GOVERNMENT OF THE REPUBLIC OF VANUATU

PRIME MINISTER'S OFFICE

CERTVU
DEPARTMENT OF COMMUNICATIONS & DIGITAL TRANSFORMATION

PM B 9108 Port Vila, Vanuatu
Tel: (678) 33380

GOUVERNEMENT DE LA REPUBLIQUE DU VANUATU

BUREAU DU PREMIER MINISTRE

CERTVU

DEPARTMENT DE COMMUNICATION ET DE TRANSFORMATION NUMERIQUE

SPP 9108 Port Vila, Vanuatu
Tel: (678) 33380

8 July 2026

Advisory 155: CVE-2026-54369_Improper Link Resolution Before File Access Vulnerability

Release Date: 06th July 2026
Impact: HIGH / CRITICAL
TLP: CLEAR

The Department of Communications and Digital Transformation (DCDT) through CERT Vanuatu (CERTVU), provides the following advisory.

This alert is relevant to Organizations and System/Network administrators that utilize the above products. This alert is intended to be understood by technical users and systems administrators.

What is it?

A symlink traversal vulnerability in libacl's pathname-based API functions:

- acl_get_file()
- acl_set_file()
- acl_extended_file()
- acl_delete_def_file()

These functions resolve a supplied pathname to read or write POSIX ACLs, but prior to 2.4.0 they did so without safely handling symbolic links encountered along the path. If any component of the path can be replaced with a symlink between when the caller decides on a path and when libacl actually resolves it, the operation can be redirected to a different file or directory than intended.

What are the systems affected?

Any Linux system with `libacl`/`acl` before version 2.4.0 installed — which is effectively all major distributions at the time of disclosure (Debian, Ubuntu, RHEL/Amazon Linux, etc., per the Debian security tracker and Snyk advisories). Practical risk is highest where:

- Privileged processes (root, or SUID/SGID binaries) call the affected libacl functions on paths that a lower-privileged local user can influence — e.g., backup tools, ACL restoration/provisioning scripts, home-directory or shared-storage ACL management, config management tooling (Ansible/Puppet/Chef ACL modules).
- Multi-user systems where non-root users have write access to some component of a path a privileged job later touches (shared uploads directories, home directories, temp/staging areas).

What does this mean?

• Attacker requires local, low-privilege access — no user interaction from a victim is needed.
• Attacker identifies a path that a privileged process will later pass to one of the four vulnerable functions (e.g., a scheduled job that restores ACLs recursively over a directory the attacker partly controls).
• Attacker replaces a component of that path with a symbolic link pointing at a sensitive target (e.g., `/etc/shadow`, a SUID binary, or another file whose ACL controls access).
• When the privileged process runs, libacl follows the symlink and applies the ACL read/write operation to the attacker's chosen target instead of the intended file.
• Depending on direction (read vs. write) this can disclose ACL data on a protected file, or — more seriously — let the attacker grant themselves ACL-based access to a file they shouldn't be able to touch, which can chain into privilege escalation (e.g., granting themselves write access to a SUID binary or a credentials file).

Mitigation process

CERTVU recommends the following:

• **Patch:** Upgrade `acl`/`libacl` to 2.4.0 or later via the distribution package manager. Apply the equivalent fix for `attr` (2.6.0+) at the same time, since CVE-2026-54371 shares the same class of bug.
• **Compensating controls until patched:**

- Restrict write access to any directory tree that privileged jobs will later run `setfacl`/`getfacl`/`chacl` or libacl-based tooling against — don't let low-privilege users control any path component in that tree.
- Audit and harden backup, provisioning, and ACL-restoration scripts that run as root over user-writable paths; where possible, have them operate via file descriptors (`openat`/`AT_SYMLINK_NOFOLLOW`) rather than raw pathnames — the actual fix in 2.4.0 adds symlink-safe APIs for this.

- Enable audit rules on `symlink`/`symlinkat` creation events correlated with subsequent `setfacl`/`getfacl`/ACL syscalls, to detect exploitation attempts.
- File-integrity monitor SUID/SGID binaries and sensitive files for unexpected ACL changes.

• **Verification:** Confirm patched version with `getfacl --version` / package manager query (`dpkg -l acl`, `rpm -q acl`) across all Linux endpoints, prioritizing multi-user servers and any host running scheduled ACL-management jobs as root.

Reference

1. https://www.cisa.gov/news-events/bulletins/sb26-187
2. https://www.cve.org/CVERecord?id=CVE-2026-54369

.