contact@namiq.io · namiq.io

NamiQ

SECURITY ADVISORY

ArcGIS Server SQL injection

CVE-2026-10757 · Esri ArcGIS Server 12.1 and earlier

Assessment & Action Plan

A 9.8 SQL injection in Esri ArcGIS Server where the score assumes an anonymous attacker, but the vendor says that applies only to services set to public. How to tell which of your servers that is.

PREPARED BY	Hung Le, NamiQ	DATE	2 November 2026
FACTS AS OF	26 October 2026	VERSION	1.1
SEVERITY	**Critical: CVSS 3.1 9.8**	STATUS	Patch available for 12.1, 12.0, 11.5, 11.3, 11.1

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/arcgis-server-sql-injection-cve-2026-10757

Update, 2 November 2026: this note now ends with a section on checking your own servers and projects with lattice-scan, for IT admins, AI agent skills and CI/CD, and the analysis report is reissued with it. Everything else is as of 26 October 2026; check CISA's catalog for any change since.

On 26 October 2026 Esri disclosed CVE-2026-10757, a SQL injection in ArcGIS Server, in its October security bulletin. It is scored 9.8, and several early summaries read that as "unauthenticated, anyone, anywhere". The vendor's wording is narrower, and the gap decides which of your servers are urgent.

This note is for you if you run ArcGIS Server, or host GIS services for others. Everything is as of 26 October 2026 and links to its source at the end.

The short version

- Affected: ArcGIS Server 12.1 and earlier, on Windows and Linux.
- A remote attacker with privileges to access the service can issue database-specific functions that should not be allowed. For a service in the non-default "public" access configuration, that attacker can be anonymous.
- CVSS 3.1 9.8, scored with no privileges required. CWE-89, SQL injection.
- Esri provides a patch for versions in General Availability and Extended support: 12.1, 12.0, 11.5, 11.3 and 11.1. Other versions are mature or retired and not patch-eligible; the advice there is to upgrade and apply the patch.
- The record does not name the vulnerable endpoint, report exploitation or include exploit code.

What the score assumes, and what the vendor says

The 9.8 uses PR:N, no privileges. Esri's description says the attacker needs privileges to access the service, and that anonymous access follows only if the service has been configured as public. That is not the default. So the same version can be a critical exposure on one server and a lower one on another.

Some write-ups go further: that the flaw is reachable at any REST endpoint, that it leads to operating-system command execution on the database host, or that it ranks with the worst flaws of recent years. The record says none of that. It says a caller can issue database functions that should not be allowed. How far that goes depends on the database, the account ArcGIS Server uses, and what that account may do.

Who is exposed

- Highest: ArcGIS Server services published as public, on the internet, on an affected version.
- High: services reachable by many authenticated users, such as a whole organisation or a customer base, because the attacker only needs the privilege to use the service.
- Lower: servers limited to a small trusted group, behind your own network boundary.
- No patch: versions that are mature or retired. They are exposed and cannot be fixed in place.

Our SOC analyst's assessment

- Treat any public-facing affected server as an emergency patch. A SQL injection in a widely deployed product with a clear fix is the sort of flaw that gets scanned for quickly. That is an estimate, not a fact about this CVE.
- GIS databases often hold more than maps: addresses, asset locations, infrastructure layouts. A data exposure from a GIS service can be reportable, so check what your database account can read.
- A web application firewall rule for SQL metacharacters is a stopgap at most. Filters are weak against injection in JSON bodies, and they do not replace the patch.
- Least privilege on the database account is the control that limits what any injection can do. Review the account's grants whether or not you are affected.
- Look back before looking forward: review database and web logs for the period before disclosure for unusual queries from service endpoints.

> *The score tells you the worst case. The configuration tells you which case you are in.*

Step 1: inventory and classify

```
# Version, from the server itself (the info endpoint reports currentVersion)
curl -s "https://YOUR-HOST/arcgis/rest/info?f=json"

# Then, in ArcGIS Server Manager or the admin API, list every service
# and its security setting: which are public, which need a login,
# which are limited to named roles.
```

Record each server's exact version and each service's access setting. The public ones on affected versions go first.

Step 2: patch

- Apply Esri's patch for your version. It exists for 12.1, 12.0, 11.5, 11.3 and 11.1.
- If you run a mature or retired version, plan an upgrade to a supported one and apply the patch there. Until then, reduce exposure as below.
- Test in staging first: server upgrades can disrupt running services.
- Verify the patch is installed on every server, not only the ones you remember.

If you cannot patch this week

- Change public services to require authentication where the use case allows.

- Limit network access to the server to the client networks that need it.
- Reduce the database account to the privileges the services need, and nothing on system schemas.
- Add a filter in front as a stopgap, knowing what it cannot see.

What to look for

- Requests to service endpoints carrying SQL syntax in parameters or JSON bodies, and unusual error responses from those endpoints.
- Database queries from the ArcGIS account that your services do not normally issue, such as system catalogue reads or functions outside the normal set.
- Large reads from tables behind the services, and new accounts or changed grants.

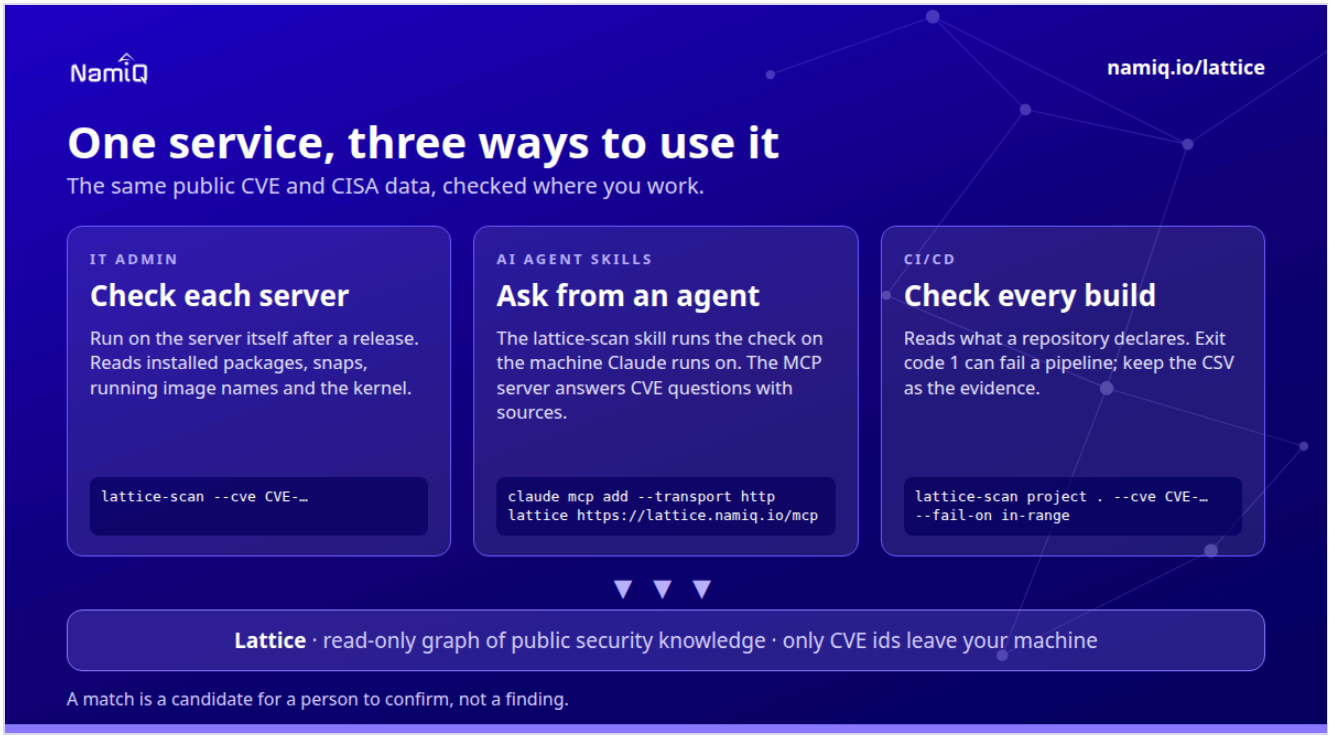
Check your own servers and projects

The facts above are as of 26 October 2026. lattice-scan, the open-source command line for checking this CVE against your own estate, cannot check this product: ArcGIS Server is commercial software installed outside the package managers the host check reads, and it is not declared in a project's manifests. A match from it would also never be a finding, only a candidate for a person to confirm.

What you can do instead: ask an AI agent connected to the Lattice MCP server to look up CVE-2026-10757. The keyless graph_lookup returns CISA's entry (listed, due date, required action) and the NVD entry (affected products and version ranges), each with its source. Then check the vendor's own advisory for your version. If you run Linux servers or build software around it, the command line still covers those parts.

Who can use it, and how

For	What to do	Command
AI agent skills	The lattice-scan skill runs the check on the machine Claude runs on. The Lattice MCP server answers questions about the CVE with its sources.	claude mcp add --transport http lattice https://lattice.namiq.io/mcp



Where the same check is used. The three ways read the same public data; only CVE ids leave your machine.

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

- The vulnerable endpoint and parameter. The record does not name them.
- Whether anyone is exploiting it, and whether public exploit code appears.
- Which Esri-based products inherit the issue.

If you take one thing from this: do not stop at the number. Find out which of your services are public, which version they run, and what the database account behind them is allowed to do. Those three facts are your exposure.

Sources

- Esri: October 2026 ArcGIS security bulletin: https://www.esri.com/arcgis-blog/products/trust-arcgis/administration/october-2026-arcgis-security-bulletin
- CVE-2026-10757 record (Esri CNA): https://www.cve.org/CVERecord?id=CVE-2026-10757
- CISA Known Exploited Vulnerabilities catalog: https://www.cisa.gov/known-exploited-vulnerabilities-catalog