CVE-2026-23416 — Linux Kernel `mseal` Invariant Violation

Discovered by: Antonius (Blue Dragon Security)
Country : Indonesia
www.bluedragonsec.com
github.com/bluedragonsecurity

Overview

An invariant violation (`VM_WARN_ON_VMG`) fires at `mm/vma.c:830` inside `vma_merge_existing_range()` when `mseal(2)` is called with a range spanning two adjacent VMAs where one has `VM_SEALED` set and the other does not.

Affected Versions

Linux kernel 6.17 through Linux kernel 7.0-rc5 (confirmed).

Call Path

The vulnerability is triggered through the following call chain: `mseal(2)` → `do_mseal()` in `mm/mseal.c` → `mseal_apply()` → `vma_modify_flags()` in `mm/vma.c` → `vma_modify()` → `vma_merge_existing_range()`, where the `VM_WARN_ON_VMG` assertion fires at line 830.

Root Cause

`do_mseal()` calls `vma_modify_flags()` with the original `mseal()` start address without clamping it to the current VMA's `vm_start` when the mseal range spans two VMAs with different `VM_SEALED` states. This causes `vma_merge_existing_range()` to receive an inconsistent `vmg` state, triggering the assertion: `vmg->start != middle->vm_start`.

Security Relevance

The bug is reachable from unprivileged userspace (UID 1000, no capabilities required — only `memfd_create`, `mmap`, and `mseal` syscalls are needed). Since `mseal(2)` is itself a security primitive protecting VMA immutability, an invariant violation in its application logic means `VM_SEALED` may be applied incorrectly when spanning VMAs with mixed seal states, potentially undermining the security guarantee `mseal` provides. In production kernels where `WARN` compiles to a no-op, the inconsistent `vmg` state proceeds silently — the VMA tree could be left with incorrect seal state without any visible error.

Exploitation Characteristics

- **Access required:** Unprivileged (UID 1000, no `CAP_*`)
- **Reproducibility:** 100% deterministic, triggers in under 1 second, no fault injection needed

- **Impact:** Silent corruption of VMA seal state in production kernels, potentially allowing sealed memory regions to be incorrectly modified

References

https://github.com/bluedragonsecurity/CVE-2026-23416-POC

https://www.cve.org/CVERecord?id=CVE-2026-23416

https://git.kernel.org/pub/scm/linux/kernel/git/stable/linux.git/commit/?id=40b3f4700e5535fbe74738cebb9379a40ec66bed

https://git.kernel.org/pub/scm/linux/kernel/git/stable/linux.git/commit/?id=83737e34b83a23b2a9bcf586b058b2c2a54c7c6b

https://lore.kernel.org/all/?q=bluedragonsecurity

https://medium.com/@w1sdom/cve-2026-23416-poc-affecting-linux-kernel-6-17-linux-kernel-7-0-rc-5-457afc9ad9e3