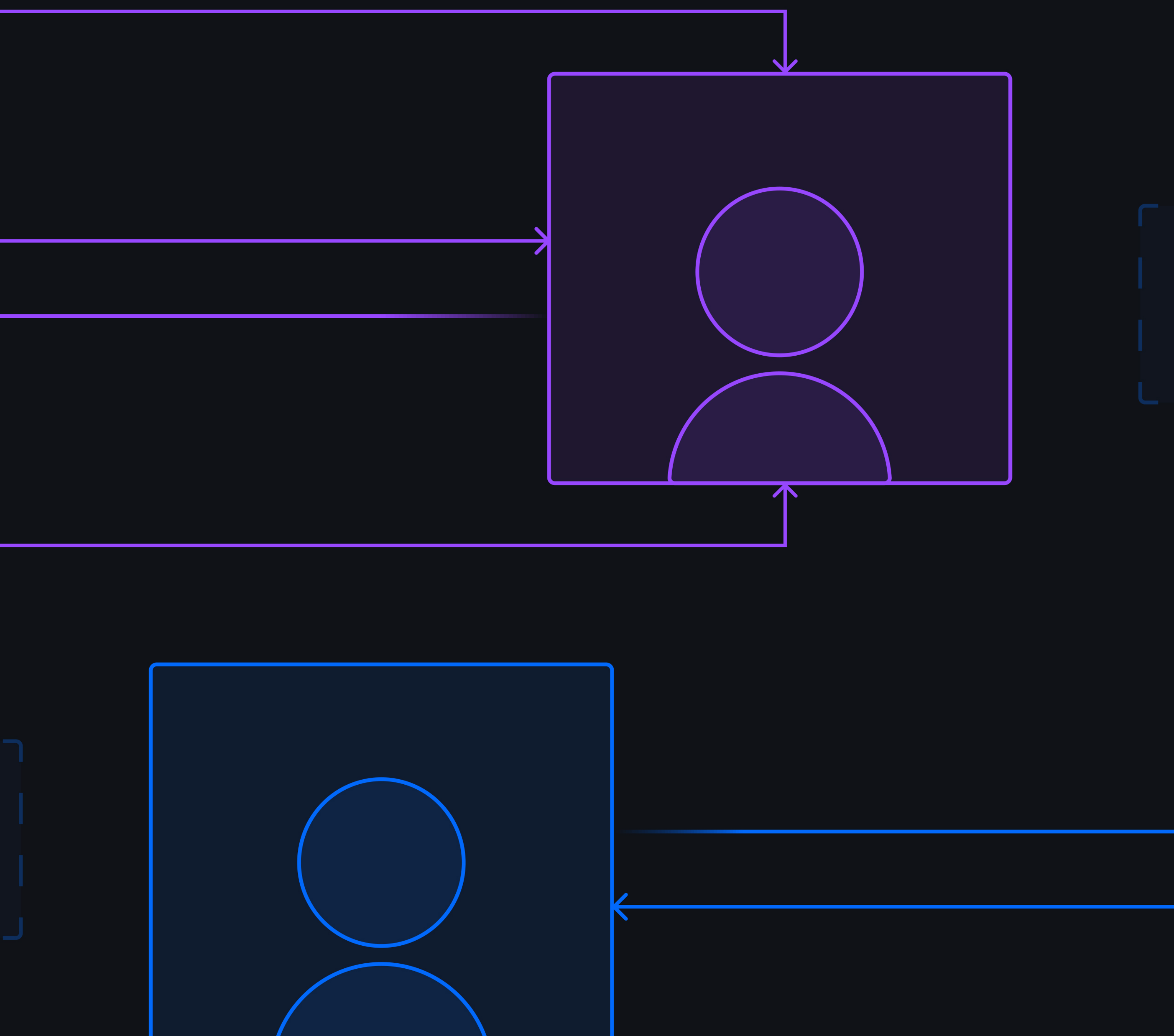
Aiceberg

# Security for Agentic AI



# Security for Agentic AI

The technologies that underpin large language models have been in development for over a decade. However, the launch of ChatGPT in November 2022 showcased the potential for human-machine interaction using natural language, capturing public interest and igniting a mix of excitement, concerns, and policy discussions. This release led to an unprecedented surge in investments, research, and development in the field. Initially, the use cases for such technology were limited and relatively straightforward, whereas we are now entering a new era characterized by the emergence of agentic AI, which promises to supercharge the degree of AI-driven automation and autonomy, allowing AI-driven agents to carry out increasingly complex tasks.

## Applications versus AI Agents

An application is a software program designed to perform specific tasks for users, such as word processing, web browsing, or data analysis, often requiring direct user interaction, whereas an AI agent operates autonomously, increasingly capable of planning and performing tasks that it deduces from the user input. Technically, therefore, any AI that can perform tasks on behalf of a user's natural language input is an agent. If we ask ChatGPT to summarize an article, we give agency to the large language model – authority to perform a specific task on our behalf. While we would not necessarily use the concept of agency in the workplace for example, we would find the same principle present there. We might ask a coworker to provide a summary or to prepare a presentation, in which case we give agency to that individual. Agentic AI use cases often involve moving agency from a human to a machine. An AI-driven agent that can book an entire holiday itinerary for us holds agency for tasks that, in the past, would have involved at least one human agent.

## Attack Vector Bonanza

While such technological outlooks are exciting, they come with significant risk due to the wide range of attack vectors that such environments expose. Today's generative AI use cases are mostly based on a single LLM, consisting of natural language input, (optional) context retrieval, and natural language output by the large language model. While agentic AI will equally consist of natural language input and natural language output or feedback, the scope and complexity carried out by the machine are increasing as it now also encompasses elements such as (behavioral) profiles, memory, planning, and actions which will be used by the agent(s) to achieve the respective objective for the user. This will include tasks such as information retrieval from private and/or public sources, triggering of other agents, or production of source code on the fly. Attack vectors therefore include the change of profile (behavior), change or exposure of memory, manipulation of tasks, generation of insecure code, calls to malicious websites, and manipulation of contextual data or user input to name a few.

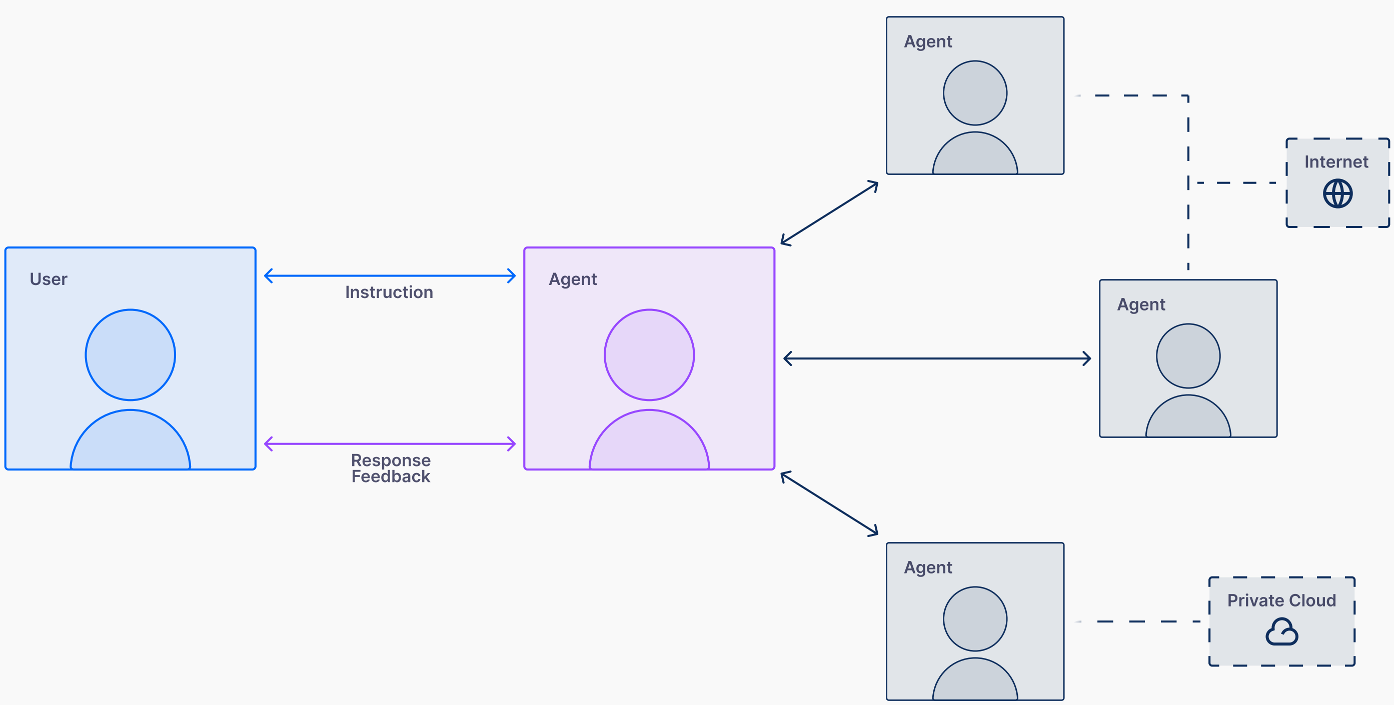
## AI Security versus Securing AI

Not all the above-listed attack vectors are novel, and it is therefore important to distinguish between AI Security and Securing AI. The latter involves protecting the entire environment in which AI is operating – from data, application, and network security, to access management and vulnerability management. AI Security encompasses capabilities specific and exclusive to the utilization of advanced AI systems, predominantly concerned with detecting security risks and threats. An agent performing API calls to retrieve data from a website, for example, would be part of Securing AI as there are effective tools today to monitor API calls and network access. What this function cannot provide us with and where we would need to rely on AI Security therefore is to determine whether the API call actually is in line with the original user instruction. We cannot read a co-worker's thoughts or comprehend how exactly he or she is structuring thought to plan for a given task, but we could observe the execution of each task by the co-worker and by doing so assess whether each task or step is in line with the provided objective.

Large language models have enabled a new way of user-to-machine interaction, allowing instructions in natural language without any restrictions other than what is imposed by humans (grammar, syntax, flow, etc.) The challenge of Securing AI is to align semantic information with syntactic information in real time. In other words, are the steps and code executed by an AI agent in line with the user's intention and objective? While it is possible to establish safeguards for both individually (language + code), it is the assurance of semantic and syntactic alignment that will be most critical for safe and responsible use of AI-driven agents.

In addition to securing end-users and AI operators from adverse AI outcomes, AI operators also need to be secured from the end-user itself, which could be a sanctioned user attempting to use an agent in an unsanctioned way (a retail banking service agent being asked by a user for advice on medication) or a malicious user trying to extract information from the agent or the model.

FIGURE 1



### Figure 1

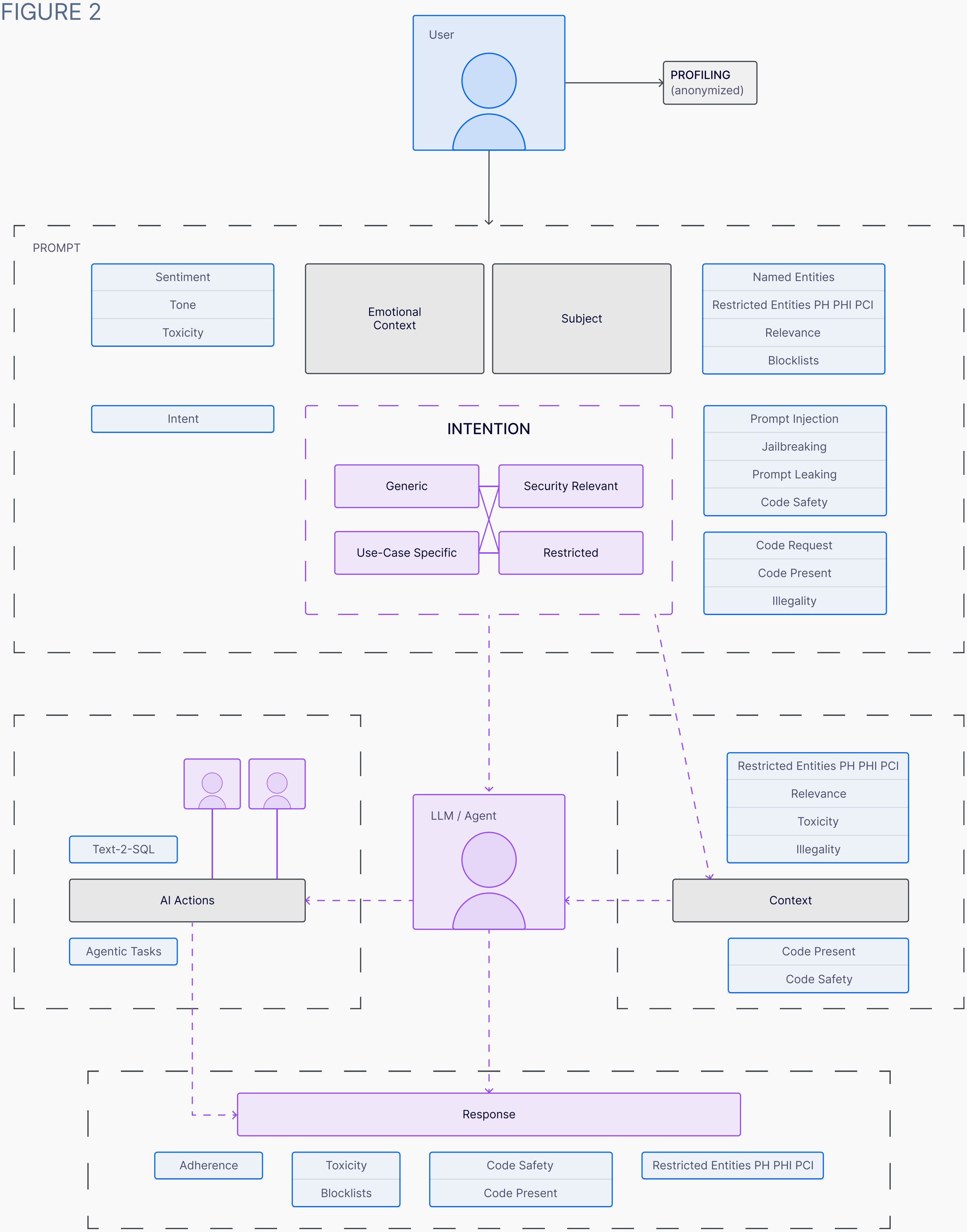
Figure 1 shows a simplified topology of agentic AI usage with a master agent performing planning to derive individual tasks and assign tasks to other agents, which in turn perform actions to satisfy the master agent's task objective. For simplicity, other functions such as memory, profile, or orchestration have been omitted. Actions in the above example include data retrieval from private and public sources, as well as API calls to execute code on third-party systems.

## Staying in control

AI Security as described earlier provides us with security and safety measures that are detection driven and designed to avoid unintended, malicious, or catastrophic outcomes. As none of the AI models involved in large language or agentic use cases are explainable, the optimum point of validation and enforcement that permits AI operators to stay in control is the point of model input and output. These points are accessible and transparent to every AI Operator and remain stable independent of technological changes within the agentic environment. In other words, as the inner workings of AI-driven agents or AI in general will change over time, human (natural language) input and output (or at least feedback in human-readable form) will always be constant. This also means that we should not rely on the same technology for agentic outputs (LLMs, LAMs) and agentic oversight. Using a large language model or models of comparable architecture for in- and output analysis and validation would be the proverbial fox guarding the hen house. LLM-based analysis of user input and machine output is slow compared to alternatives, adding significant latency to the end-user experience. Scaling an LLM for real-time analysis of high volumes of in- and outputs is costly and the fact that a black box model is being used to observe and police other black box models represents a potential compliance issue and a single point of failure. An alternative is to use significantly smaller and simpler models that are explainable, each being specialized on one aspect of in- and output analysis such as intent classification, PII detection, source code vulnerability, semantic and syntactic alignment, etc.

# Anatomy of securing agentic interactions

FIGURE 2

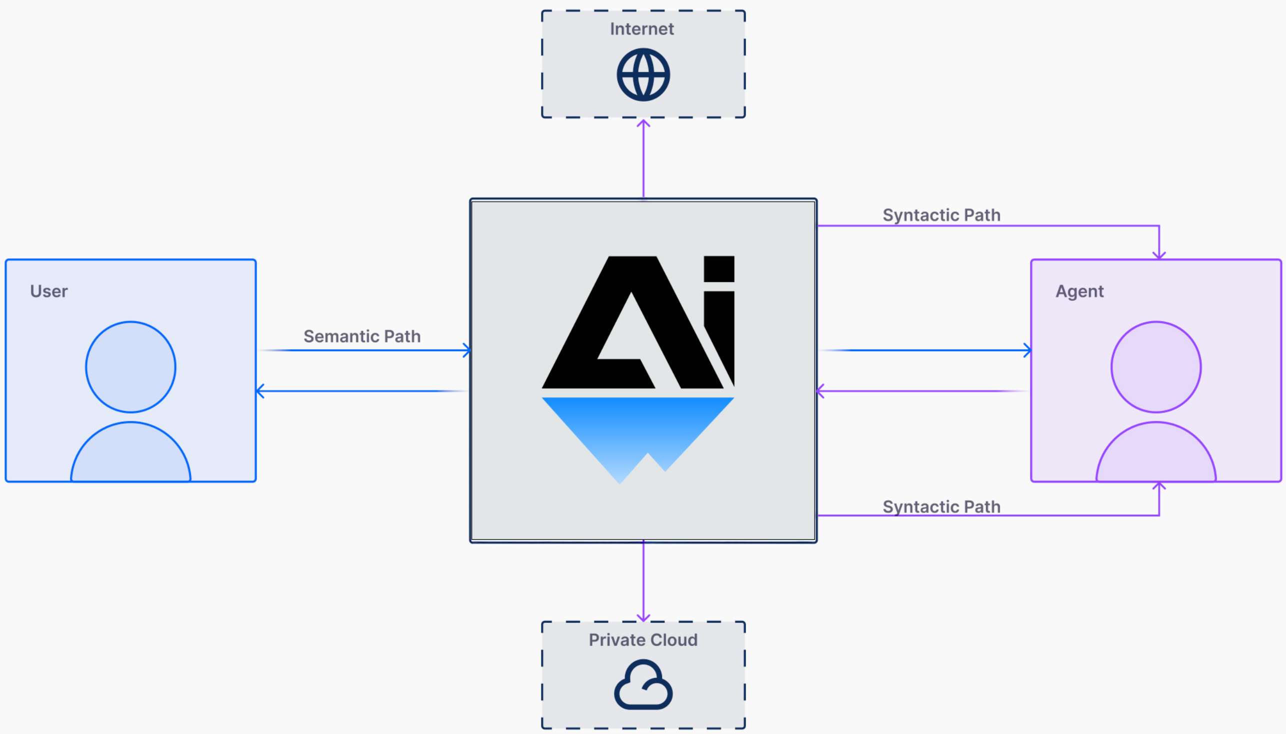


# Figure 2

Figure 2 depicts the main pathways for a user's interaction with an LLM and / or agent. The simplest form of interaction is a user prompt directly sent into an LLM, which produces a corresponding response. Optionally, a user prompt can be enriched by providing context (RAG) to the prompt before feeding into the LLM. When the receiver is an LLM-powered agent, the agent might be carrying out specific actions before being able to provide a response and, finally, the agent might enlist other agents to perform tasks towards its main objective.

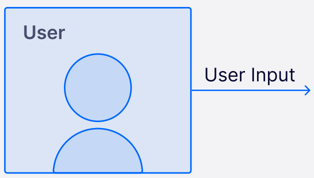
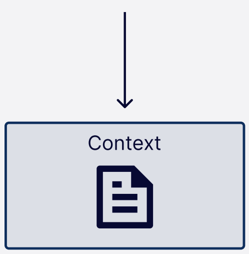
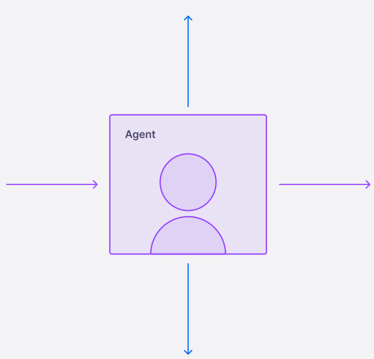
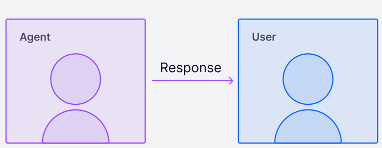
The blue boxes represent individual (specialized) models that extract signals along the pathway that allow for quantification and qualification of the interaction. If we position an AI Security function at the center, we can imagine the pathway consisting of semantic flows (language) and syntactic flows (source code) which, if in alignment, represent a valid and safe transaction.

FIGURE 3



## Figure 3

Given Figure 2) and Figure 3) we can segment an LLM / agentic interaction as following:

| Segment | Scope | Risk Vectors |
|---|---|---|
|  | Emotional context (Sentiment, Tone)<br>Speech (Toxicity)<br>Data exposure (PII, PHI, PCI, data loss...)<br>Relevance (scope)<br>Intention & Objective<br>Code used / Code safety | Unsanctioned speech<br>Data exposure/loss<br>Usage out of scope<br>Unsanctioned objectives<br>Unsafe code |
|  | Source(s)<br>Data Exposure (PII, PHI, PCI, data loss)<br>Relevance / Context Fit | Data exposure/loss<br>Retrieval errors<br>Data Poisoning |
|  | Code safety<br>Network access<br>API usage<br>Profile<br>Actions<br>Semantic and syntactic alignment | Unsafe code<br>Malicious site access<br>Profile attack<br>Unsanctioned actions<br>Inappropriate actions<br>Misalignment |
|  | Unsanctioned responses (toxicity, data exposure...)<br>Context adherence/specificity<br>Code used / Code safety<br>Alignment with intention/objective | Unsanctioned speech<br>Ineffective/wrong use of context<br>Unsafe code<br>Misalignment |

# Conclusion

Utilizing large language models or AI agent-based systems requires a solid and robust cybersecurity posture based on best and common practices, capabilities and tools that, in most cases, are already present within an enterprise whereas each function should be re-evaluated along the data and transactional path of a user to machine interaction. Due to the unexplainable and autonomous nature of AI agents, a dedicated detection function is required that is effective in breaking down such an interaction into segments and signals as well as aligning the user's intention and objective with the actions and corresponding source code created, used, or executed by an agent. Only the combination of an effective cybersecurity posture with dedicated detection capabilities will enable an AI Operator to use generative models and agents in a safe, secure, compliant, and responsible way.