contact@namiq.io · namiq.io

NamiQ

SECURITY ADVISORY

Microsoft Azure App Service for Linux: new CVE

CVE-2026-109484 · record published 24 November 2026

Analysis Report

Teams on Azure App Service for Linux should confirm with Microsoft that the fix for this 10/10 flaw has been applied to their environment; the record's affected builds run from 110.0.0.0 up to, but ...

PREPARED BY	NamiQ	DATE	24 November 2026
FACTS AS OF	24 November 2026	VERSION	1.0
SEVERITY	**Critical: CVSS 3.1 10.0**	STATUS	Not on CISA's exploited list; exploitation not recorded

NamiQ © 2026 NamiQ. All rights reserved. This document is the property of NamiQ. It may be shared unaltered, with attribution.
Companion article: namiq.io/insights/microsoft-azure-app-service-for-linux-cve-2026-109484

The bottom line

> *Teams on Azure App Service for Linux should confirm with Microsoft that the fix for this 10/10 flaw has been applied to their environment; the record's affected builds run from 110.0.0.0 up to, but not including, 110.0.7.78 and 111.0.7.3. Contact Azure support and ask for the current platform build and written confirmation.*

At a glance

Fact	Detail
Product	Microsoft Azure App Service for Linux (CVE-2026-109484)
Severity	Critical: 10 (CVSS 3.1)
Status	A new CVE record, published by Microsoft on 24 November 2026. Not on CISA's list of exploited vulnerabilities when we read it on 24 November 2026, which means nothing public says yet that anyone is using it.
Fixed in	The record's affected range ends just before 110.0.7.78 and 111.0.7.3, suggesting those are the first unaffected builds; no unaffected releases are listed and no vendor advisory could be read.
Who looks first	Teams hosting workloads on Azure App Service for Linux where the platform build falls in the range 110.0.0.0 up to, but not including, 110.0.7.78 or 111.0.7.3, per the record.

Does this apply to you?

Answer in order. Stop at the first no.

Question	If no	If yes
Do you host any workload on Azure App Service for Linux?	Nothing to do for this CVE.	Find out which platform version your App Service environment is on (next question).
Is the platform build from 110.0.0.0 up to, not including, 110.0.7.78 or 111.0.7.3?	You are outside the stated affected range; no further action needed.	Confirm with Microsoft support that the fix has been applied to your environment.
Can untrusted users or the public internet reach this App Service endpoint?	Exposure is reduced; still confirm the fix with Microsoft given the score.	Raise with Microsoft support urgently and ask for the patched platform build number in writing.

What it is

A flaw in how Azure App Service for Linux handles certain input lets a network attacker with no login raise their privileges on the platform. The record describes this as special element injection (CWE-75) and scores it 10 out of 10.

Why it matters

A score of 10 is the ceiling; the record adds no login and network reachable, meaning an attacker who can send a request to the service needs no prior foothold. CWE-75 means crafted input crosses a boundary between privilege levels that should be enforced. Our reading is that the combination — no authentication, full network path, maximum score — means any internet-facing App Service for Linux on an affected build should be treated as exposed until Microsoft confirms otherwise. Because this is managed infrastructure, the fix is Microsoft's to apply; the customer's part is confirming it happened.

Things people will get wrong

- Microsoft manages the platform, so the fix will be applied without us doing anything. Microsoft applies platform updates, but customers should confirm the patch is in place for their specific environment, given the maximum CVSS score and the no-authentication requirement.
- The version numbers look like internal build identifiers; this probably does not apply to us. The record names Azure App Service for Linux and states those build ranges explicitly; it does not explain what the numbers refer to, but they are the ranges the record gives.
- There is no CISA listing, so we can wait until the next normal patch window. A score of 10 with no authentication required over a network justifies confirming the fix sooner than a routine patch cycle, regardless of CISA's listing status.

How we would triage it

This is our judgement.

Step 1: Find which App Service for Linux environments you run

- List all App Service resources across your Azure subscriptions and identify those on a Linux plan; the affected range is 110.0.0.0 up to, but not including, 110.0.7.78 and 111.0.7.3.
- The platform build version referenced in the record may not be directly visible in the portal; you may need to raise a support case to obtain it.

Step 2: Confirm with Microsoft that the fix has been applied

- Because Azure App Service is managed infrastructure, the platform patch is applied by Microsoft; open a support ticket and ask for the host platform build now running on your App Service for Linux environments.
- Ask for written confirmation with the build number and the date it was applied.

Step 3: Restrict inbound access while awaiting confirmation

- If your App Service for Linux is reachable from the internet and the platform version is unconfirmed, use App Service access restrictions to limit inbound traffic to known IP ranges until Microsoft confirms the fix.
- Remove the restriction once Microsoft provides written confirmation of the patched build.

Step 4: Record the outcome

- Once Microsoft confirms the fix, record the platform build number, the confirmation date, and the support ticket reference in your asset register.
- Update the asset record for each App Service so the same question can be answered faster for future CVEs in this product.

To close it, read the version from the product itself, not from an old inventory, and save the output with the date. If you cannot update today, write down who owns it, the new date, and what limits exposure until then.

Send this to your IT provider

Copy it, change the names and dates, and send it. It is a draft for you to check, not a template tested with any provider.

> *We are following up on CVE-2026-109484, a vulnerability in Microsoft Azure App Service for Linux with a CVSS score of 10. The record states that an attacker with network access and no login can elevate privileges. The affected platform build range is 110.0.0.0 up to, but not including, 110.0.7.78 and 111.0.7.3. Could you please confirm: (1) whether we run Azure App Service for Linux, and if so, in which subscriptions; (2) which platform build version those environments are on; (3) whether Microsoft has applied the fix, and what the patched build number is. We would appreciate written confirmation of the fix status and build version by [date]. Thank you.*

The weakness behind it

The record classes this as CWE-75.

Likely ATT&CK mapping

MITRE ATT&CK is a public catalogue of the steps attackers take, from getting in to causing damage. No source has mapped this flaw to it yet, so the steps below are our prediction from what the record says the flaw allows. They are not MITRE's, the vendor's or CISA's, and an attacker may do other things.

Steps an attacker would likely take with this flaw

The step, in plain words	Stage of an attack	Why we think so
Exploit Public-Facing Application (T1190): Breaking in through a service that faces the internet, such as a web server, VPN or login page.	Initial Access: getting in	The flaw is reachable over a network with no authentication; exploiting a public-facing App Service endpoint would likely be the initial access vector for an attacker.
Exploitation for Privilege Escalation (T1068): Using a flaw to gain higher rights than the account had, such as administrator or root.	Privilege Escalation: gaining higher rights	The record states the outcome is privilege escalation; CWE-75 injection crossing a privilege boundary would typically map to exploitation for privilege escalation.
Escape to Host (T1611): Breaking out of a container to reach the machine that hosts it.	Privilege Escalation: gaining higher rights	On a Linux container platform, privilege escalation from application level could lead to escape to host; the Linux context and maximum-severity outcome suggest this as a likely next step.

How to use this: ask your IT provider which control or log would show each step, for example whether sign-in attempts from the internet are recorded. If nobody can answer, that is the gap to close first.

MITRE ATT&CK® is a registered trademark of The MITRE Corporation. Technique names are MITRE's.

The sources, in their words

The CVE record:

> *Failure to sanitize special elements into a different plane (special element injection) in Azure App Service allows an unauthorized attacker to elevate privileges over a network.*

Versions of Microsoft Azure App Service for Linux the record lists as affected:

- 110.0.0.0 up to, but not including, 110.0.7.78, 111.0.7.3

The record gives no version ranges. Use the vendor's advisory for the affected versions.

Check your own project

If you build software, you can compare what a project declares with this CVE from a terminal. The tool reads requirements files, package.json, Maven, Gradle, Cargo, CMake and Dockerfile files, runs nothing, and only the CVE id leaves your machine. Naming the CVEs needs no key.

```
lattice-scan project . --cve CVE-2026-109484
```

Each row says in-range, outside-range, undetermined or lead, with the file and line. That is evidence about what a manifest declares, not about what runs, and a person decides. The verdicts, the CSV columns and a pipeline recipe are in the guide.

Check the servers you run

The same id can be checked against a host. Run this on the machine itself, not on the laptop that deploys to it: it reads installed packages (dpkg, rpm or apk), snaps, the hardware (maker, model, firmware), the running container images with their operating system and packages, and the kernel version, compares them on the spot, and only the CVE ids leave the machine; the inventory stays on it, in ~/.lattice-scan. Naming the CVEs needs no key. Do it on each server after a deployment, and again when a new entry lands.

```
lattice-scan --cve CVE-2026-109484
```

A match is a candidate for a person to confirm, not a finding. A distribution can fix a CVE under an unchanged version number, so a match is a candidate; on Debian-family hosts the report also shows whether the package's own changelog names the CVE. By default it opens the images your containers are running and reads their operating system and package list; a deep scan (--deep, a few minutes) opens every local image and the Python, Node, Java and Ruby packages inside them; for what an image declares, use the project check on its Dockerfile.

Guide: namiq.io/lattice/guides/choose-your-channel. MCP server (io.namiq/lattice in the MCP registry): namiq.io/lattice. Package: pypi.org/project/lattice-scan, or namiq.io/apt.

What we do not know yet

- No vendor advisory could be read; it is not known whether Microsoft has published a workaround, a confirmed fixed build, or a customer notification for this CVE.
- It is not known whether the platform update is applied automatically to all App Service for Linux environments or whether customer action is needed to trigger it.
- The record does not describe what privileges an attacker gains, nor whether exploitation could affect other tenants on the same underlying host infrastructure.

- It is not known what the 110.0.x and 111.0.x build numbers refer to in the Azure platform, making it difficult to identify affected environments without asking Microsoft.
- Exploitation is not recorded. That is not the same as none.
- This is a new record: scores, versions and wording can change in the first days, and they are as at 24 November 2026. We have not read the vendor's advisory page and we have not tested it.
- The sections headed "The bottom line", "Does this apply to you?", "What it is", "Why it matters", "Exposure by situation", "Things people will get wrong", "Likely ATT&CK mapping", "How we would triage it" and "Send this to your IT provider" are our interpretation of the sources, not a statement by CISA or the vendor.

Appendix 1. Severity decoded

CVE-2026-109484: score 10, `CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H/E:U/RL:O/RC:C` As the CVE record gives it.

Metric	Value	What it means here
Attack vector	Network	Reachable over a network, with no need to be on the same segment.
Attack complexity	Low	No special conditions beyond the flaw itself.
Privileges required	None	No account is needed.
User interaction	None	No one has to do anything.
Scope	Changed	Damage reaches beyond the vulnerable component.
Confidentiality	High	Total loss of confidentiality.
Integrity	High	Total loss of integrity.
Availability	High	Total loss of availability.

Appendix 2. Timeline

Date	Event
24 November 2026	CVE record published by Microsoft.

Appendix 3. Exposure by situation

These priorities are our judgement from the attacker prerequisites in the sources. They are not vendor ratings.

Your situation	Priority	Timeframe
Azure App Service for Linux on an affected build, reachable from the internet	Medium	This week: contact Microsoft and confirm the fix is in place
Azure App Service for Linux on an affected build, internal or VNet-only access	Medium	This week: confirm fix is applied; the no-auth requirement justifies urgency
Azure App Service for Linux, platform version unknown	Medium	This week: determine build version, then act as above if affected
Azure App Service for Linux, confirmed on a build at or above 110.0.7.78 or 111.0.7.3	Low	Confirm once in writing; no further action needed

Azure App Service on a Windows plan, not Linux	None	Nothing to do: the record names Linux only

Appendix 4. Definition of done

- Every Microsoft Azure App Service for Linux you run is listed, with its version.
- Each is outside the affected list, or the vendor's fix or mitigation is in place.
- The version output or a screenshot is saved with the date.
- Anything not done has an owner and a date.

Sources

- CVE record for CVE-2026-109484: https://www.cve.org/CVERecord?id=CVE-2026-109484
- CISA Known Exploited Vulnerabilities catalogue: https://www.cisa.gov/known-exploited-vulnerabilities-catalog
- Reference in the CVE record: msrc.microsoft.com: https://msrc.microsoft.com/update-guide/vulnerability/CVE-2026-109484