

CVE-2026-98226

Microsoft mm, swap: fix SWAP_USAGE_OFFLIST_BIT collision with real usage count

In the Linux kernel, the following vulnerability has been resolved: mm, swap: fix SWAP_USAGE_OFFLIST_BIT collision with real usage count SWAP_USAGE_OFFLIST_BIT is embedded in the si->inuse_pages usage counter, and is meant to sit above any value that counter can reach. However, it is defined from BITS_PER_TYPE(atomic_t), so it is bit 30. On a system with 4 KiB pages the flag collides with the usage count once that count reaches 4 TiB.

EVRS 0 /100 Informational priority	CVSS — Not scored	EPSS 0.002 6.4th percentile · FIRST	CISA DEADLINE — No CISA deadline
---	-------------------------	--	--

Signals behind the score

☒ counted in the score ☐ not counted

evidence for 2 of 8 signals

☐ CISA KEY	Not currently CISA KEY-listed	☐ Vendor exploitation	Active-exploitation status not yet checked - scored at the population base rate (none) until known
☐ Ransomware	Ransomware association unavailable	☐ KEV recency	CISA KEY listing date unavailable
☐ EPSS	0.002 probability	☐ Pre-patch disclosure	Disclosure status unavailable
☐ CVSS	CVSS base score unavailable	☐ Microsoft Exploitability Index	Not a Microsoft CVE

Remediation

Monitor Untagged reference available; patch status is not confirmed

NVD lists one or more references for this CVE that aren't tagged as a patch or vendor advisory. Review them directly to confirm patch availability and applicability before making a remediation decision.

Basis: NVD reference (untagged)

swap_usage_in_pages() masks bit 30 out, so whenever the real count has that bit set, every caller of it reads 4 TiB low: * /proc/ swaps understates Used by 4 TiB. * A raw count of exactly 2^30 masks to zero, so try_to_unuse() takes its "if (! swap_usage_in_pages(si)) goto success;" early exit and swapoff tears the device down while pages are still swapped out. Nothing in the rest of swapoff aborts the teardown, so those pages are lost. Independently of swapoff, the collision also corrupts the counter and the plist. On a device in normal use, a free that leaves bit 30 set in the count makes swap_usage_sub() see the flag where there is only count, and call add_to_avail_list(). It clears the bit with fetch_and(~SWAP_USAGE_OFFLIST_BIT), leaving the stored count 4 TiB below the real one, and calls plist_add() on a device that is already listed, tripping the WARN_ON(!plist_node_empty(node)) in plist_add() and linking the node a second time. Change the definition of SWAP_USAGE_OFFLIST_BIT to be based on atomic_long_t instead. Note that the usage counter field itself is of this same type, so it is still a valid bit.

Affected and fixed versions

PACKAGE	AFFECTED	FIXED IN	SOURCE
Not identified by the connected sources	—	no fixed version listed	NVD

Sources

Reference	git.kernel.org/stable/c/0a41a46859df0d2bd7ed127197aac7c7389c00ed	Additional reference	git.kernel.org/stable/c/12e9ac7bc5b254048f886bf421e3a15491106c1f
Additional reference	git.kernel.org/stable/c/7c0d0d6de46076b43a2acaf1026b6648f8434c39	NVD	nvd.nist.gov/vuln/detail/CVE-2026-98226
ENISA EUVD	euvd.enisa.europa.eu/vulnerability/EUVD-2026-93154	Current data	evulnable.com/vulnerability-management/cve/CVE-2026-98226

Snapshot at generation time. EPSS can move quickly. A public exploit module is not evidence of active exploitation.