

CVE-2026-98361

Microsoft RDMA/rxe: Restore HMM_PFN_WRITE check in ODP write paths

In the Linux kernel, the following vulnerability has been resolved: RDMA/rxe: Restore HMM_PFN_WRITE check in ODP write paths Commit 0b261d7c1cd3 ("RDMA/rxe: Break endless pagefault loop for RO pages") dropped the access permission test from rxe_check_pagefault() and left only HMM_PFN_VALID. A page faulted in read-only, for example a page-cache folio behind a PROT_READ file mapping, then satisfies the check and ODP write operations (RDMA WRITE, RDMA READ response, SEND payload, atomics) modify it through kmap

EVRS 12 /100 Elevated priority	CVSS V3.1 7.8 High · NVD	EPSS 0.001 2.2nd percentile · FIRST	CISA DEADLINE — No CISA deadline
--	--	---	---

Signals behind the score

☒ counted in the score ☐ not counted

evidence for 3 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	0.001 probability	☐ Pre-patch disclosure	Disclosure status unavailable
☒ CVSS	Base score 7.8	☐ Microsoft Exploitability Index	Not a Microsoft CVE

Remediation

Scheduled Maintenance Patch or fixed version identified

Basis: External remediation intelligence

without ever breaking CoW. An unprivileged user can register an ODP MR over such a mapping and have incoming RDMA traffic overwrite the page cache of a file it only holds O_RDONLY, including /etc/passwd or setuid binaries. This is the same primitive class as Dirty COW and CVE-2022-2590. mlx5 has the missing invariant: its ODP path sets the device write bit only for pfns that carry HMM_PFN_WRITE. Restore it in rxe by requiring HMM_PFN_WRITE in rxe_check_pagefault() for every operation except RXE_PAGEFAULT_RDONLY. A write to a non-writable VMA now fails the one fault attempt with -EPERM from hmm_vma_fault() instead of re-faulting forever. For a writable VMA the fault breaks CoW and the write lands in the private page. Keep pmem flushes on the read-only check. arch_wb_cache_pmem() never modifies memory, and the FLUSH access bits do not make the umem writable, so classifying flushes as writes would make every flush against a flush-only MR fail.

Affected and fixed versions

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

Sources

Vendor guidance	git.kernel.org/stable/c/2d6c6f94d3e1cde049fe5db71dbd5549f10b2de1	Additional guidance	git.kernel.org/stable/c/769001ce838d907ecaa95f1d0a4e8fc86f761f9f
Additional guidance	git.kernel.org/stable/c/c6f56982ec5c7da83c4bdfcce3b6db18bcfe6425	NVD	nvd.nist.gov/vuln/detail/CVE-2026-98361
ENISA EUVD	euvd.enisa.europa.eu/vulnerability/EUVD-2026-93346	Current data	evulnable.com/vulnerability-management/cve/CVE-2026-98361

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