

CVE-2026-98467

Microsoft KVM: arm64: nv: Fix null ptr deref on nested wp/unmap, teardown race

In the Linux kernel, the following vulnerability has been resolved: KVM: arm64: nv: Fix null ptr deref on nested wp/unmap, teardown race Commit 7270cc9157f4 ("KVM: arm64: nv: Handle VNCR_EL2 invalidation from MMU notifiers") introduced VNCR_EL2 invalidation in both `kvm_nested_s2_unmap()` and `kvm_nested_s2_wp()`. However at the point of this being performed concurrent stage 2 teardown of a nested guest can cause `kvm->arch.mmu.pgt` to be set to NULL.

EVRS 0 /100 Informational priority	CVSS — Not scored	EPSS 0.002 5.5th 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)

This happens in `kvm_flush_shadow_all()` -> `kvm_arch_flush_shadow_all()` -> `kvm_free_stage2_pgd()` and is performed under the `kvm->mmu_lock`. Commit `ec14c272408a` ("KVM: arm64: nv: Unmap/flush shadow stage 2 page tables") introduced the teardown of the entire nested MMU range, which then invokes `stage2_apply_range()` with `resched=true`: `mmu_notifier_invalidate_range_start()` -> ... -> `kvm_mmu_notifier_invalidate_range_start()` -> `kvm_mmu_unmap_gfn_range()` -> `kvm_unmap_gfn_range()` -> `kvm_nested_s2_unmap()` -> `kvm_stage2_unmap_range()` -> `__unmap_stage2_range()` -> `stage2_apply_range()`. This means that `stage2_apply_range()` can drop the `kvm->mmu_lock` and thus concurrent progress can be made in lockstep with `kvm_arch_flush_shadow_all()`. If `kvm_arch_flush_shadow_all()` advances ahead of `stage2_apply_range()` and completes its operation it guarantees a NULL pointer deref. Since `kvm_free_stage2_pgd()` is performed under the `kvm->mmu_lock` this will either be observed NULL or not and serialised against `kvm_free_stage2_pgd()`. Resolve the issue by abstracting the invalidation to a new function, `kvm_invalidate_vnocr_ipa_all()`, and check that the `pgt` is non-NULL before dereferencing it.

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/3cefc1ab511fad6a0f708e7a4aee9f91346e9457	Additional reference	git.kernel.org/stable/c/4c74e233cdedd11592775fae2a6243e67ca3f891
Additional reference	git.kernel.org/stable/c/970de111c087488e53cadf5480b94e0f695db985	NVD	nvd.nist.gov/vuln/detail/CVE-2026-98467
ENISA EUVD	euvd.enisa.europa.eu/vulnerability/EUVD-2026-103707	Current data	evulnable.com/vulnerability-management/cve/CVE-2026-98467



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