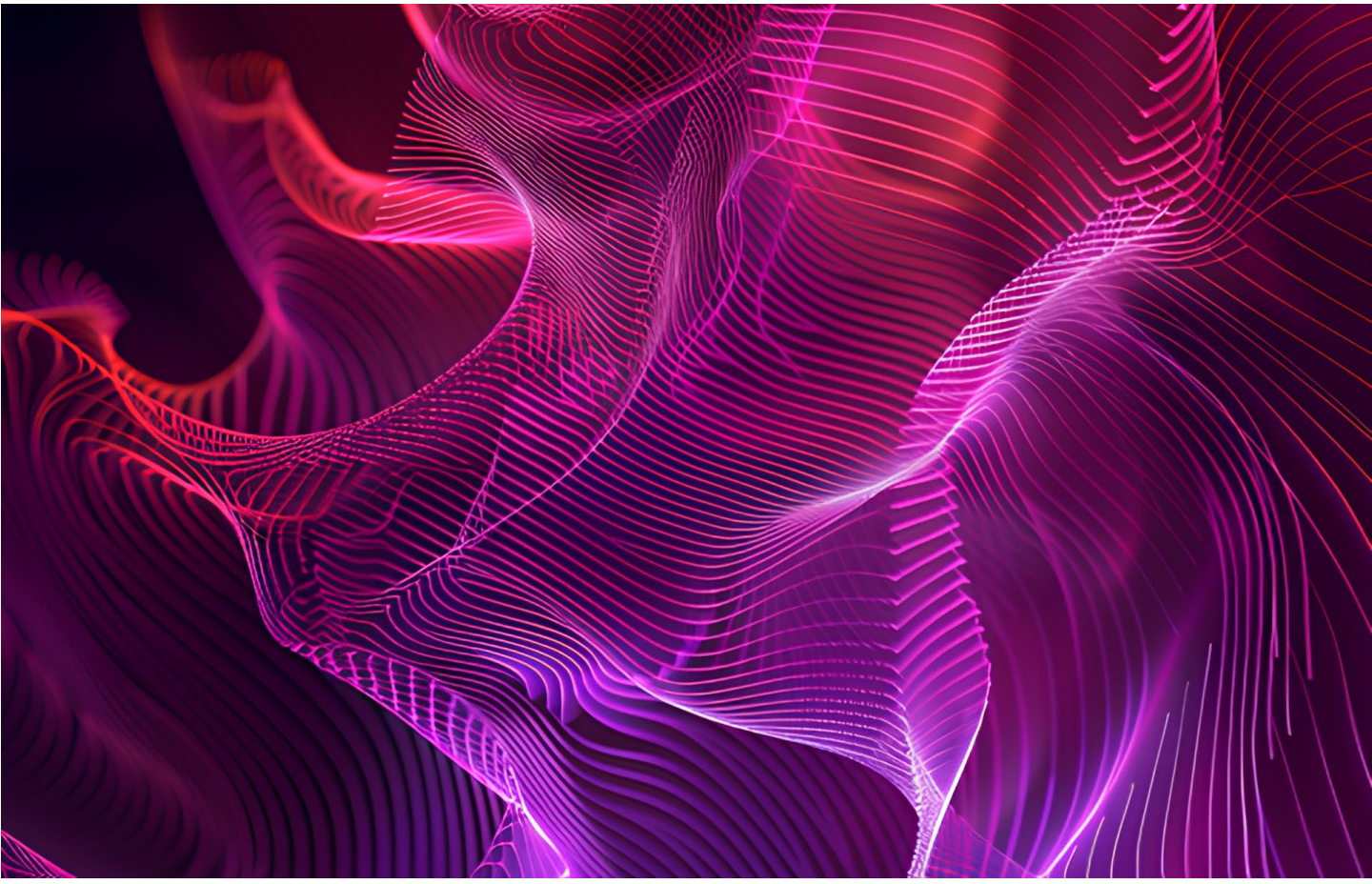
Tuning and alignment are crucial aspects of LLM technology: initially, the raw models (pre-trained in general language mode) were adapted via the tuning instruction and the **RLHF** (Reinforcement Learning from Human Feedback) to better follow user instructions while respecting constraints. Each publisher offers specialized variants (Claude Opus vs Sonnet, Gemini Pro vs Flash).

The LLM market in 2025 is characterized by:

- **Efficient proprietary giants** (GPT-5, Claude 4, Google Gemini...) pushing the context and reasoning ever further, often available via cloud API only.
- **Open source models** increasingly competitive, thanks to academic and industrial contributions (LLaMA, BigScience BLOOM EleutherAI, Mistral AI, DeepSeek…).
- **Specialized LLMs** : templates for the code (e.g. LLaMA), for speed ("Turbo" or "Flash" versions), for multimodality (text+image)
- **Sovereign models:** These models are independent of the major providers, often for reasons of digital sovereignty, privacy or local language support. (Mistral, BLOOM)

Cyber Analysis

Regardless of the family of belonging (open source, proprietary), LLMs share a structural weakness: they **are not** designed to be **Natively secure** in a hostile environment. **Proprietary models** offer safeguards (which do not guarantee complete coverage against semantic attacks) but remain **Black boxes**, **difficult to audit**. **Open source models**, offer complete control of the model: downloadable weights, fine-tuning capability, deployment on a sovereign cloud or on-the-shelf premise. However, their safety depends on **therefore entirely on the architecture deployed around the model** (middleware, sandbox, scoring, filtered RAG, logs, supervision).



Architectures integration

When a company wants to use an LLM, there are several integration options available to them. Each of these approaches has advantages and disadvantages.

Public SaaS LLM

Access via web interface (ChatGPT, Claude, Gemini, etc.) is common for exploration phases or in SMEs. It does not require **any technical integration**. But he exposes the organization to the **shadow AI** : **Lack of supervision, zero traceability, possible data leaks**. This mode is incompatible with GDPR or ISO 27001 compliance whenever sensitive data is at stake. The risks of prompt injection or toxic generation are real, without the possibility of post-incident analysis.

LLM As-A-Service

APIs (OpenAI, Anthropic…) Enable more professional integration, with powerful, scalable templates. However **data leaves the IS**, **Customization remains limited, and the supplier imposes its safeguards**. Lack of visibility into logs, model logic, or updates poses a problem in the event of abuse or compromise. Without an intermediate proxy (filtering or monitoring), the use of APIs creates a critical blind spot.

Recommended mitigations:

- Integrate a middleware for filtering and prompt logging.
- Isolate usage in a cloud VPC or dedicated network.
- Enable end-to-end encryption and external logging.

Self-hosted LLM

Deploying an LLM on your infrastructure offers maximum control over data, personalization, and compliance, a choice often favored in sensitive industries (healthcare, defense, finance). But this level of autonomy comes with strong constraints: significant GPU resources, ML skills, maintenance effort. **Above all, just because the LLM is hosted locally does not mean that it is trustworthy**. In the absence of specific security controls, the model remains vulnerable to AI-specific attacks. It's not enough to control the infrastructure; You also have to master the logic and outputs of the model, and establish **A verifiable level of trust in every interaction**.

Recommended mitigations:

- Couple with local moderation tools (sandwiched LLM).
- Deploy in siloed environments.
- Regularly scan usage with LLM security tools (e.g.: Lakera Guard, Mirror Security).

Fine-tuning

Fine-tuning **Improves business performance** by adapting the model to specific cases. It can be done via API (OpenAI,…) or locally on an open source model. But it **Increases the attack surface** : A poorly mastered dataset can inject biases or open the door to jailbreaks. If it is poorly supervised (no secure pipeline, no post-training robustness tests), the **fine-tuning can weaken the initial safeguards**, with no ability to go back. Outsourcing tuning data requires rigorous encryption or anonymization.

Safety risks specific to LLMs

Many threats to LLMs can be reminiscent of classic web or system vulnerabilities, but transposed to the AI context. For example, **prompt injection** resembles SQL or XSS injections (injection of unfiltered content into a runtime context) and the **model Denial-of-service** (overwhelming the LLM with requests to slow it down or be expensive) is reminiscent of traditional DDoS.

Likewise **the Poisoning of training data** can be compared to an attack of Supply Chain software (introduction of a backdoor via a third-party component), here the component is data, vulnerable like code. And the **model flight** is a new face of industrial espionage, consisting of stealing valuable software intellectual property (the weights of the LLM) rather than source code.

Top Security Concerns About Adopting Generative AI

1. Exposure of sensitive data to underlying LLMs
2. Enemy attacks on generative AI tools (, techniques designed to hijack, deceive or manipulate AI models)
3. Lack of guardrails or controls in generative AI tools
4. AI hallucinations
5. Insufficient public regulations for the use of generative AI

Cyber Analysis

However, there are notable differences. On the one hand, LLMs introduce **By-design vulnerabilities** : they will always have a certain probability of being wrong or manipulated, because they do not work on deterministic rules but on statistical correlations. This makes them less predictable than traditional technologies, and therefore more difficult to secure perfectly (you can't patch a "hallucination" with a binary patch like you would patch a buffer overflow).

On the other hand, LLM attacks often combine **Technical and human angles** : where a purely technical SQL injection exploits a code flaw, a prompt injection exploits language understanding and deceives a model that *imitates a human*. We are closer to the **social engineering** applied to an AI.

Another distinction is that some classic attacks can become easier to perform when an LLM is used internally. Like what **Phishing** is multiplied in quality and volume thanks to LLMs, so an attacker can launch much more credible campaigns.

Conversely, some technical attacks such as memory exploitation or RCE remote code execution do not directly target the LLM, we do not exploit its binary, but its logic. However, these attacks can occur indirectly via poorly designed plugins, which execute without verification what the LLM generates, thus turning the LLM into an attack vector.

Threat modelling for LLMs

The integration of LLM into information systems disrupts traditional approaches to threat modeling. Their probabilistic nature, their natural language interaction and their possible autonomy make risks more complex to identify. It is therefore essential to adapt proven methods to the specificities of LLMs, while exploring how these models can themselves help anticipate attack scenarios.

DFD (Data Flow Diagram)

A mapping of sensitive flows (user input, LLM API call, RAG database, output to SI). Critical areas: Internet/API border, vector storage, unfiltered outputs.

STRIDE

The STRIDE method (Spoofing, Tampering, Repudiation, Information disclosure, Denial of service, Elevation of privilege) is a framework **A structured approach to identify and analyze potential threats associated with each component of the DFD.**

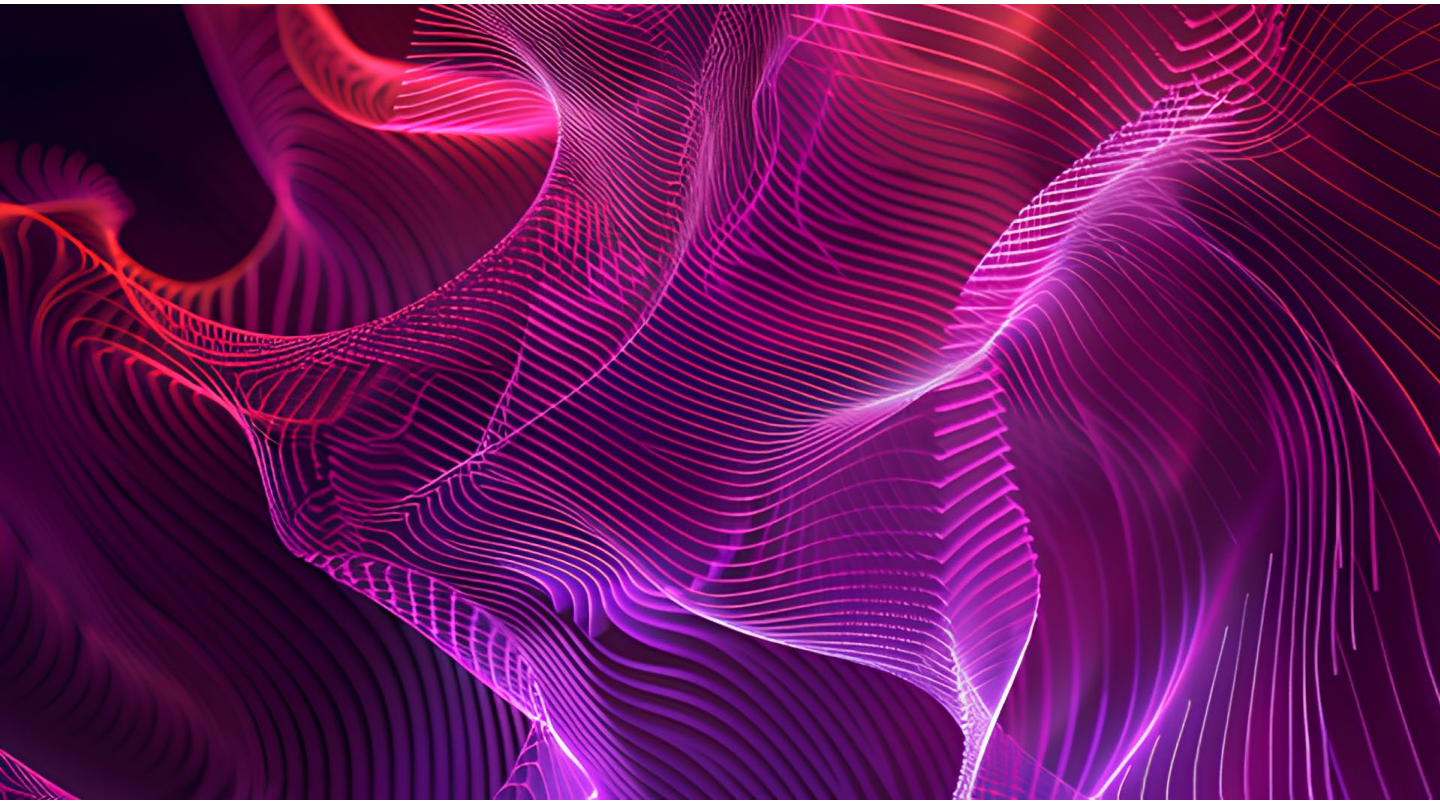
- **Spoofing** : Token theft/session,
- **Tampering** : basic corruption RAG/system prompt,
- **Repudiation** : signed logs, traceable audit of requests
- **Information Disclosure** : Unintentional disclosure of sensitive information via responses or data storage
- **Denial of Service** : heavy or massive prompts,
- **Elevation of Privilege** : hijacking plugins

PASTA (Process for Attack Simulation and Threat Analysis)

An approach, focused on attack scenarios, often carried out in **7 steps** (definition of business objectives, definition of the attack surface, decomposition, threat analysis, simulation of attacks, impact analysis, treatment).

Cyber Analysis

Traditional methods such as DFD, STRIDE, or PASTA are still useful for mapping LLM threats, but they need to be adapted to their non-deterministic nature. **Traditional DFDs struggle to represent the grey areas introduced by probabilistic model behaviors or the integration of dynamic plugins. STRIDE provides good coverage of technical threats (spoofing, tampering, Back...), but does not fully capture the linguistic or contextual drifts specific to LLMs (hallucination, implicit manipulation)**. **PASTA offers a more realistic approach by integrating attack scenarios and business issues**, but it remains underutilized in AI architectures. In practice, too few organizations formalize these analyses or equip them with tests opponents. For these methods to be effective in the LLM context, they must be enriched with behavioral analyses, semantic logs, and observational tools that integrate natural language as an attack vector in its own right.



How to choose your LLM

Cartography

The choice of an LLM must first be based on a **Clear use case mapping** : types of data processed (sensitive, public, confidential), business criticality, and position of the model in the flows (support, decision, user interaction).

Reconciling performance, control and safety

The right model is the one that **Aligns with both business needs and the level of control needed** : hosting (SaaS, Self-Hosting), observability, isolation, personalization.

of incidents related to generative AI in companies come from a poorly contextualized deployment of the model.
(Lakera)

Cyber Analysis

Many organizations **omit this step**. As a result, powerful LLMs are deployed on innocuous tasks... or, worse, undersized on critical cases. A misconfigured LLM in a sensitive context can:

- **Violating the GDPR, AI Act** (e.g., patient data leaks);
- **Generate regulatory errors** (legal, tax, health);
- Be **vulnerable to attacks** prompt injection type, undermining the integrity of the system.

Cyber Analysis

To think that an open source model is secure because it is self-hosted is an illusion.
Without sandbox, logging, nor prompt filter, sovereignty is superficial.
Conversely, a non-isolated SaaS LLM can **Leak via API logs, plugins, or metadata**.
Environments with agents or RAGs are highly exposed, especially if they do not include **sandbox, semantic verification or supervision**.

Safety alignment

It is then necessary to **assess the requirements for confidentiality, integrity, auditability, traceability and resilience**.

Examples:

- Generating dangerous code;
- Amplification of falsified content;
- Outdoor exposure becomes a target for reverse engineering or fuzzing.

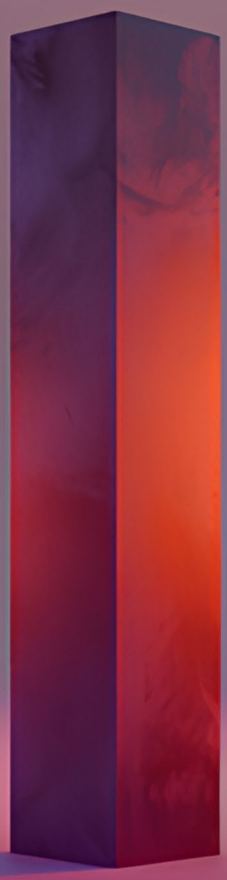
Towards a weighted choice grid

In 2025, the majority of generative AI deployments fail to integrate security as a variable of choice. Yet, a good LLM benchmark is not limited to the MMLU score (Massive Multitask Language Understanding which measures an LLM's ability to reason and answer correctly on a wide range of academic and professional topics) or speed: it must **Include behavioral threat resistance scores** (prompt injection, jailbreak, hallucination, attacks adversariales), **Traceability guarantees**and **Integration capabilities** in an architecture Zero trust or DevSecOps.

LLM benchmark

To assess the robustness of LLM models in the face of cyber threats, several initiatives are emerging:

- **Lakera LLM Risk Index** : Evaluates resiliency to prompt injections, jailbreaks, and inadequate filtering attacks.
- **CyberSecEval** : Testing LLMs for exposure to malicious requests, injection, unethical response, or dangerous code.
- **PromptFoo** : platform allowing customized tests on security scenarios, behavior and robustness (injection, contextual confusion, etc.).
- **OWASP LLM Top 10 / MITRE ATLAS** : Structured threat repositories to be included in model audits.



LLM benchmark

The weighting and evaluation presented here are based on a generic framework designed for use cases in companies. However, security priorities can vary greatly depending on the sector (health, finance, industry, etc.) and the contexts of use (LLM in interaction with the public, in internal backend, in business support,...). This benchmark should be seen as a structuring starting point, to be adapted according to regulatory constraints, the level of sensitivity of the data, or the risk tolerance of each organization. It is recommended to recontextualize these scores via an analysis specific to each project (threat modeling, audit, etc.), in order to arrive at a security policy that is truly aligned with uses.

To help objectively compare the robustness of the main LLM models against attacks, we have built an overall security score based on five key criteria: prompt injection, jailbreak, hallucination, secure personalization capability and performance on the benchmark tests mentioned above. Each of these criteria is weighted according to its impact on the actual attack surface in the enterprise.

The final score, scored out of 10, does not claim to reflect the absolute safety of a model, but provides a standardised basis for comparison**. It helps identify the strengths and weaknesses of each LLM, linking them to concrete threats (OWASP LLM Top 10, MITRE ATLAS, etc.). This benchmark does not replace a contextualized audit, but is a useful starting point to guide deployment choices and the implementation of additional safeguards.**

The overall score presented (out of 10) is a weighted average based on 5 key axes.

Example for GPT-4o: *(0.25×9) + (0.20×7) + (0.15×6) + (0.15×9) + (0.25×8.2) = 7.8 / 10*

Criterion	Weighting	Description	Justification
Prompt injection	25 %	Resistance to adversarial manipulation of user input	This is the most common attack, often an entry point for more complex attacks (exfiltration, privilege escalation).
Jailbreak / Bypass	20 %	Ability to enforce moderation rules	Critical threat in sensitive cases (compliance, ethics, reputation). The model's ability to enforce its safeguards is strategic.
Hallucination	15 %	Ability to provide reliable answers in critical scenarios	Less often exploited directly, but very risky in regulated sectors (legal, health, etc.).
Custom Guardrail	15 %	Ability to define and apply tailor-made security policies	Reflects the model's ability to be securely integrated into a business context. Important to limit drifts in a real environment.
Score Promptfoo	25 %	Average of scores obtained in test cases adversarial	Measuring Robustness to Scenarios opponents. High weighting because it is a dynamic and reproducible indicator.

The scores below give a **Synthetic view of the observable defensive posture** of all the models. It is not a substitute for a contextualized audit or dynamic analysis in a real environment.

- A score of **7.8 or higher** indicates good resistance to attacks such as prompt injections, jailbreaks, data leaks, or misaligned behavior.
- A score around **6.5 to 7.5** suggests a **correct but uneven maturity**, with certain weaknesses that will have to be compensated for by integration.
- A score **less than 6** reflects a **wider attack surface or less effective guardrails** : these models will have to be strongly supervised in terms of architecture, filtering, sandboxing, etc.

Model	Prompt injection	Jailbreak	Hallucination	Guardrail Custom	Score Promptfoo	Score Risk Lakera	Overall Security Score
GPT 4o	9.0	7	6	8	8.2	Low	7.8
Claude 4 Sonnet	9.0	6.5	7.5	8	7.8	Low	7.825
Gemini 1.5 Pro	7.5	6.0	6.5	6.5	6.9	Moderate	6.675
Gemini 2.0 Flash	7.0	6.5	6.0	7.0	6.9	High	6.575
LLaMA 3 70B	6.5	6.0	5.5	5.5	6.2	Moderate	6.125
Meta LLaMA Maverick	6.0	7.0	5.5	5.0	5.8	High	5.925
DeepSeek V3	5.0	4.5	6.0	4.0	5.7	High	5.425
Mistral 7B	4.0	3.0	4.5	3.0	4.5	High	4.025
DeepSeek R1	3.5	2.5	4.0	2.0	3.9	High	3.625

Why is it hard to reach 10/10?

- The attacks **Evolve faster** that the protections
- **LLMs are probabilistic and not deterministic,** which makes their behavior unpredictable in some cases.
- **Safeguards (guardrails) can be circumvented** via advanced techniques (adaptive prompt injection, Jailbreaking recursive).
- **Proprietary models (GPT-4o, Claude...)** do not allow for a auditability complete.
- **Finally, scores measure performance in a given context,** however, integration in companies (plugins, fine-tuning, RAG) reintroduces vulnerabilities.

Outlook 2025-2026

Here are some major trends and insights, along with their security implications:

Autonomous agents and use of tools : LLMs will increasingly be **wrapped in agents** capable of calling APIs, interacting with environments (OS, browsers, etc.) and performing complex tasks autonomously. Projects such as AutoGPT, LangChain or OpenAI (functions in the API) open up this path. This will make it possible to automate entire processes, but it will **also increase the risks** uncontrolled actions (cf. over-autonomy). We can expect the emergence of **specific security frameworks for AI agents**: for example, the Cloud Security Alliance initiated the Framework *MAESTRO* for agentic AIs.

Sovereign LLMs : As a result of the impetus of Europe and other regions, we will see the proliferation of **Local models** adapted to regional languages and contexts, often open source. For example, by 2026, OpenEuroLLM should deliver production-usable models for EU languages. Similarly, large companies could train their own "in-house" LLMs to free themselves from third parties (there is more and more talk of *"Private GPT"* for large groups). These models *Sovereign* will give more control (data sovereignty, alignment with local values, etc.), but will raise the question of **their updating and security** : without the means toOpenAI, will they be able to keep up in quality AND in safety?

Regulatory changes : The legal framework around AI will be strengthened. For business users, sector-specific regulations such as **DORA** (Digital Operational Resilience Act, financial sector) and **NIS2** (network and information security, critical sectors) will require them to include AI systems in their risk management

Maturation of practices DevSecOps AI : Currently, many teams are developing LLM-based POCs without fully integrating security into the cycle. By 2026, we expect DevSecOps practices to natively include AI. This means: integrating **Model-specific security steps** in CI/CD (prompt scans, analysis of models for biases or weaknesses before deployment, automatic robustness tests).

In conclusion, the years 2025–2026 will see **the secure industrialization of AI**. We will move from the emergent stage (where everyone did their own rules) to a more normalized stage, with frameworks (technical and legal) consolidated. LLMs will no longer be an experimental novelty but integrated into many critical products – security will have to be up to the task. Defenders will also have better AI-powered tools to counter threats. We can imagine next-generation SIEMs where AI collaborates with analysts in real timeto detect a complex attack, or intelligent honeypots conversing with automated attackers to lure them.

AI SECURITY SERVICE OFFER

A comprehensive portfolio of targeted services and solutions

Types of services

- Advisory Services
- Expertise & Integration
- Managed Services

Prevention

- AI Risk Management
- Governance & Dashboards
- AI Compliance Act
- Functional & Technical (Offensive) Audits
- Awareness & Communication

Detection & Response

- Threat Intelligence & CERT Augmented
- Shared crisis management with the AI provider
- Vulnerability Management (AI, LLM, Agents, Protocols)

Protection

- Secure AI architecture
- Security MLSecOps / LLMSecOps
- Data and flow security
- Mastery of Shadow AI
- Specialized red teaming
- Protecting Agent Identity

DISCOVER OUR OFFERS CYBER

The world is how we shape it*