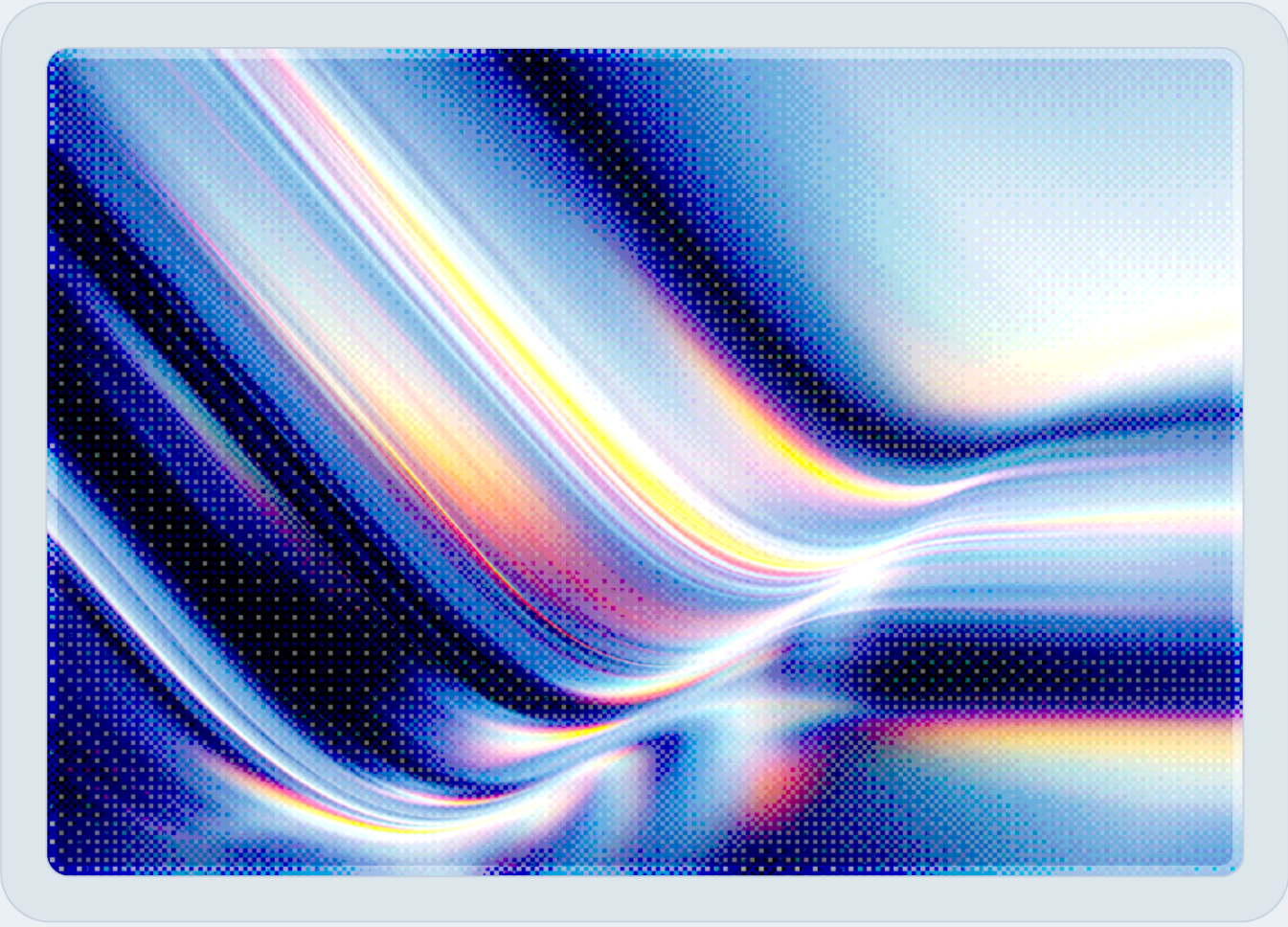
CSAI Foundation | Cloud Security Alliance

Exposed AI Infrastructure: Self-Hosted Services Under Attack

31% of Surveyed Ollama Servers Respond to Unauthenticated Requests as Active Threat Campaigns Systematically Map the Attack Surface

2026-05-07

AI-assisted Rapid Research



CSAI

CSA cloud security alliance®

Key Takeaways

- A large-scale internet scan of approximately one million exposed AI services found that 31% of queried Ollama API servers responded to unauthenticated test prompts — representing 1,652 of more than 5,200 instances directly accessible without any access control [1].
- Beyond Ollama, more than 12,000 Open WebUI instances, 2,650 Flowise deployments, and 300 Langflow servers were found exposed; a subset of each lacked authentication entirely, with some revealing full agentic workflows, enterprise credential stores, and plaintext API keys [1].
- GreyNoise honeypot infrastructure captured 91,403 attack sessions targeting exposed LLM endpoints between October 2025 and January 2026, identifying two distinct threat campaigns conducting systematic infrastructure reconnaissance at scale [3][9].
- Two critical and high-severity vulnerabilities — CVE-2026-7482 ("Bleeding Llama," CVSS 9.1) in Ollama and CVE-2025-64496 (CVSS 7.3) in Open WebUI — compound the exposure risk through distinct mechanisms: the former enables unauthenticated data exfiltration from any reachable instance; the latter targets authenticated users through a social-engineering vector [4][5].
- Organizations adopting self-hosted AI infrastructure must treat network exposure boundaries and authentication requirements as first-order security controls, not afterthoughts, before deploying any AI inference or orchestration service.

Background

The rapid proliferation of self-hosted AI infrastructure reflects a broader organizational shift: as open-weight models like Llama 3, Mistral, and DeepSeek become capable enough for production use, many organizations are deploying inference servers, chat interfaces, and agentic orchestration platforms on their own hardware and cloud instances. The appeal is real — self-hosted deployments offer data privacy, cost predictability, and freedom from vendor rate limits. However, the survey findings described in this note suggest that the pace of adoption has substantially outrun the security practices required to operate these systems safely — a pattern consistent across multiple platforms and deployment contexts.

Ollama, one of the most widely adopted frameworks for running large language models locally and on self-hosted servers, does not enable authentication by default. Open WebUI, which provides a web-based chat interface layered atop Ollama and other backends, has accumulated a significant install base as the de facto front end for self-hosted deployments. Workflow orchestration platforms such as n8n and Flowise extend these capabilities further, wiring LLM inference into enterprise systems, databases, and third-party APIs. Langflow serves a similar function for Python-centric agentic pipelines. Each of these tools emerged from developer-centric communities where ease of setup was prioritized over hardened defaults — a design philosophy appropriate for local experimentation but deeply problematic when instances are reachable from the public internet.

Security researchers at Intruder identified this gap through a large-scale internet survey conducted in early 2026 [1]. Using certificate transparency logs and OSINT techniques targeting AI-adjacent domain naming conventions, they assembled a dataset of approximately one million exposed AI service endpoints and performed systematic analysis of authentication posture, vulnerability exposure, and active data leakage. Their findings documented high rates of misconfiguration and unauthenticated exposure across a broad cross-section of the self-hosted AI ecosystem [1] — a pattern that GreyNoise honeypot data confirms is already being targeted by organized threat actors [3].

Security Analysis

The Scale of Unauthenticated Exposure

Of the 5,200-plus Ollama API servers directly queried by Intruder researchers, 31% — approximately 1,652 instances — responded to unauthenticated test prompts with full inference capability [1][2]. These servers were not merely reachable; they were willing to process arbitrary requests from any source on the internet. The implications extend well beyond unauthorized compute consumption. An unauthenticated caller to an exposed Ollama API can enumerate installed models, submit arbitrary system and user prompts, extract information about connected tool integrations, and in some configurations interact with agentic capabilities that have write access to connected systems.

Of particular concern, researchers identified 518 frontier models across the surveyed servers that were acting as unauthorized proxies to paid frontier model APIs — including services from Anthropic, OpenAI, Google, and others [1]. Organizations or individuals whose API keys were embedded in these proxy configurations may be unaware they are bearing the financial cost and, more critically, the risk of credential exposure to unauthenticated third parties. Discovery of such a configuration by a financially

motivated attacker would allow them to run up substantial API charges, exfiltrate proprietary prompt engineering work embedded in system prompts, and potentially pivot to other services using the same credential material.

The Flowise findings extend this picture further. Of the 2,650-plus exposed Flowise instances identified, 92 disclosed complete agentic workflows, including prompt templates, external integrations, and authentication credentials for connected services [1]. Flowise deployments are commonly configured to orchestrate access to databases, CRM systems, vector stores, and third-party APIs [1] — meaning a fully exposed instance may carry blast-radius consequences comparable to a compromised service account in the connected systems. The business logic embedded in these workflows often represents significant intellectual property in addition to the security risk posed by the credential exposure.

Active Threat Campaigns Confirm Attacker Interest

The Intruder research provides a snapshot of misconfiguration at scale, but GreyNoise's longitudinal honeypot data demonstrates that attackers have moved beyond passive discovery to systematic, targeted probing of this attack surface [3]. Between October 2025 and January 2026, GreyNoise's infrastructure captured 91,403 sessions targeting Ollama LLM endpoints across two operationally distinct campaigns [3][9].

The first campaign, running throughout the October-to-January window, exploited server-side request forgery vulnerabilities in Ollama's model-pull functionality and Twilio SMS webhook integrations. Attackers used ProjectDiscovery's OAST infrastructure to validate callback connections, and a single JA4H signature appeared in 99% of attack traffic, indicating shared automated tooling — most likely the Nuclei vulnerability scanner configured with a custom template. The campaign involved 62 source IP addresses across 27 countries, consistent with distributed reconnaissance infrastructure.

The second campaign showed markedly different operational characteristics. Beginning December 28, 2025, two IP addresses — 45.88.186.70 (AS210558, 1337 Services GmbH) and 204.76.203.125 (AS51396, Pfcloud UG) — launched a methodical eleven-day probe of 73-plus LLM model endpoints, generating 80,469 enumeration sessions [3]. The queries were deliberately benign ("hi," "How many states are there in the United States?") but targeted OpenAI-compatible and Google Gemini API formats across every major model family, including GPT-4o, Claude Sonnet, Opus, and Haiku, Llama 3.x, DeepSeek-R1, Gemini, Mistral, Qwen, and Grok. Both IPs carry extensive histories of CVE exploitation across more than 200 vulnerabilities, and infrastructure overlap suggests the enumeration output feeds a larger downstream exploitation pipeline. As GreyNoise's researchers noted: eighty thousand enumeration requests represent deliberate investment — threat actors do not map infrastructure at this scale without plans to use that map [3].

Compounding Vulnerabilities in Core Platforms

Two recently disclosed vulnerabilities compound the exposure risk through distinct mechanisms. CVE-2026-7482 exploits Ollama's lack of default authentication, enabling data exfiltration from any publicly reachable instance. CVE-2025-64496 targets authenticated Open WebUI users through a social-engineering vector, allowing an attacker who controls a reachable server to achieve account takeover and, in some configurations, remote code execution.

CVE-2026-7482, dubbed "Bleeding Llama" by researchers at Cyera, is a CVSS 9.1 critical vulnerability in Ollama's model quantization pipeline [4]. The flaw is an out-of-bounds heap read triggered when a user uploads a crafted GGUF model file. A malicious file can declare a tensor size substantially larger than the actual data provided, causing Ollama to read well beyond the intended buffer boundary and into adjacent heap memory. That adjacent memory may contain system prompts, user message history, environment variables, API keys, and other sensitive runtime data. Exploitation requires only three unauthenticated API calls: uploading the crafted file via the blob endpoint, triggering model creation, and pushing the resulting memory dump to an attacker-controlled server. Since Ollama launches without authentication by default, the roughly 300,000 internet-facing Ollama instances identified by SecurityWeek represent a substantial attack surface for this vulnerability [6].

CVE-2025-64496, discovered by Cato Networks researchers, affects Open WebUI versions 0.6.34 and earlier and carries a CVSS score of 7.3 [5]. The vulnerability resides in the Direct Connections feature, which allows users to link Open WebUI to external OpenAI-compatible model servers. An attacker who controls a server reachable by the victim can send a crafted server-sent event of type "execute," which Open WebUI evaluates using the `new Function()` constructor without sanitization. This allows arbitrary JavaScript execution in the victim's browser context, where it can read the authentication token from localStorage and transmit it to attacker infrastructure. For users with the `workspace.tools` permission, a second attack path enables backend remote code execution through an unsandboxed Python `exec()` call. The patch, available in v0.6.35, blocks execution of untrusted "execute" events.

The n8n orchestration platform has faced its own escalating vulnerability series. CVE-2025-68613, a critical remote code execution flaw in n8n's expression evaluation engine, allows an authenticated attacker to execute arbitrary code by submitting malicious workflow expressions [7]. A subsequent disclosure in February 2026 revealed that the December fix was bypassed, compounding the urgency for organizations running self-hosted n8n instances [8]. Researchers warned that a compromised n8n environment grants access to all credential stores configured within it — potentially including Salesforce, AWS, OpenAI, and other cloud services that form the backbone of an organization's service integrations.

Systemic Root Causes

The research reveals that the problem is not simply a collection of individual misconfigurations. Several structural patterns appear repeatedly across the affected platforms. Many of these tools ship with authentication disabled by default, placing the entire burden of security on operators who may not appreciate the exposure risk when connecting a service to a cloud instance or running it behind a misconfigured reverse proxy. Container deployment practices compound the issue: hardcoded credentials in docker-compose files, services running as root, and missing network segmentation between AI inference layers and connected enterprise systems all appear frequently in the survey data [1].

There is also a temporal dimension to the vulnerability problem that organizations are not tracking adequately. Researchers found that one analyzed tool had more than 90% of its internet-exposed instances running with serious known vulnerabilities — and that within a week of a new remote code execution vulnerability being disclosed, zero instances had applied the patch [1]. This evidence suggests that self-hosted AI infrastructure is not being patched at the cadence the current threat environment demands.

Recommendations

Immediate Actions

Organizations should audit their external attack surface for any AI inference or orchestration services that are reachable from the public internet. This includes Ollama, Open WebUI, Flowise, Langflow, n8n, and any locally deployed LLM API wrappers. Services that do not require public internet access — the appropriate configuration for most internal production deployments — should be restricted to internal networks or VPN-gated access immediately. Services requiring external access should enforce authentication at the network boundary before any inference request is processed. Port 11434 (Ollama's default) and the web ports of UI frameworks should be reviewed in firewall and cloud security group configurations without delay.

For any instance of Ollama running on an internet-accessible host, operators should apply the patch addressing CVE-2026-7482 and verify that the service is bound to localhost or a private network interface rather than 0.0.0.0. Open WebUI deployments should be upgraded to v0.6.35 or later, and the Direct Connections feature should be disabled unless specifically required. N8n instances should be patched to the current release and assessed for credential exposure resulting from prior CVE-2025-68613 exploitation.

Short-Term Mitigations

Where direct network restriction is not immediately feasible, organizations should place an authenticated reverse proxy in front of AI inference services. API key or OAuth-based authentication should be enforced at the proxy layer, with requests logged for anomaly detection. No self-hosted LLM API should be reachable without authentication under any operational posture.

Credential rotation is an urgent priority for any organization that has operated Ollama or n8n instances without network controls. API keys for Anthropic, OpenAI, AWS, Salesforce, and other services configured in these environments should be treated as potentially compromised and rotated proactively, even in the absence of confirmed exploitation evidence. The frontier models identified as proxying paid frontier model APIs represent a concrete financial and data integrity risk that warrants immediate investigation [1].

Organizations should also implement continuous attack surface monitoring for AI infrastructure. Static network controls alone may not keep pace with the rate at which development and data science teams deploy new AI tooling — continuous external exposure assessment provides complementary detection coverage. Automated discovery of newly exposed AI services, integrated into a continuous external exposure assessment program, substantially reduces the window in which misconfigurations remain undetected.

Strategic Considerations

The self-hosted AI security problem is, at its foundation, a shared responsibility problem that is not being correctly mapped. Platforms like Ollama, Open WebUI, and Flowise ship with configurations optimized for local developer use — a reasonable default for their origin context, but one that becomes a systemic hazard when instances are routinely deployed in cloud environments without the network controls appropriate for production infrastructure. CSA encourages platform maintainers to consider authentication-enabled defaults, mandatory acknowledgment prompts when services are bound to non-localhost interfaces, and stronger documentation of the security implications of default configurations.

For enterprises standardizing on self-hosted AI, a formal procurement and deployment standard should be established that treats AI inference infrastructure with the same rigor applied to databases and internal APIs. This means documented network segmentation requirements, mandatory authentication controls, defined patch SLAs tied to CVE severity, and inclusion of AI infrastructure in routine vulnerability scanning programs. The GreyNoise campaign data demonstrates that threat actors have already operationalized the process of finding and enumerating these services at scale [3]; defenders cannot afford to treat self-hosted AI as lower-risk than equivalent internal infrastructure simply because it is newer.

CSA Resource Alignment

The vulnerabilities and patterns described in this note map directly to several areas of CSA's AI security framework guidance and should be understood in that context.

CSA's MAESTRO framework for agentic AI threat modeling addresses the threat class represented by exposed Flowise and n8n instances: agentic systems that have broad tool access and external integration points represent a significantly expanded blast radius compared to standalone inference services. The systemic finding that many exposed instances provide unauthenticated access to agentic capabilities — including code execution and file write access — corresponds to MAESTRO's Layer 4 (Tool and Integration Security) and Layer 5 (Application Interface Security) threat categories, where insufficient access control on tool-calling interfaces enables lateral movement into connected enterprise systems.

The AI Controls Matrix (AICM) v1.0 provides the most directly applicable control framework for the remediation guidance in this note. AICM's AI Infrastructure Security domain addresses network isolation requirements for AI serving components, and the Shared Security Responsibility Model (SSRM) for AI clarifies that infrastructure-level controls — including network exposure and authentication — are the responsibility of the organization deploying the AI system, not the model provider or framework developer. Organizations seeking a structured assessment approach should use the AICM's Application Provider (AP) implementation guidelines to evaluate their current AI infrastructure posture.

CSA's Zero Trust guidance reinforces the principle that self-hosted AI inference services should never be implicitly trusted based on network location alone. The pattern of services exposed on cloud instances — where an operator assumes internal network protection that a misconfigured security group or container network setting has silently removed — is a canonical example of implicit trust leading to breach. Zero Trust architecture applied to AI infrastructure requires explicit authentication at every service boundary and verification of caller identity and authorization scope before any inference request is processed.

The STAR for AI program provides a mechanism for organizations deploying AI infrastructure to self-assess and communicate their security posture against AICM controls, including those directly relevant to network exposure and access management. Given the scale of the misconfiguration problem documented in this survey, CSA encourages AI infrastructure operators to use the STAR for AI assessment process as a structured way to identify and remediate exposure gaps before they are discovered by adversaries who are already systematically scanning for them.

References

[1] Intruder. "From Enterprise Chatbots to Gooner Caves: Exposed AI Infrastructure Is Rampant." Intruder Research, May 2026.

[2] The Hacker News. "We Scanned 1 Million Exposed AI Services. Here's How Bad the Security Actually Is." The Hacker News, May 2026.

[3] GreyNoise Intelligence. "Threat Actors Actively Targeting LLMs." GreyNoise Blog, January 2026.

[4] Cyera Research. "Bleeding Llama: A Critical Memory Leak in the World's Most Popular Local AI Platform." Cyera Blog, 2026.

[5] Cato Networks CTRL. "Vulnerability Discovered in Open WebUI Enables Account Takeover and Remote Code Execution (CVE-2025-64496)." Cato Networks Blog, 2025.

[6] SecurityWeek. "Critical Bug Could Expose 300,000 Ollama Deployments to Information Theft." SecurityWeek, 2026.

[7] SOCRadar. "CVE-2025-68613: Critical RCE Vulnerability Disclosed in n8n Workflow Automation." SOCRadar, 2025.

[8] The Register. "n8n's Latest Critical Flaws Bypass December Fix." The Register, February 2026.

[9] Dark Reading. "2 Separate Campaigns Probe Corporate LLMs for Secrets." Dark Reading, January 2026.