

CVE-2026-98496

Microsoft net: airoha: npu: cancel wdt_work after releasing the WDT IRQ

In the Linux kernel, the following vulnerability has been resolved: net: airoha: npu: cancel wdt_work after releasing the WDT IRQ airoha_npu_remove() calls cancel_work_sync() on each core's wdt_work, but the watchdog IRQ that queues it is requested with devm_request_irq() and is freed only after .remove() returns. airoha_npu_wdt_handler() can therefore schedule_work() again once the cancel has returned.

EVRS 0 /100 Informational priority	CVSS — Not scored	EPSS 0.002 9th 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)

struct airoha_npu, which contains the work, is devm_kzalloc()'d and is freed in that same unwind, so the late work dereferences freed memory. Register the work with devm_work_autocancel() before devm_request_irq() and drop .remove(). Devres runs in reverse order, so the IRQ is freed before cancel_work_sync(), including when probe fails. A cancel left in .remove() cannot get that order. Initializing the work first also stops a pending watchdog interrupt from queuing an uninitialized work item. Probe currently calls INIT_WORK() only after devm_request_irq(). This issue was identified during our ongoing static-analysis research while reviewing kernel code.

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/4bdee8060d1e4581624e68fbd369b1afb14df4bc	Additional reference	git.kernel.org/stable/c/aca91d8fa0ecbc990ebf9a688c684dda78a43c00
Additional reference	git.kernel.org/stable/c/da9d42531f3a6dfea2e9c775b62a46185dbf280a	NVD	nvd.nist.gov/vuln/detail/CVE-2026-98496
ENISA EUVD	euvd.enisa.europa.eu/vulnerability/EUVD-2026-103736	Current data	evulnable.com/vulnerability-management/cve/CVE-2026-98496

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