# AI Transparency Notes

## FortiAI on FortiAnalyzer 1.1

**FORTINET DOCUMENT LIBRARY**
https://docs.fortinet.com

**FORTINET VIDEO LIBRARY**
https://video.fortinet.com

**FORTINET BLOG**
https://blog.fortinet.com

**CUSTOMER SERVICE & SUPPORT**
https://support.fortinet.com

**FORTINET TRAINING & CERTIFICATION PROGRAM**
https://www.fortinet.com/training-certification

**FORTINET TRAINING INSTITUTE**
https://training.fortinet.com

**FORTIGUARD LABS**
https://www.fortiguard.com

**END USER LICENSE AGREEMENT**
https://www.fortinet.com/doc/legal/EULA.pdf

**FEEDBACK**
Email: techdoc@fortinet.com

June 2, 2026
FortiAI on FortiAnalyzer 1.1 AI Transparency Notes
05-110-1299421-20260602

# TABLE OF CONTENTS

# Change Log

| Date | Change Description |
| --- | --- |
| 2026-06-02 | Initial release. |
| | |
| | |

# 1. Purpose and Scope

## 1.1 Document Purpose

This AI Transparency Notes document provides a disclosure of the FortiAI feature embedded within FortiAnalyzer. It describes how the AI assistant works, what data it processes, how it is designed, and what safeguards are in place — enabling administrators, compliance teams, and auditors to make informed decisions about its deployment and use. All content in this document is based directly on the FortiAI FortiAnalyzer product specification.

## 1.2 What is FortiAI on FortiAnalyzer?

FortiAnalyzer (FAZ) delivers centralized visibility and analytics across the security infrastructure, providing a solution to handle logs, events, traffic patterns, threat activity, and compliance posture. It aggregates and correlates data from FortiGate devices and the broader Fortinet ecosystem to deliver actionable insights, automated reporting, forensic analysis, and long-term log retention. This enables consistent monitoring, threat detection, and compliance reporting.

While FortiAnalyzer is a robust analytics and reporting platform, navigating large volumes of log data and constructing advanced queries or reports can introduce operational complexity. FortiAI on FortiAnalyzer is an intelligent assistant designed to reduce that complexity by streamlining log analysis, accelerating investigations, simplifying report generation, and enabling natural-language-driven insights. FortiAI maintains strict data handling controls and security first design principles.

FortiAI provides a chat window within the FortiAnalyzer UI through which administrators can ask questions and request tasks in natural language. Answers and results are returned in the same chat window. Specific features can also be triggered directly from the UI via dedicated buttons, for example, creation of a new Event Handler with LLM-assistant configuration.

## 1.3 Features

| Agent / Feature | Description |
|---|---|
| **Log Statistics and Filtering** | Uses natural language queries to generate log-based statistics such as top sources, destinations, applications, users, and threat categories. Automatically applies and refines filters (time range, device, severity, subtype, etc.) to narrow datasets and produce summarized insights without requiring manual query construction. |

| Agent / Feature | Description |
|---|---|
| **Event Summary and Filtering** | Aggregates and summarizes security events across devices and ADOMs, highlighting severity, frequency, trends, and impacted assets. Supports natural language-driven filtering to isolate specific event types, timeframes, threat levels, or affected systems for rapid investigation. |
| **Log Table Visualization** | Dynamically generates graphs based on log tables based from user prompts. Selects relevant contextual filters, sorts and groups data, and presents results in a format to accelerate triage and analysis workflows. |
| **Incident Creation, Update, Report** | Assists with creating new incidents from investigations, auto-populating relevant contextual details. Supports updating incident via adding notes, comments, and refining incident details into downloadable pdf reports. |
| **Event Handler Support** | Provides visibility into configured event handlers, explains trigger logic and conditions, and assists in identifying which handlers correspond to specific alerts. Can guide users in modifying or validating handler configurations through contextual explanations. |
| **FAZ Status Checks** | Uses FAZ CLI operations to report current performance metrics, including storage, log rates, ADOM partitions, etc. Can give further recommendations for improvement with the combined usage of General Knowledge Agent. |
| **General Knowledge Agent** | Answers product questions, compares features and specifications, navigates to specific UI pages within FortiAnalyzer, and locates relevant Fortinet documentation. Serves as a guided entry point for FortiAnalyzer and Fortinet product knowledge. |

## 1.4 Target Audience

| Audience | Relevance |
|---|---|
| **Network Security Administrators** | Primary users; interact with FortiAI daily for device management, policy work, and diagnostics |
| **Security Operations Teams** | Consumers of AI-generated diagnostics, reports, and remediation guidance |
| **IT Compliance and Audit Teams** | Review of AI governance, data handling, and acceptable use boundaries |
| **CISOs / IT Leadership** | Strategic oversight of AI deployment and associated risk posture |
| **Data Protection Officers** | Privacy impact review; data flows, masking practices, and retention policies |

# 2. Model(s) Used and Hosting

## 2.1 Large Language Model

FortiAI on FortiAnalyzer selects between Azure OpenAI GPT-4.1 and GPT-4.1-mini as its large language model (LLM). The selection is pre-determined, based on the specific action or tool executed by the model, to best fit the needs of the feature. Fortinet reserves the right to update the LLM provider.

The LLM is used to: generate text responses; call tools from the FortiAnalyzer server or UI; read and edit data in FortiAnalyzer's database (subject to user permission); display UI components to present results; obtain user confirmation before taking actions; and summarize Fortinet documentation retrieved from the vector database via file search.

## 2.2 System Components

| Component | Role | Hosting Location |
|---|---|---|
| **FortiAI UI** | Collects user queries and tasks. Displays results and answers. Obtains user permission before any data in FortiAnalyzer is updated. | On-premises (FortiAnalyzer appliance) |
| **FortiAI Client (on FortiAnalyzer)** | Assembles system prompts for supplementary features, such as startup suggestions. Provides examples for proper tool call usage, such few-shot prompting. Initializes the execution of tools (to display information or retrieve input from UI). Queries the vector DB for documentation search via API. Performs data masking before sending to LLM and unmasking on receiving the response. Uses the LLM chat completion API. | On-premises (FortiAnalyzer appliance) |
| **FortiAI Server (on FortiAnalyzer)** | Assembles system prompts for main execution loop. Provides tool descriptions to get data, update data and run services on FortiAnalyzer. Hosts and implements the data masking service. Implements authentication — only UI login users or system API users can access it. | On-premises (FortiAnalyzer appliance) |
| **Fortinet AI Proxy** | Fortinet's internal cloud service through which all Fortinet products access the LLM via a centralized hub. Logs all traffic to and from the LLM. Controls AI license and token usage. | Fortinet cloud |

| Component | Role | Hosting Location |
|---|---|---|
| **Vector DB (RAG)** | Contains curated Fortinet documentation used for retrieval-augmented generation. See Section 2.3 for content list. | Fortinet cloud / FortiAnalyzer |
| **Azure OpenAI GPT-4.1/4.1-mini** | The large language model. Performs text generation, tool calling, image transcription, and documentation summarization. | Microsoft Azure (accessed via Fortinet AI Proxy) |

# 2.3 Vector Database Contents (RAG Knowledge Base)

The vector database used for Retrieval-Augmented Generation currently contains the following Fortinet documentation. Content is refreshed with each product release cycle.

- Latest release of the FortiAnalyzer Administrator Guide
- Latest release of the FortiAnalyzer CLI Reference
- Latest FortiGate Product Matrix and Purchase Guide
- Fortinet Community – Knowledge Base Articles Produced by Fortinet Staff

# 2.4 Azure OpenAI Default Safety Policies

Azure OpenAI default guardrail and content control policies are enabled for the GPT-4.1 deployment used by FortiAI. The details of these policies are published by Microsoft and can be found at: Default Guardrails and controls policies for Microsoft Foundry Models (classic)

# 3. Design Methodology, Inputs and Outputs

## 3.1 Design Philosophy

FortiAI on FortiAnalyzer is built around two core goals: (1) to streamline investigation, reporting, and administrative workflows by enabling users to interact with logs, analytics, and system through natural language; and (2) to maintain strict protection of customer data through secure handling and controlled execution of actions.

Within FortiAnalyzer, FortiAI acts as an interactive assistant that helps users navigate the interface, retrieve and interpret log data, generate reports, refine filters, and configure analytics-related functions. It reduces the need to manually traverse multiple menus or construct complex queries, making advanced capabilities more accessible while preserving full user control.

FortiAI operates as a guided assistant rather than an autonomous system. Any operation that changes configuration, creates objects (such as reports or event handlers), or modifies system settings requires explicit user confirmation in the UI before execution. It does not independently alter configurations or data. Additionally, sensitive information is subject to controlled masking and unmasking processes, ensuring that data privacy and integrity are maintained throughout interactions.

## 3.2 Interaction Model

User input may be processed in two ways depending on the task:

- Simple query: user input is sent directly to the LLM once and a response is returned
- Search and execute tool: input is interpreted by the LLM and used to request a tool call be executed, to fetch FortiAnalyzer data, processor data, generate a configuration, trigger an action, or a variety of other actions.

Importantly, some output may also be delivered directly from the tool results to the UI using standard UI components, without passing through the LLM for summarization. This path is used when structured data can be displayed directly without requiring language generation, such as visualized charts or configuration menus.

# 3.3 Inputs

| Input Category | Examples | Sensitivity | Handling Before LLM Transmission |
|---|---|---|---|
| **Natural language queries typed by user** | "Show me the top 5 threats identified in the last 24 hours", "What is the destination IP associated with this threat?" | Low to variable | Not modified. Note: sensitive values typed directly by the user (e.g. a device name or username) may not be masked unless they are already present in session context — see Section 7.2 |
| **FortiAnalyzer Raw Log Data** | SIEM & FortiGate Log data, endpoints, aggregated data, session data, bandwidth statistics | Medium | Retrieved via FortiAI tools. Sensitive fields masked before LLM transmission. |
| **FortiAnalyzer Generated Data** | Incidents and attachments, Alerts, Enriched Indicators, Event Handler Configuration | Medium | Retrieved via FortiAI tools. Sensitive fields masked before LLM transmission. |
| **Current Device Inventory, Topology, Metrics** | Device names, ADOM/VDOM structure, disk usage, log rate, CPU/memory usage | Medium-High | Retrieved via existing FAZ CLI diagnostic tools. Sensitive fields masked before LLM transmission. |
| **User feedback (opt-in)** | Quality rating submitted via the Send Feedback feature | Low | Contains platform name (FortiAnalyzer), version number, user rating, and conversation context immediately prior to submission only. No other user information is collected. |

# 3.4 Outputs

| Output Type | Description | Delivery Method | User Action Required |
|---|---|---|---|
| **Natural language responses** | Plain-text answers, explanations, and summaries generated by the LLM. May use Markdown formatting, such as bolding, italics, and simple tables. | Chat window | None (informational) |

| Output Type | Description | Delivery Method | User Action Required |
|---|---|---|---|
| **Event or Incident Summary Reports** | Formatted text-based documents summarizing activity from the FortiAI chat, retrieved data, and FortiAI-based inferences. | Chat window or PDF download | Review; act as appropriate |
| **Event Hander Configurations** | New Event Handler configuration menus | Chat window + Interactable Component | Explicit user confirmation required |
| **Data Visualization** | Bar and Pie charts based on existing Logview Tables | Chat window + FortiAnalyzer UI | Explicit user confirmation required before applying |
| **UI Filter Application** | FortiAI generates and applies search filters to the current specific FortiAnalyzer UI pages in response to natural language requests. Occurs in Logview, Fortiview, and Event Monitor pages. | FortiAnalyzer UI | None (webpage interaction) |
| **UI navigation** | FortiAI navigates the user to specific FortiAnalyzer UI pages in response to natural language requests | FortiAnalyzer UI | None (navigational) |
| **Direct tool data via UI components** | Structured data from tool responses rendered directly in the UI without LLM summarization | Standard FortiAnalyzer UI components | None; displayed as-is from FortiAnalyzer data |
| **RAG-based documentation answers** | Summaries drawn from the vector database (Fortinet admin guides, best practices, CLI reference, SD-WAN resources) | Chat window | None (informational) |

# 4. Data Flows, Protection, and Retention

## 4.1 Data Flow Overview

When a user interacts with FortiAI on FortiAnalyzer, data flows as follows: it is read from FortiAnalyzer local databases via the FortiAI client and backend API tools such as the JSON-RPC API, then passes through the Fortinet AI Proxy, then to the LLM, then back to the AI Proxy, then to the FortiAI Client, and finally to the FortiAI UI. The AI Proxy may save the LLM chat completion requests and responses, but the data at that point is already masked.

The following table describes each stage in detail:

| # | Stage | Description |
|---|---|---|
| 1 | User Input (UI) | Administrator enters a natural language query or uploads an image in the FortiAI chat window. Alternatively, the user clicks a button in the FortiAnalyzer UI to trigger a specific feature, such as a startup suggestion. (e.g. What are my top threats today? How is my FAZ device performing?). |
| 2 | FortiAI Client — Plan | The FortiAI Client assembles the message payload, specifying the user input, any previous message history, and current chat settings. |
| 3 | FortiAI Client — Masking | The FortiAI Client masking service replaces sensitive fields in the message history and user input (IPs, MAC addresses, passwords, secret keys, certificates, names) with masked tokens if they exist in the key-value map. Note: masking applies to FortiAI tool response data, which populates the key-value map. If the user types new sensitive values directly into the chat, those may not be masked, depending on the type — see Section 7.2. |
| 4 | FortiAI Server — Prompting | The FortiAI backend server assembles the system prompt, provides tool descriptions available to FortiAI. Authentication is enforced: When first accessing the FortiAI GUI, the server will authenticate the user and only allow access for users with a valid GUI session. |

| # | Stage | Description |
|---|---|---|
| 5 | LLM Request via AI Proxy | The masked chat request is transmitted to the Fortinet AI Proxy, which routes it to the relevant model: Azure OpenAI GPT-4.1 or GPT-4.1-mini. All traffic to and from the LLM passes through the AI Proxy. The AI Proxy may log these requests and responses; since data is masked before this point, the logs contain masked data. The AI Proxy employs a second layer of masking to potentially catch missed masked data. |
| 6 | LLM Inference | The Azure model processes the masked request. The response may include generated text, tool call instructions, or structured output. For complex tasks, multiple tool calls may be requested. |
| 7 | FortiAI Server & Client — Receive | The FortiAI server receives the response and forwards it to the client. The Client replaces masked tokens in the LLM response with their original values as the result is passed to the UI. |
| 8 | IF: Tool Call Requested<br>FortiAI Client — Tool Execution | If a tool call is requested, the FortiAI client will use provided parameters, which may require unmasking as it contains obfuscated data (such as a requested filter), and executes the relevant tool using the provided parameters. Some tool call execution may require a follow-up chat request to refine the parameters. For example, building a complex filter may require multiple request/response rounds, repeating steps 2-8. |
| 9 | IF: Tool Call Requested<br>FortiAI Client — Tool Result | The tool result may contain sensitive data, so the FortiAI Client applies a mask again. The tool response is sent to the LLM via steps 2-8. |
| 10 | IF: Tool Completed / No Tool<br>Output to UI | The response is delivered to the FortiAI UI. Output may be: (a) LLM-generated text displayed in the chat window; (b) data from tool results rendered directly via UI components; (c) a downloadable PDF report; or (d) UI navigation to a specific FortiAnalyzer page. |
| 11 | User Confirmation (if data mutation) | If the task requires modifying the FortiAnalyzer database, the UI presents the proposed change and requires explicit user confirmation before writing any data. Without confirmation, no changes are made. FortiAI operations do not perform destructive actions. While it may create data (such as incidents or indicators), delete operations are not available to FortiAI. |

# 4.2 Data Masking and Obfuscation

Before data retrieved from FortiAnalyzer via tool execution is sent to the LLM, the FortiAI Client masking service identifies and replaces sensitive fields with masked tokens. The following field types are masked:

- IP addresses
- MAC addresses
- Passwords
- Secret keys
- Certificates
- Names (device names, user names, and similar identifiers)

The masking service maintains a local key-value mapping of original-to-masked values for the session. After the LLM returns a response, the service replaces all masked tokens with their original values before display to the user.

# 4.3 Data Protection Layers

The following access and data protection controls are implemented in FortiAI on FortiAnalyzer:

| Layer | Control | Description |
|---|---|---|
| 1 | AI License Requirement | Users must hold a valid AI license to use any FortiAI feature. |
| 2 | Feature Toggle | Administrators can hide all FortiAI features entirely via a CLI configuration option, if desired. |
| 3 | FortiAI User Designation | Up to 3 administrators can be designated as FortiAI users. Only those designated administrators can use FortiAI features; other admins cannot access them. |
| 4 | Authentication | To use FortiAI from the UI, a user must have a valid, active FortiAnalyzer GUI login session. Currently, this is the only available access point. |
| 5 | Data Masking (FortiAI Client) | Sensitive fields in tool response data are masked by the FortiAI Client before transmission to the LLM. Data is unmasked from the LLM response before display to the user. |

# 4.4 AI Proxy Logging

All LLM traffic — chat completion requests and responses — passes through the Fortinet AI Proxy with sensitive data masked. The AI Proxy logs all of this traffic. Because data is masked by the FortiAI Client before

transmission to the AI Proxy and LLM, the logs contain masked data rather than original sensitive values. The retention period does not exceed 90 days.

## 4.5 Feedback Data Collection

FortiAI includes an optional "Send Feedback" feature for users to rate the quality and performance of FortiAI responses. Feedback submissions contain only the following: the platform name (FortiAnalyzer), the FortiAnalyzer version number, the user's rating, and the context of the conversation immediately prior to submission. No other user information is collected through this mechanism.

# 5. Testing and Validation

## 5.1 QA Testing

A dedicated QA team tests and validates FortiAI outputs. Testing covers the following: user input, tool execution, LLM inference, data masking and unmasking, and final response quality.

## 5.2 Hallucination Reduction

GPT-4.1 may hallucinate in some outputs. The FortiAI system prompts are specifically designed to limit LLM output to content relevant to FortiAnalyzer and FortiGate. Testing has been conducted to reduce hallucination rates. A persistent disclaimer — "FortiAI can make mistakes" — is displayed in the FortiAI chat window at all times, ensuring users maintain appropriate critical review of AI-generated content.

## 5.3 Azure OpenAI Default Guardrails

Azure OpenAI default guardrail and content control policies are enabled on the GPT-4.1 deployment. These provide a baseline layer of safety filtering at the model level. Full details are published at: Default Guardrails and controls policies for Microsoft Foundry Models (classic)

# 6. Performance Metrics

## 6.1 Status

LLM Performance test on AI Proxy server and LLM has been carried out by FortiAI Proxy team.

FortiAnalyzer AI Accuracy and Safety test has been carried out by QA team.

## 6.2 Metrics to Measure and Report

| Category | Metric | Suggested Measurement Method |
|---|---|---|
| **Accuracy** | Response accuracy rate — percentage of LLM responses judged factually correct for the FortiAnalyzer context | Human expert evaluation on a holdout test set of representative queries |
| **Accuracy** | Hallucination rate — percentage of responses containing FortiAnalyzer facts not grounded in the retrieved context or tool data | Cross-check of AI responses against FortiAnalyzer ground truth state |
| **Safety** | Data mutation confirmation compliance — percentage of write operations that correctly require and receive user confirmation before execution | QA functional test suite covering all mutation-capable agents |
| **Safety** | Masking coverage rate — percentage of defined sensitive field types correctly masked before LLM transmission, across a representative dataset | Automated masking validation using the tool call debug window |
| **Performance** | P50 response latency (end-to-end, UI submission to UI display) | Load testing under representative concurrent query volume |
| **Performance** | P95 response latency | Load testing under representative concurrent query volume |
| **Performance** | P99 response latency | Load testing under representative concurrent query volume |
| **Reliability** | AI Proxy / LLM service availability (uptime percentage) | AI Proxy monitoring and incident log |

| Category | Metric | Suggested Measurement Method |
|---|---|---|
| **Reliability** | Graceful degradation — FortiAnalyzer core functionality operates correctly when FortiAI service is unavailable | Failure injection testing (AI Proxy unreachable scenarios) |
| **User Quality** | User feedback rating distribution (from in-product Send Feedback feature) | Aggregated from AI Proxy or FortiAnalyzer feedback data |
| **Token Usage** | Average prompt token count per query type; average completion token count; total token usage per period | AI Proxy token usage logs |

# 7. Known Limitations

## 7.1 LLM Hallucination

GPT-4.1 may hallucinate — producing plausible but incorrect answers. While FortiAI system prompts are designed to constrain LLM output to FortiAnalyzer topics, and testing has been conducted to reduce this, hallucination cannot be fully eliminated. A disclaimer ("FortiAI can make mistakes") is permanently displayed in the chat window. Users should always validate AI-generated scripts, policy recommendations, and diagnostic conclusions before acting on them. For any action that modifies FortiAnalyzer data, a user confirmation step is required — but this does not substitute for careful human review of the proposed change.

## 7.2 User-Typed Sensitive Data — Partial Masking Gap

The data masking service operates on data returned from FortiAnalyzer via a backend service. If a user types sensitive values directly into the chat window — for example, a device name or an IP address that is not already present in the current session context — those values may not be masked before being sent to the LLM. Users should avoid typing sensitive credentials, keys, or other high-sensitivity values directly into the FortiAI chat window. However, any sensitive data that are already known to current AI Chat session, then those sensitive data will be masked automatically from user input.

## 7.3 Scope of Data Submitted to the LLM

FortiAI may submit data collected from FortiAnalyzer to the LLM. This data may include device configurations, policy packages, and monitoring data from managed devices. While defined sensitive fields within this data are masked (Section 4.2), the overall volume and breadth of configuration data transmitted to the LLM is by design — the AI requires this context to answer management questions accurately. Administrators should factor this into their assessment when deploying FortiAI in highly sensitive or restricted environments.

## 7.4 Masking Field Coverage

The masking service covers the field types identified as sensitive by the product team: IP addresses, MAC addresses, passwords, secret keys, certificates, and names. There may be additional fields that specific

customers or environments consider sensitive which are not currently included in the masking scope. Requests for expanded masking coverage can be submitted through Fortinet support and will be evaluated for future releases.

## 7.5 No Session Memory across Conversations

FortiAI does not retain context between separate user sessions. Each new session begins without knowledge of prior conversations. Users must re-provide relevant context when starting a new session.

## 7.6 FortiAI Feature Availability Depends on AI Service Connectivity

FortiAI features depend on connectivity to the Fortinet AI Proxy and Azure OpenAI. If these services are unavailable, FortiAI features will not function. FortiAnalyzer's core functionality is unaffected by AI service interruptions — FortiAI is an enhancement layer and is not required for core operations.

## 7.7 MCP Server — Not Yet Available

The FortiAnalyzer MCP Server has not yet been released for internal FortiAnalyzer use or for public third-party access. Fortinet reserves the right to release it in a future version. When released, it will enforce the same RBAC-based permission checks as the current internal implementation.

# 8. Risk Analysis and Categorization

FortiAI on FortiAnalyzer is best categorized as a non-high-risk AI system under Regulation (EU) 2024/1689. Based on its documented intended purpose, FortiAI assists authorized administrators with network and security management tasks, including diagnostics, policy drafting, script generation, configuration explanation, and documentation retrieval. It is not intended to make, recommend, or materially influence decisions in any Annex III high-risk domain, including employment, education, law enforcement, access to essential services, migration, critical infrastructure safety decisioning, or the administration of justice.

FortiAI is also not designed or intended for any prohibited AI practice under Article 5, such as manipulative, exploitative, social scoring, certain biometric, or impermissible law-enforcement uses. Although FortiAI is not currently classified as high-risk, it is designed to support applicable transparency obligations where users interact with an AI system or receive AI-generated outputs. The AI Act transparency framework includes disclosure obligations for certain AI interactions and AI-generated content. As an additional safeguard, FortiAI does not autonomously execute proposed configuration changes; any such change requires review and affirmative confirmation by an authorized administrator before implementation.

# Appendix A: Component Summary

| Component | Role | Hosting Location |
|---|---|---|
| **FortiAI UI** | Collects user queries and tasks. Displays results and answers. Obtains user permission before any data in FortiAnalyzer is updated. | On-premises (FortiAnalyzer appliance) |
| **FortiAI Client (on FortiAnalyzer)** | Assembles system prompts for supplementary features, such as startup suggestions. Provides examples for proper tool call usage, such few-shot prompting. Initializes the execution of tools (to display information or retrieve input from UI). Queries the vector DB for documentation search via API. Performs data masking before sending to LLM and unmasking on receiving the response. Uses the LLM chat completion API. | On-premises (FortiAnalyzer appliance) |
| **FortiAI Server (on FortiAnalyzer)** | Assembles system prompts for main execution loop. Provides tool descriptions to get data, update data and run services on FortiAnalyzer. Hosts and implements the data masking service. Implements authentication — only UI login users or system API users can access it. | On-premises (FortiAnalyzer appliance) |
| **AI Proxy** | Fortinet's internal cloud service through which all Fortinet products access the LLM via a centralized hub. Logs all traffic to and from the LLM. Controls AI license and token usage. | Fortinet internal cloud |
| **Vector DB (RAG)** | Contains curated Fortinet documentation used for retrieval-augmented generation. See Section 2.3 for content list. | Fortinet cloud / FortiAnalyzer |
| **Azure OpenAI GPT-4.1/4.1-mini** | The large language model. Performs text generation, tool calling, image transcription, and documentation summarization. | Microsoft Azure (accessed via Fortinet AI Proxy) |

# Appendix B: Glossary

| Term | Definition |
|---|---|
| **ADOM** | Administrative Domain — a logical partition in FortiAnalyzer for managing a subset of devices, logs, or administrative access. |
| **AI Proxy** | Fortinet's internal cloud service that routes LLM traffic for Fortinet products, logs all LLM interactions (in masked form), and controls AI licensing and token usage. |
| **FortiAI / FortiAI Agent** | The branded name for FortiAI's generative AI capabilities embedded within FortiAnalyzer. |
| **GPT-4.1 & GPT-4.1-mini** | The Azure OpenAI large language model currently used by FortiAI on FortiAnalyzer. |
| **Hallucination** | A phenomenon where an LLM generates plausible-sounding but factually incorrect content. |
| **MCP (Model Context Protocol)** | A protocol enabling standardized communication between AI models and tool/data providers. |
| **MCP Server** | A server that implements the MCP standard. Exposes structured tools, data sources, and operational context to AI assistants through a standardized interface. It enables standardized interaction between LLM systems and platforms (such as FortiAnalyzer), allowing the model to retrieve data, execute actions, and maintain contextual awareness while enforcing authentication, authorization, and policy controls. |
| **RAG** | Retrieval-Augmented Generation — an AI architecture that retrieves relevant reference documents and includes them as context before generating a response. |

www.fortinet.com

Copyright© 2026 Fortinet, Inc. All rights reserved. Fortinet®, FortiGate®, FortiCare® and FortiGuard®, and certain other marks are registered trademarks of Fortinet, Inc., and other Fortinet names herein may also be registered and/or common law trademarks of Fortinet. All other product or company names may be trademarks of their respective owners. Performance and other metrics contained herein were attained in internal lab tests under ideal conditions, and actual performance and other results may vary. Network variables, different network environments and other conditions may affect performance results. Nothing herein represents any binding commitment by Fortinet, and Fortinet disclaims all warranties, whether express or implied, except to the extent Fortinet enters a binding written contract, signed by Fortinet's Chief Legal Officer, with a purchaser that expressly warrants that the identified product will perform according to certain expressly-identified performance metrics and, in such event, only the specific performance metrics expressly identified in such binding written contract shall be binding on Fortinet. For absolute clarity, any such warranty will be limited to performance in the same ideal conditions as in Fortinet's internal lab tests. Fortinet disclaims in full any covenants, representations, and guarantees pursuant hereto, whether express or implied. Fortinet reserves the right to change, modify, transfer, or otherwise revise this publication without notice, and the most current version of the publication shall be applicable.