Linux bitten by second severe vulnerability in as many weeks

Both privilege escalation vulnerabilities stem from bugs in the kernel's handling of page caches stored in memory, allowing untrusted users to modify them. They target caches in networking and memory-fragment handling components. Specifically, CVE-2026-43284 attacks the esp4 and esp6 () processes, and CVE-2026-43500 zeroes in on rxrpc. Last week's CopyFail exploited faulty page caching in the authencesn AEAD template process, which is used for IPsec extended sequence numbers. A 2022 vulnerability named Dirty Pipe also stemmed from flaws that allow attackers to overwrite page caches.

Researchers from security firm Automox wrote:

> *Dirty Frag belongs to the same bug family as Dirty Pipe and Copy Fail, but it targets the frag member of the kernel's struct sk_buff rather than pipe_buffer. The exploit uses splice() to plant a reference to a read-only page-cache page (for example, /etc/passwd or /usr/bin/su) into the frag slot of a sender-side skb. Receiver-side kernel code then performs in-place cryptographic operations on that frag, modifying the page cache in RAM. Every subsequent read of the file sees the corrupted version, even though the attacker only ever had read access.*

CVE-2026-43284 is found in the esp_input() process on the IPsec ESP receive path. When an skb object is non-linear but lacks a frag list, the code skips skb_cow_data() and decrypts AEAD in place on the planted frag. From there, an attacker can

control the file offset and the 4-byte value of each store.

CVE-2026-43500, meanwhile, resides in rxkad_verify_packet_1(). The process decrypts RxRPC payloads using a single-block process. Splice-pinned pages become both a source and destination. That, paired with the decryption key being freely extracted using the add_key (rxrpc), allows an attacker to rewrite contents in memory.

Either exploit used separately is unreliable. Some Ubuntu configurations use AppArmor to prevent untrusted users from creating namespace contents. That, in turn, neutralizes the ESP technique. Most other distributions by default don't run rxrpc.ko, which neutralizes the RxRPC arm. When chained together, however, the two exploits allow attackers to obtain root on every major distribution Kim tested. Once the exploits run, attackers can use SSH access, web-shell execution, container escapes, or compromise low-privilege accounts.

"Dirty Frag is notable because it introduces multiple kernel attack paths involving rxrpc and esp/xfrm networking components to improve exploitation reliability," Microsoft researchers wrote. "Rather than relying on narrow timing windows or unstable corruption conditions often associated with Linux local privilege escalation exploits, Dirty Frag appears designed to increase consistency across vulnerable environments."

Researchers at Google-owned Wiz said exploits will be less likely to break out of hardened containerized environments such as Kubernets with default security settings in place. "However, the risk remains significant for virtual machines or less restricted environments."

The best response for anyone using Linux is to install patches immediately. While fixes likely require a reboot, protection from a threat as severe as Dirty Frag outweighs the cost of disruptions. Anyone who can't install immediately should

follow the mitigation steps laid out in the posts linked above. Additional guidance can be found here.

https://arstechnica.com/security/2026/05/linux-bitten-by-second-severe-vulnerability-in-as-many-weeks/