CSAI Foundation

Cloud Security Alliance AI Safety Initiative

Marimo Pre-Auth RCE: AI Toolchain Credentials at Risk

CVE-2026-39987 Exploited Within 10 Hours of Disclosure

Unofficial AI-assisted Research

2026-04-12

Key Takeaways

- **CVE-2026-39987** is a pre-authentication remote code execution vulnerability (CVSS v4.0: 9.3 / Critical) in Marimo, a widely used reactive Python notebook platform, classified under CWE-306 (Missing Authentication for Critical Function) [1][9].
- Active exploitation was observed by Sysdig's Threat Research Team within 9 hours and 41 minutes of public advisory disclosure, with a credential theft sequence completed in under 3 minutes — and without any publicly available proof-of-concept code [1][2].
- The vulnerability is rooted in an authentication bypass on Marimo's integrated terminal WebSocket endpoint (`/terminal/ws`), which provides full interactive shell access to any unauthenticated remote attacker [3].
- Because Marimo environments are often configured with API keys for commercial LLM providers (OpenAI, Anthropic, Google), cloud provider credentials, and access to training datasets, a successful exploit carries outsized supply chain risk to the victim organization's broader AI infrastructure [2][4].
- CISA added CVE-2026-39987 to its Known Exploited Vulnerabilities (KEV) catalog on April 10, 2026, with a mandatory remediation deadline of April 11, 2026 for U.S. Federal agencies [5].
- **Immediate action required**: Upgrade all Marimo instances to version 0.23.0 or later. Organizations unable to patch immediately must place Marimo behind an authenticated reverse proxy or restrict network access to trusted IP ranges.

Background

Marimo is a reactive, open-source Python notebook designed as a next-generation alternative to Jupyter. Notebooks are stored as pure Python files, execute cells in a deterministic dependency order, and eliminate the hidden-state problems that often plague conventional notebook environments [6][7]. Since its introduction, Marimo has accumulated approximately 19,600 GitHub stars at the time of disclosure and established a strong following among data scientists, ML engineers, and AI researchers, supported in part by its native integrations with commercial AI providers including OpenAI, Anthropic, Google, and Ollama [3][8]. The platform can generate notebook cells using an embedded AI assistant with full context about the variables in memory, can be deployed as an interactive web application, and supports SQL cells against external databases, positioning it as an integration hub for modern AI development workflows.

That centrality is precisely what makes CVE-2026-39987 consequential. A vulnerability in a standalone text editor or a generic web framework would expose whatever data happened to be present. A vulnerability in an AI development notebook exposes the full credential estate that an organization's machine-learning team has assembled: API keys to commercial LLM providers, cloud provider access credentials, SSH keys, database connection strings, and, often, references to proprietary model artifacts and training datasets. When those credentials are exfiltrated, the organizational exposure extends well beyond the compromised machine.

The vulnerability was publicly disclosed on April 8, 2026, accompanied by a GitHub Security Advisory (GHSA-2679-6MX9-H9XC) [1][9]. The disclosure identified all Marimo versions through 0.20.4 as affected and announced that a fix had been incorporated into version 0.23.0. Within 9 hours and 41 minutes of that announcement, the vulnerability had already been exploited in the wild.

Security Analysis

The Vulnerability: Missing Authentication on a Critical WebSocket Endpoint

The root cause of CVE-2026-39987 is an inconsistency in authentication enforcement across Marimo's server-side WebSocket endpoints. Marimo exposes multiple WebSocket routes to support its interactive features. The primary notebook synchronization endpoint (`/ws`) correctly enforces authentication by calling `validate_auth()` before accepting a connection. The terminal endpoint (`/terminal/ws`), which provisions a full interactive PTY shell, does not. Instead, it checks only whether the server is running in an appropriate mode and whether the host platform supports terminal features, then accepts the connection unconditionally [3][4].

The asymmetry suggests that the terminal feature was added or significantly extended at a later stage of development without full integration into the authentication framework that governed the notebook's other endpoints. The result is that any actor capable of reaching the Marimo server's network port — including an unauthenticated remote attacker — can open a WebSocket connection to `/terminal/ws` and receive a fully interactive shell running under the same operating system user account as the Marimo process. There are no credentials to guess and no tokens to forge; the connection is accepted as a matter of course.

From the shell, an attacker has access to the entire filesystem visible to the Marimo process, including environment files, SSH keys, configuration directories, and any secrets passed through environment variables — a configuration pattern common in containerized AI development environments. The CWE-306 classification (Missing Authentication for Critical Function) accurately describes the failure mode: the function in question, a remote interactive terminal, is among the most sensitive capabilities a server can expose, yet it was protected by no authentication gate at all [3].

Exploitation in the Wild: Speed, Stealth, and No PoC Required

The exploitation timeline documented by Sysdig's Threat Research Team illustrates three notable characteristics [1][2][10][11][12]. First, the attacker achieved a working exploit with no public proof-of-concept code — the advisory description alone was sufficient to reconstruct the attack path, suggesting that the vulnerability required no specialized knowledge beyond the ability to open a WebSocket connection to a documented endpoint. Second, the first confirmed exploitation attempt occurred within 9 hours and 41 minutes of the advisory being published. Third, once inside the Marimo shell, the attacker completed a credential harvesting sequence — inspecting the environment, reading `.env` files, and searching for SSH keys — in under 3 minutes.

This combination of factors illustrates an important pattern in the contemporary threat landscape. The 9-hour-41-minute window between advisory publication and confirmed exploitation demonstrates that attacker monitoring capabilities can outpace organizational patch cycles, particularly for vulnerabilities requiring no specialized exploit development. When the vulnerability in question requires only a single unauthenticated network request to achieve code execution, the asymmetry between attacker and defender becomes particularly acute.

The observed attacker behavior — manual reconnaissance, targeted `.env` enumeration, SSH key harvesting — is consistent with a credential collection operation rather than destructive interference. The attacker appeared primarily interested in extracting material that could be used for subsequent access: LLM provider API keys that could be monetized by selling AI compute access under the victim's account, cloud provider credentials that could support lateral movement into storage or compute infrastructure, and SSH keys that could facilitate access to additional systems [2][4].

Supply Chain Implications for AI Infrastructure

The credential theft profile of this exploit carries supply chain implications that extend beyond the immediate compromised host. Organizations running Marimo as part of an AI development pipeline often configure it with credentials that span a wide attack surface. A single exfiltrated OpenAI or Anthropic API key may expose not only billing abuse but also an adversary's ability to observe prompt structures, retrieve conversation metadata, and infer proprietary development strategies from usage patterns. Cloud provider credentials may enable access to object storage containing training datasets, fine-tuned model weights, evaluation benchmarks, and inference pipeline configurations — assets that represent months of engineering investment and may contain sensitive data from production systems.

This lateral reach is materially different from a credential theft in a more bounded context. When an attacker obtains LLM API keys, they are not merely gaining access to a service account; they may be gaining persistent read access into an organization's AI development strategy, competitive research, and, in regulated industries, data that was processed through the AI pipeline. CSA's AI Controls Matrix (AICM) explicitly identifies this class of risk under its supply chain and credential management domains, recognizing that AI-native development tooling creates new credential surfaces that require the same protection as production infrastructure.

Affected Versions and Patch Status

CVE-2026-39987 affects all Marimo releases through and including version 0.20.4. The vulnerability was remediated in version 0.23.0, which applies proper authentication enforcement to the `/terminal/ws` endpoint [1][3][9]. The gap between the last vulnerable version (0.20.4) and the first fixed version (0.23.0) reflects a series of intervening releases; organizations pinned to any version in that range remain vulnerable unless they have implemented compensating network controls.

Recommendations

Immediate Actions

The primary remediation is straightforward: **upgrade to Marimo 0.23.0 or later immediately**. This is the only fix that addresses the root cause and should be treated as the definitive remediation. For organizations that maintain Marimo via `pip`, the upgrade is a single command. For containerized deployments, the base image or requirements file must be updated and redeployed. CISA's April 11, 2026 KEV deadline has passed for federal agencies, but all other organizations operating Marimo in any network-accessible configuration should treat this upgrade with the same urgency [5].

For organizations that cannot immediately upgrade — for example, due to dependency conflicts or deployment approval cycles — the following compensating controls materially reduce exploitability: placing Marimo behind a reverse proxy that enforces authentication before any WebSocket connections are permitted; restricting network access to the Marimo port to a list of known, trusted IP addresses via firewall or security group rules; and auditing environment variables, `.env` files, and SSH keys on any host that has run a vulnerable Marimo version to determine whether credential exfiltration may have already occurred.

Organizations should also rotate any API keys or credentials that were present in environments running Marimo ≤0.20.4 as a precautionary measure, particularly LLM provider API keys and cloud provider access credentials. The Sysdig-documented attack sequence targeted these assets first, and they may have been compromised in systems that did not generate host-level intrusion alerts.

Short-Term Mitigations

Beyond the immediate patch, organizations should reassess how AI development tools are exposed on their networks. Marimo is frequently run locally during development but may also be deployed on shared servers, cloud instances, or containerized infrastructure accessible to broader network segments. A network audit to identify all Marimo instances — including those deployed as part of CI/CD pipelines, experiment tracking workflows, or interactive demo environments — should precede any assumption that the vulnerability has been fully addressed.

Log analysis for the period surrounding April 8, 2026 should examine whether any connections to the `/terminal/ws` endpoint were made by IP addresses outside the expected developer population. The attack described by Sysdig involved human-paced manual reconnaissance; in environments with logging sufficient to capture WebSocket connection events, the activity should be detectable. Organizations without WebSocket-level logging should implement it as a baseline capability for any server running interactive notebook infrastructure.

Security teams should also evaluate whether Marimo credentials were stored in environment variables, secrets managers, or `.env` files on the affected hosts, and trace the downstream access that those credentials would have permitted. In environments where cloud credentials were present, CloudTrail, GCP Audit Logs, or equivalent provider logs should be reviewed for anomalous access to storage, compute, or API gateway resources in the window following disclosure.

Strategic Considerations

CVE-2026-39987 exemplifies a broader pattern: as AI development tooling matures and expands, it is accumulating security complexity that was not present in earlier generations of developer tooling. Reactive notebooks, AI code assistants, MCP servers, and local model-serving endpoints all represent network-accessible surfaces that sit in developer environments rather than production infrastructure, and are therefore less likely to be covered by the hardening controls, monitoring, and patch management processes that govern production systems. An emerging body of security research in 2026 has documented this growing toolchain attack surface, and this incident reinforces that AI development tooling itself must be treated as a security boundary.

Organizations building AI capabilities should inventory the network-accessible services running in developer and research environments with the same rigor applied to production systems. Tools that integrate with commercial LLM providers or cloud infrastructure — as Marimo explicitly does — should be subject to the credential management practices appropriate for services with production access, including secrets rotation schedules, access logging, and regular vulnerability scanning.

CSA Resource Alignment

This vulnerability and its associated risks map directly to several Cloud Security Alliance frameworks and guidance documents.

AICM (AI Controls Matrix v1.0) addresses supply chain security controls for AI Application Providers and Orchestrated Service Providers, including controls for credential management, developer toolchain security, and access control for AI-integrated environments. CVE-2026-39987 illustrates the consequences of failing to apply production-grade authentication controls to AI development infrastructure, which AICM identifies as a shared responsibility surface requiring explicit ownership.

MAESTRO (Agentic AI Threat Modeling) provides a framework for identifying threat vectors in AI agent architectures. The exploitation pattern observed in this incident — where credential theft from a developer notebook enables lateral movement into AI provider accounts and cloud infrastructure — aligns with MAESTRO's treatment of AI agent credential risks, specifically the threat scenario in which access to an agent's underlying credentials cascades into broader organizational compromise.

CSA Cloud Controls Matrix (CCM) control domains relevant to this incident include IAM-01 through IAM-09 (Identity and Access Management), with particular emphasis on the requirement that all management interfaces — including developer tooling with administrative-level access to organizational credentials — enforce authentication and maintain audit logs. The CVE-2026-39987 failure mode is a direct violation of the IAM principle that authentication must be enforced at every programmatic entry point.

CSA Zero Trust Guidance reinforces that network location should never confer implicit trust. Marimo instances running on internal networks or developer VPNs are not inherently protected; the `/terminal/ws` authentication gap is exploitable from any network position that can reach the

Marimo server port. The Zero Trust principle of "never trust, always verify" applies as directly to developer tooling as to production API endpoints.

CSA guidance on organizational AI responsibilities identifies the obligation for organizations deploying AI development infrastructure to maintain patch currency, conduct regular toolchain security assessments, and ensure that credentials used by AI development tools are subject to the same rotation and monitoring requirements as production credentials.

References

[1] The Hacker News. "Marimo RCE Flaw CVE-2026-39987 Exploited Within 10 Hours of Disclosure." The Hacker News, April 10, 2026.

[2] Sysdig Threat Research Team. "Marimo OSS Python Notebook RCE: From Disclosure to Exploitation in Under 10 Hours." Sysdig, April 2026.

[3] Endor Labs. "Root in One Request: Marimo's Critical Pre-Auth RCE (CVE-2026-39987)." Endor Labs Blog, April 2026.

[4] Vulert. "Marimo RCE Flaw CVE-2026-39987 Exploited Within 10 Hours of Disclosure." Vulert Blog, April 2026.

[5] CISA. "Known Exploited Vulnerabilities Catalog." Cybersecurity and Infrastructure Security Agency, accessed April 12, 2026.

[6] Marimo. "marimo — a next-generation Python notebook." marimo.io, accessed April 2026.

[7] Marimo. "marimo Documentation." docs.marimo.io, accessed April 2026.

[8] marimo-team. "marimo: A reactive notebook for Python." GitHub, accessed April 2026.

[9] marimo-team. "Releases · marimo-team/marimo." GitHub, accessed April 2026.

[10] Security Affairs. "CVE-2026-39987: Marimo RCE exploited in hours after disclosure." Security Affairs, April 2026.

[11] SecurityWeek. "Critical Marimo Flaw Exploited Hours After Public Disclosure." SecurityWeek, April 2026.

[12] LufSec. "Critical CVE-2026-39987 Exploited in Marimo Python Notebook Within Hours of Disclosure." LufSec Blog, April 2026.