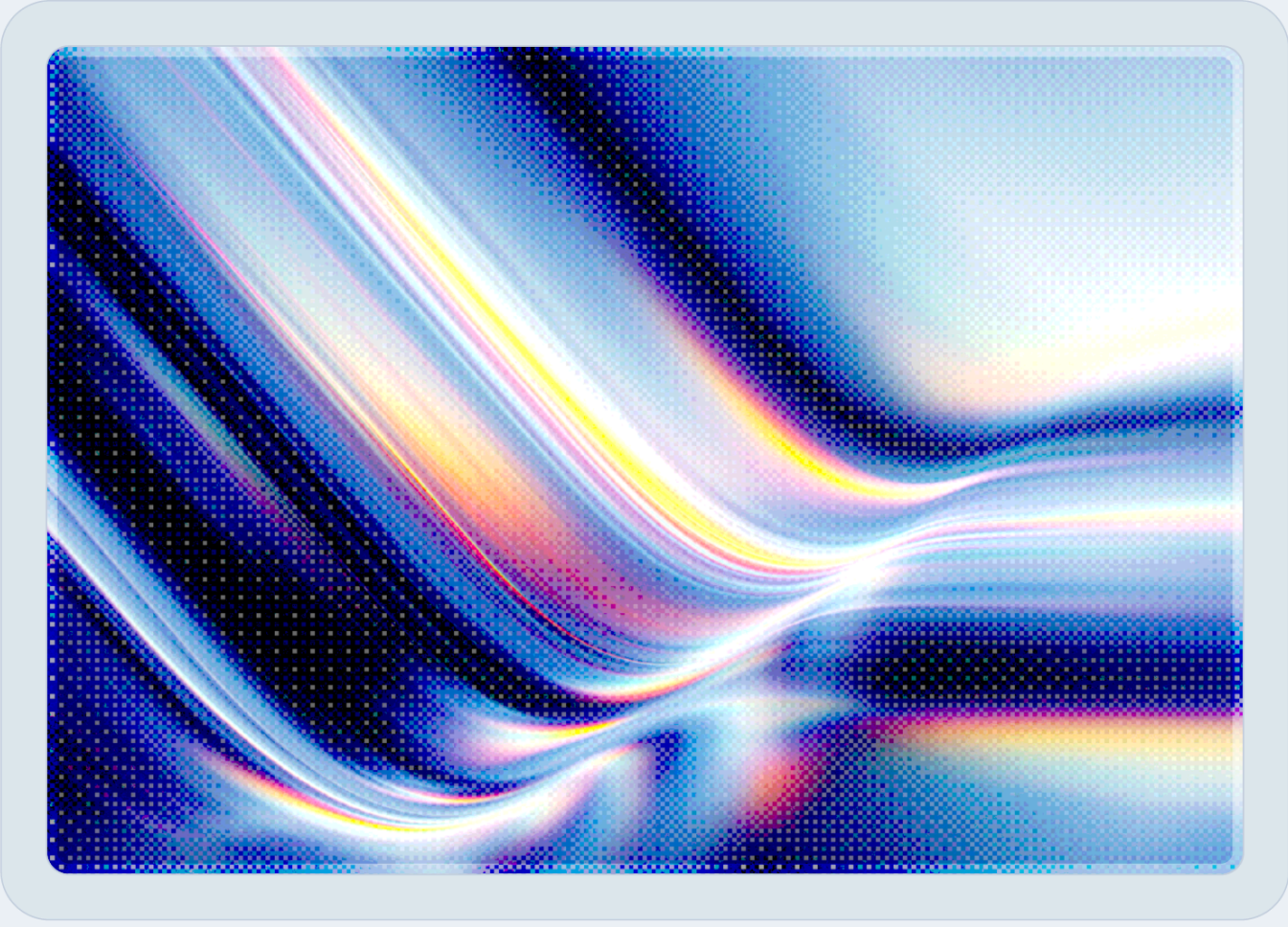
CSAI Foundation | Cloud Security Alliance

Azure API Management Classic Exploitation: When the AI Gateway Leaks the Keys

Security Implications and Guidance for APIM Classic Secret Exposure (Advisory MZK9-GH0)

2026-11-24

AI-assisted Rapid Research



CSAI

CSA cloud security alliance®

© 2026 Cloud Security Alliance. Some rights reserved.

You may download, store, display, view, print, redistribute, and link to this document in its original, unmodified form, provided that attribution to the Cloud Security Alliance is maintained and all trademark and copyright notices remain intact.

This document may not be modified or altered. You may quote portions of the document as permitted by the Fair Use provisions of the United States Copyright Act, provided that attribution is given to the Cloud Security Alliance.

This document may be shared on professional and social media platforms in its original form with attribution.

This document was generated with AI assistance and has not undergone official CSA review and approval processes.

Key Takeaways

An attacker exploited a flaw in the Classic tiers of Azure API Management (APIM) to reach management functions and read the secrets those instances store. Microsoft says it identified the issue on November 18, 2026 and closed the access path on November 19. It notified customers through a private Azure Service Health advisory, Tracking ID MZK9-GH0 [1][2]. Norway's National Security Authority (NSM) has linked the activity to attacks on the Norwegian energy sector, and The Register reports that NSM named Equinor and Skagerak Energi among the affected organizations. NSM says an unknown actor "has been actively looking for access tokens, secrets, and named values in APIM" [2][3][4].

The fix does not undo theft that has already happened. Microsoft states that the mitigation "does not automatically invalidate credentials that may have been accessed before the fix was deployed" [1]. NSM's position is that anyone using the service must assume their secrets and named values are compromised and rotate them [3][5]. Rotation has to happen at the system that issued each credential, because changing the stored value inside APIM does not revoke the old credential at its issuer [6][7].

The AI angle matters. Microsoft positions APIM as its AI gateway for Azure OpenAI and other model endpoints. Its documentation describes storing model provider API keys as APIM named values, which is one of the secret types reported as exposed [8][9][12]. This note infers that organizations which put model provider keys in Classic instances without Key Vault references should treat those keys as stolen. Public reporting has not confirmed that AI workloads specifically were targeted.

As of this writing, secondary vulnerability trackers report that no CVE or CVSS score has been assigned [10][11], and Microsoft has not published a public security bulletin for the issue. Defenders who rely solely on CVE feeds or CVE-keyed scanner findings are therefore unlikely to see this incident.

Background

What Azure API Management Holds

Azure API Management is Microsoft's managed API gateway. It sits between API consumers and backend services, where it enforces authentication, rate limits, transformation policies, and observability. To do this, the gateway has to hold credentials for nearly everything behind it. Named values are key-value pairs,

often marked as secrets, that policies reference to inject backend credentials into requests. Subscription keys authenticate API consumers, while identity provider and OpenID Connect client secrets support the developer portal and OAuth flows. Named values can be stored directly in APIM or as references to Azure Key Vault secrets, and Microsoft documents the Key Vault option as the more secure one [8].

Microsoft also markets APIM as an AI gateway for Azure OpenAI and other model endpoints. Its AI gateway capabilities add token-based rate limiting, token metrics, semantic caching, and load balancing across model backends [9]. Microsoft's guidance on authenticating to AI APIs documents two options. The gateway can authenticate to the model backend with a managed identity, or it can hold the backend API key as a named value and inject it through policy, for example with `set-backend-service` or `set-header` [12]. The API-key option takes fewer configuration steps than managed identity. Deployments that chose it placed model provider keys, with their billing exposure and access to prompt and completion data, in the same store the attacker was searching.

Incident Timeline

The public timeline has been assembled from Microsoft's private advisory as reported by the press, from NSM's public advisory, and from the affected companies' statements. Several details remain unsettled. Equinor disclosed an ongoing cyberattack on November 17, one day before the date on which Microsoft says it identified the issue [1]. Microsoft's advisory also states that customers were exposed before Microsoft identified the problem [2]. The table below sets out the sequence of events as the cited sources report it.

Date (2026)	Event	Source
November 17	Equinor discloses an ongoing cyberattack on certain IT systems	[1]
November 18	Microsoft identifies the APIM Classic issue	[2]
November 19	Microsoft closes the management-plane access path and rotates Microsoft-controlled keys where appropriate	[1][2]
November 20	NSM publishes an advisory linking the flaw to attacks on the Norwegian energy sector	[3][4]

Date (2026)	Event	Source
November 21	Microsoft publishes public guidance on rotating credentials in APIM	[2][6]
November 23	NSM updates its advisory (editorial correction); BleepingComputer and The Register report private advisory MZK9-GH0	[1][3] [4]

Skagerak Energi has also reported an attempted cyberattack [1], but the cited sources do not give a date for it, so it is not placed on the timeline.

Affected Scope

According to Microsoft's statement to BleepingComputer, the affected tiers are the Classic Developer, Basic, Standard, and Premium tiers. The Classic Consumption tier and all v2 tiers were not affected [1]. Microsoft describes exposure as limited to Classic instances "not configured according to documented security best practices" [2]. Its advisory, as paraphrased by Help Net Security, names two risky configurations: instances without the recommended network integration, and instances that store secrets directly in APIM rather than referencing Azure Key Vault [5]. That framing deserves a caveat. In the Classic tiers, virtual network injection is available only on Developer and Premium, so customers on Classic Basic and Standard could not have applied the network-integration portion of this guidance without changing tiers. Among the named victims, Equinor says it stopped the attack and found no indication that data was stolen or altered, and Skagerak reported no operational impact [1].

Security Analysis

The Access Path: A Legacy Management Plane

Microsoft has not published a root cause, and the most detailed public reconstruction comes from a secondary analyst rather than from Microsoft. ThreatAft points to the legacy direct management API that each Classic instance exposes at `<instance>.management.azure-api.net` [10]. That interface authenticates with shared access signature tokens signed by the instance's tenant access key rather than through Microsoft Entra ID and Azure RBAC. ThreatAft reports that its calls are not recorded in the Azure Resource Manager Activity Log. Microsoft's own retirement notice confirms that the direct

management API was retired on March 15, 2025 and advises customers to disable it if it is still enabled [18]. ThreatAft goes further and says retirement ended support without removing the endpoint [10], a claim Microsoft has not confirmed.

A Reddit administrator described by BleepingComputer gives a similar account. As BleepingComputer paraphrases it, attackers exploited a zero-day authentication bypass in a deprecated legacy API to obtain a privileged system account, and the activity generated no logs customers could see [1]. BleepingComputer did not verify that account, and Microsoft has not confirmed it, so it should be treated as plausible rather than established. The fact that "tokens used to call direct management APIs" appear on Microsoft's own list of exposed secret types [2] lends it real circumstantial support. ThreatAft also recommends disabling the API as a precaution [10], although a recommendation of that kind is not itself evidence of the access path.

If this reconstruction is correct, the incident shows how a legacy interface that survives a platform migration can keep its own authentication model outside the identity controls customers actually monitor. Organizations that had moved to Entra ID–governed access and watched their Activity Logs would, under this reconstruction, have seen nothing there. ThreatAft warns accordingly that quiet APIM logs are "fully consistent" with successful theft, and that the evidence is more likely to appear downstream [10].

Why the Gateway Is the Prize

API gateways concentrate credentials by design, and that concentration is what makes this incident severe. Microsoft lists the potentially exposed material as secrets stored in named values, client secrets for identity providers and OpenID Connect providers, subscription keys, and tokens for the direct management API [1][2]. ThreatAft's rotation guidance also covers backend credentials and tenant access keys [10]. Kristiansen, an independent researcher quoted by The Register, observes that a holder of a stolen credential "may be able to use it directly against the service it protects, outside APIM" [4]. The attacker therefore does not need the gateway once the secrets are out. A database connection string, a SaaS API token, or a model provider key can be used from any network location the backend accepts until it is revoked. Backends restricted to private endpoints or allowlisted addresses limit that exposure but do not eliminate it.

The table below summarizes each exposed secret class, what an attacker could do with it, and where it must be rotated. The rotation targets follow Microsoft's published guidance [6] and NSM's advice to rotate all tokens and secrets [3].

Exposed secret class	Plausible attacker use	Where rotation must occur
Secret named values (backend keys, connection strings, model provider API keys)	Direct access to backend APIs, databases, SaaS, or AI model endpoints, bypassing APIM policies	The issuing backend or provider; then update APIM or the Key Vault reference
Identity provider and OIDC client secrets	Impersonate the APIM application to the identity provider; abuse developer portal sign-in flows	Microsoft Entra ID, Azure AD B2C, or the external IdP application registration
Subscription keys	Call published APIs as a legitimate consumer, including metered AI endpoints	APIM (regenerate primary and secondary keys); the subscription ID does not change
Direct management API tokens and tenant access keys	Further management-plane operations if the instance remains reachable	APIM (regenerate tenant access keys; disable the direct management API)
Credentials embedded directly in policy XML	Same as named values, but harder to inventory	The issuing system; then remove from policy text

The AI Gateway Dimension

Public reporting has not tied this campaign to AI workloads, and this note does not claim that it was. The exposure is still structurally significant for AI programs. Microsoft's Learn documentation describes storing an Azure OpenAI key "as a secret" named value and presents Key Vault references as an optional extra protection [12]. That is exactly the configuration Microsoft's advisory identifies as risky [5]. A stolen model provider key gives an attacker inference billed to the victim and outside the gateway's token limits and policies. Where provider-side logging or stored completions are enabled, it may also give access to stored prompts, files, or fine-tuning artifacts tied to that deployment.

Stolen APIM subscription keys for AI APIs carry a related risk. They let an attacker spend a team's token quota through the gateway's own policies, and that traffic can be hard to tell apart from legitimate use. NSM itself noted that the rapid growth of AI use affects the threat picture and increases the likelihood that other actors will try to exploit the weakness [3][4].

The incident also repeats a lesson CSA has documented across 2026 for open-source AI gateways such as LiteLLM [14][15][16]. The component built to keep provider keys away from applications becomes the single place where all of those keys can be stolen at once. The managed-service version of the problem differs in one important way: customers cannot patch the gateway themselves, and they depend on the provider both to fix the flaw and to tell them about it.

Disclosure and Detection Gaps

Two features of the disclosure will slow many organizations' response. First, there is no CVE identifier or public Microsoft Security Response Center bulletin, and notification went through a private Azure Service Health advisory [10][11]. This is a departure from Microsoft's own stated practice. In June 2024, the Microsoft Security Response Center committed to issuing CVEs for critical cloud-service vulnerabilities even when customers do not need to take action [19]. Organizations that track exposure through CVE feeds, vulnerability scanners, or the CISA Known Exploited Vulnerabilities catalog are therefore unlikely to see this event, and teams that do not route Service Health security advisories to their security operations staff may have missed it as well. ThreatAft separately warns that CVE-2026-35425, a different APIM flaw from July 2026, is unrelated and should not be used as a reference for this incident [10].

Second, the sources disagree on whether customers need to take action. Microsoft told BleepingComputer that the fix for Microsoft-hosted API Management instances was fully deployed and required no customer action [1]. NSM, however, advises checking whether each instance's gateway runtime version is at or above 0.52.31936 or 0.53.31938, and contacting Microsoft if it is not [3][5]. Both positions can hold at once, since patching may need no action while secret rotation still does. Organizations should not read "no customer action required" as meaning there is no exposure to clean up.

Recommendations

Immediate Actions

Start with a complete inventory of APIM instances across every subscription and tenant, including development, test, and orphaned instances, which customers interviewed by The Register say may widen exposure [4]. For each Classic instance, read the `RuntimeVersion` value from the gateway's `/internal-status-0123456789abcdef` endpoint. Confirm that it meets NSM's thresholds

(0.52.x at build 31936 or higher; 0.53.x at build 31938 or higher), and escalate to Microsoft if it does not [3][13]. Next, disable the legacy direct management REST API on every instance and regenerate the tenant access keys [10][18].

Then assume every secret held by an affected Classic instance is compromised and rotate it at its source. Create the new credential at the backend or identity provider, update APIM or the Key Vault secret, confirm that traffic works, and only then revoke the old credential [6][7]. Updating a value inside APIM does not revoke the credential at its issuer; the old credential remains valid until it is revoked at the source [6][7]. Key Vault–backed named values may take up to about four hours to refresh unless refreshed manually, so revocation should wait until the new value has been picked up [8][10]. Give model provider keys for Azure OpenAI, Microsoft Foundry deployments, and third-party providers the same priority as database and identity provider secrets, and regenerate subscription keys for any AI-facing APIs. Credentials hard-coded into policy XML are the easiest to miss, so search policy text for them as well. Any confirmed compromise in Norway should be reported to NSM [5].

Short-Term Mitigations

Because the suspected access path bypassed the logs customers normally rely on, detection should focus on downstream use of stolen credentials rather than on APIM itself. ThreatAft recommends checking first which log sources were enabled and how long they are retained. It then suggests reviewing `AzureActivity` for APIM secret-listing and write operations, such as `namedValues/listValue` and `subscriptions/listSecrets` [10]. These Azure Resource Manager–level queries detect secret access through the ARM path, for example by a compromised Entra identity. They will not show access through the legacy direct management API, so a clean result does not rule out exposure. Further hunts cover the developer portal audit log for unexpected sign-ups or identity changes, new subscription IDs and requests with tracing enabled, and `AADServicePrincipalSignInLogs` for applications whose secrets APIM held signing in from new IP addresses [10].

For AI backends, review Azure OpenAI and other provider usage and billing for unfamiliar source addresses, unusual deployments, or token consumption that does not match gateway metrics. Because the start of exposure is unknown, this review should cover the full available log retention period before November 19 and continue afterward for any credentials not yet rotated. Review Key Vault audit logs as well for access patterns that do not match normal gateway refresh behavior, which is the anomaly the Reddit administrator described [1].

In parallel, move every secret named value to a Key Vault reference with versionless identifiers. Replace stored model API keys with managed identity authentication to Azure OpenAI and other Entra ID–integrated backends, an approach Microsoft's AI gateway guidance presents as removing the need for API keys [9][12]. Where the tier supports it, restrict network access to the management plane in line with Microsoft's network integration guidance. On Classic Basic and Standard, where that integration is unavailable, prioritize Key Vault references and migration to a v2 tier. If the developer portal is unused, unpublish it or disable self-service sign-up [10].

Strategic Considerations

In the longer term, this incident argues for treating AI gateways, managed or self-hosted, as tier-zero credential infrastructure with the same controls applied to identity providers and secrets vaults. Organizations should plan migration off APIM Classic tiers where practical. They should also confirm that no workload depends on retired management interfaces, because retirement without verified removal leaves an attack surface nobody watches. Credentials kept behind gateways should be short-lived, workload-identity-based, or both, so that a gateway breach yields little that can be reused.

Cloud customers should also change how they take in provider security notices. Microsoft committed in 2024 to issuing CVEs for critical cloud-service vulnerabilities even where no customer action is required [19], yet no CVE has been assigned to this incident as of this writing. Security teams should not rely on CVE feeds alone for managed-service exposure, and should route Service Health security advisories into incident response with the same urgency as CVE-based alerts. Procurement teams, for their part, should seek contractual commitments on timely, specific disclosure of management-plane vulnerabilities.

CSA Resource Alignment

CSA's prior research on AI gateway compromise addresses the same underlying risk, although it concerns self-hosted, open-source software rather than a managed service. *LiteLLM AI Gateway: Critical Vulnerability Chain Exposes API Keys* [14] describes how a centralized AI proxy becomes a high-value credential store whose compromise exposes every provider key it brokers. The APIM Classic exposure is the managed-cloud version of the same failure, and that paper's recommendation to rotate all provider keys the gateway holds applies directly here.

After TeamPCP: Patched LiteLLM Gateways Remain Pivot Points [15] makes the argument that matters most for responders this week: a patched gateway is not a remediated one. Credentials stolen before a fix stay valid until they are revoked at the source. Microsoft's own statement that its mitigation does not invalidate previously accessed credentials [1] confirms that this lesson applies equally to hyperscaler-

managed gateways. *LiteLLM's Default Admin Key: A Cloud Credential Vault* [16] adds the configuration-hygiene side. Gateways that keep long-lived secrets locally, rather than in a dedicated vault behind workload identity, turn one weakness into broad credential exposure. That mirrors Microsoft's finding that instances storing secrets outside Key Vault were the ones at risk [5].

For control mapping, the CSA AI Controls Matrix (AICM) v1.1 [17] provides the relevant requirements. Its Cryptography, Encryption and Key Management domain covers secret storage and rotation, and the Identity and Access Management domain covers privileged management-plane access and workload identity. The Logging and Monitoring domain covers detection of downstream credential misuse, while the Threat and Vulnerability Management domain covers tracking provider advisories that arrive without a CVE. Organizations can use AICM v1.1 to check whether their AI gateway deployments, whatever the vendor, meet these expectations, and to set requirements for cloud providers through STAR for AI assessments.

References

[1] BleepingComputer. "Microsoft Azure flaw exploited to steal customers' API credentials." BleepingComputer, November 23, 2026.

[2] The Hacker News. "Microsoft Fixes Azure API Management Classic Flaw as Actor Targets Customer Secrets." The Hacker News, November 24, 2026.

[3] Nasjonal sikkerhetsmyndighet (NSM). "NSM deler tiltak mot sårbarhet i Azure API Management." NSM, November 20, 2026 (updated November 23, 2026).

[4] The Register. "Norway links energy firm cyberattacks to critical Azure API Management flaw." The Register, November 23, 2026.

[5] Help Net Security. "Azure API Management vulnerability exploited to steal customers' secrets." Help Net Security, November 24, 2026.

[6] Microsoft. "Rotating credentials in Azure API Management." Microsoft Tech Community, November 21, 2026.

[7] Incidenture. "Azure API Management flaw exploited: how attackers went after APIM Classic secrets." Incidenture, November 24, 2026.

[8] Microsoft. "Use named values in Azure API Management policies." Microsoft Learn, accessed November 24, 2026.

[9] Microsoft. "AI gateway capabilities in Azure API Management." Microsoft Learn, accessed November 24, 2026.

[10] ThreatAft. "Azure APIM Flaw: No CVE, Secrets Out, Equinor Hit." ThreatAft, November 24, 2026.

[11] Carthage Electronics. "November 23 2026 CVE Threat Brief." Carthage Electronics, November 23, 2026.

[12] Microsoft. "Authenticate and authorize access to LLM APIs using Azure API Management." Microsoft Learn, accessed November 24, 2026.

[13] ITdaily. "Attackers steal API credentials via flaw in Azure API Management Classic." ITdaily, November 24, 2026.

[14] Cloud Security Alliance. "LiteLLM AI Gateway: Critical Vulnerability Chain Exposes API Keys." CSA AI Safety Initiative, June 2026.

[15] Cloud Security Alliance. "After TeamPCP: Patched LiteLLM Gateways Remain Pivot Points." CSA AI Safety Initiative, August 2026.

[16] Cloud Security Alliance. "LiteLLM's Default Admin Key: A Cloud Credential Vault." CSA AI Safety Initiative, September 2026.

[17] Cloud Security Alliance. "AI Controls Matrix (AICM) v1.1." Cloud Security Alliance, 2026.

[18] Microsoft. "Direct management API retirement (March 2025)." Microsoft Learn, accessed November 24, 2026.

[19] Microsoft Security Response Center. "Toward greater transparency: Unveiling Cloud Service CVEs." MSRC Blog, June 27, 2024.