

CVE-2026-98562

Microsoft net: libwx: fix races in Tx timestamp handling

In the Linux kernel, the following vulnerability has been resolved: net: libwx: fix races in Tx timestamp handling wx->ptp_tx_skb is shared between the Tx path, the PTP auxiliary worker and the timestamp cleanup paths. The WX_STATE_PTP_TX_IN_PROGRESS bit prevents multiple Tx paths from submitting timestamp requests, but does not serialize the worker against cleanup. As a result, wx_ptp_clear_tx_timestamp() can free an skb after wx_ptp_tx_hwtstamp_work() has obtained its pointer.

EVRS 0 /100 Informational priority	CVSS — Not scored	EPSS 0.002 8.1st percentile · FIRST	CISA DEADLINE — No CISA deadline
---	---------------------------------------	---	--

Signals behind the score

☒ counted in the score ☐ not counted evidence for 2 of 8 signals

☐ CISA KEV	Not currently CISA KEV-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 KEV 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)

The worker may then pass the freed skb to skb_tstamp_tx() and release the same reference again. The cleanup path may also clear the in-progress bit while the worker is still processing the old skb. This allows the Tx path to publish a new skb which the worker can subsequently overwrite with NULL, leaking its reference. Add a dedicated spinlock to protect publication and consumption of the Tx timestamp skb. Detach the skb and clear the in-progress bit while holding the lock, then deliver the timestamp and release the skb after dropping it. Use the same locked cleanup in the quiesce path, but keep the detach there free of register accesses: quiesce runs during PCIe error recovery, where MMIO is not reliable, and it deliberately did not touch the device before. The lock is taken with interrupts disabled, because netpoll can call ndo_start_xmit() with hard interrupts already off. When handling a Tx DMA mapping failure, keep the transmit path reference until after comparing the skb under the lock. This prevents skb address reuse from making the error path mistake a newer timestamp request for the failed one.

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/3173cba1170131816972ed2b6185cb970ba8b747

NVD nvd.nist.gov/vuln/detail/CVE-2026-98562

Current data evaluable.com/vulnerability-management/cve/CVE-2026-98562

Additional reference git.kernel.org/stable/c/e78b95cfd5df25d0c562aaef95d3c7fe6c7e0e66

ENISA EUVD euvd.enisa.europa.eu/vulnerability/EUVD-2026-103607

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