Vulnerabilități Linux Kernel tip „DirtyFrag / Copy Fail class”

Vulnerabilități Linux Kernel tip „DirtyFrag / Copy Fail class”

Și CVE-2026-43284 CVE-2026-43500

CVE/ID/denumire	Condiții pentru exploatare	Descriere succintă	CVSS (NVD/alte surse)	Tip de vulnerabilitate	Mod de exploatare
Dirty CBC ("Linux-RxGK decrypt-before-MAC")	Utilizator local neprivilegiat, kernel cu suport **AF_RXRPC** şi securitate **RxGK/YFS-RxGK** neactualizată; necesită drept de citire asupra unui fişier SUID	Cercetătorii Delphos Labs au arătat că rutina *rxgk_decrypt_skb()* din kernelul Linux decriptează **AES-256-CTS-HMAC-SHA1** în locul unde se află datele — scatterlist-ul poate conţine pagini din page cache livrate prin MSG_SPLICE_PAGES. Deoarece decriptarea are loc înainte de verificarea MAC-ului, paginile sunt mutate în cache chiar dacă autentificarea eşuează. Atacatorul poate construi un flux de pachete RXRPC astfel încât să intercaleze pagini controlate cu pagini ale unui fişier ţintă (/usr/bin/su), iar decryptarea CBC va scrie un payload ales în cache.	— (nu există CVE public la data redactării; vulnerability deschisă pe 15 mai 2026)	Scriere în page cache / LPE	Exploitul oferit de Delphos Labs arată cum un proces local poate plasa un payload ELF de 192 B în page cache-ul */usr/bin/*su folosind *splice()* pentru a planta pagini din fişier în bufferul rxrpc, apoi trimiterea acestuia către un server RXRPC controlat. La executarea binarului corupt se obţine shell root.
CVE-2026-43284 – Dirty Frag (xfrm-ESP)	Utilizator local neprivilegiat, dar necesită posibilitatea de a crea o asociere **IPsec** (de obicei prin dobândirea CAP_NET_ADMIN în interiorul unui	Vuln. logică în subsistemul **xfrm-ESP** al kernelului (recepţia IPsec). Când un buffer de pachete include fragmente provenite din page cache prin *splice()*, funcţia decriptare ESP scrie direct peste acele pagini. Atacatorul poate suprascrie 4 octeţi oriunde în pagina ţintă, repetând operaţia pentru a construi un payload de	**8,8 HIGH** (CVSS 3.1 la nivel de comunitate – risc mare)	LPE / scriere în page cache	Atacatorul foloseşte *splice()* pentru a aduce pagini din page cache în socketul ESP-in-UDP, trimite pachete IPsec astfel încât decriptarea în locul nepotrivit să scrie 4 octeţi aleşi; repetiţia permite inserarea unui shell-code de

Vulnerabilități Linux Kernel tip „DirtyFrag / Copy Fail class”

	user-namespace). Kernelurile care au activat modulele esp4/esp6 sunt expuse.	192 B peste un binar SUID (de exemplu /usr/bin/su).			192 B. Exploatarea necesită încărcarea modulelor esp4/esp6 şi acces la IPsec.
CVE-2026-43500 – Dirty Frag (RxRPC)	Utilizator local neprivilegiat; kernelul cu modul **rxrpc** activ; nu necesită capacităţi speciale.	Vulnerabilitate în **RxRPC**: datele din pachetul RESPONSE sunt decriptate in place. Dacă fragmentul de skbuff provine din page cache (prin splice()), decriptarea scrie un bloc de 8 octeţi în page cache înainte de verificarea integrităţii. Folosind această primitivă plus faptul că schema FCRYPT are chei de 56 biţi, atacatorul poate calcula offline un token care, la decriptare, modifică /etc/passwd astfel încât parola root devine goală.	**7,8 HIGH** (CVSS 3.1 NVD)	LPE / scriere în page cache	Atacatorul foloseşte *splice()* pentru a aduce pagina */etc/passwd* în bufferul RxRPC, apoi construieşte trei operaţii *splice()* suprapuse. Decriptarea în RxRPC scrie un bloc de 8 octeţi, transformând root:x:0:0: în root::0:0:, permiţând autentificarea root fără parolă.

Descriere

Vulnerabilităţile Dirty CBC şi Dirty Frag se înscriu în acelaşi tipar ca bug-urile Dirty Pipe (2022) şi Copy Fail (CVE-2026-31431) – toate exploatează optimizări in-place în subsisteme criptografice sau de I/O ale kernelului, care procesează date provenite din page cache prin splice() fără a copia memoria. Atacatorii folosesc această proprietate pentru a modifica în mod determinist page cache-ul fişierelor setuid sau a fişierelor de configurare sensibile (e.g. /etc/passwd) fără a schimba conţinutul de pe disc. În textul tehnic publicat de un cercetător rus, se subliniază că:

- Dirty Frag cuprinde două vulnerabilităţi similare; una foloseşte ESP/IPsec pentru a suprascrie /usr/bin/su, iar cealaltă foloseşte RxRPC pentru a modifica /etc/passwd. Exploatarea este determinstă, nu implică condiţii de cursă şi funcţionează pe diverse distribuţii (Ubuntu 24.04, RHEL 10.1, openSUSE Tumbleweed, CentOS Stream 10, AlmaLinux 10, Fedora 44). Cercetătorul notează că la momentul dezvăluirii nu existau CVE alocate sau patch-uri disponibile deoarece embargo-ul a fost încălcat.
- Bugurile provind din optimizări introduse în 2017 (ESP) şi 2023 (RxRPC); până la descoperire au stat ascunse ani întregi.

Vulnerabilități Linux Kernel tip „DirtyFrag / Copy Fail class”

Cauza principală şi detalii tehnice

Dirty CBC (RxGK decrypt-before-MAC)

În RxGK, token-urile Kerberos sunt decriptate cu **AES-256-CTS-HMAC-SHA1**. Funcţia *rxgk_decrypt_skb()* mapează regiunea de skbuff într-un scatterlist şi cheamă *crypto_krb5_decrypt()* cu acelaşi src şi dst, adică decriptare în locul exact unde sunt stocate datele. Ordinea operaţiilor din construcţia Krb5 este „decriptează apoi verifică MAC”; astfel, chiar dacă autentificarea eşuează, plaintextul este deja scris. Dacă *skb_to_sgvec()* a mappat page cache-ul unui fişier prin **MSG_SPLICE_PAGES,** page cache-ul acelui fişier este modificat. Commit-ul aa54b1d27fe0 copiază pachetele cu fragmente partajate într-un buffer liniar înainte de decriptare.

CVE-2026-43284 – xfrm-ESP

În subsistemul ESP, decriptarea se face tot în locul în care se află datele. Optimizarea din 2017 permite ca paginile din page cache să ajungă în scatterlist-ul cu scriere când se utilizează *splice()/sendfile()/MSG_SPLICE_PAGES*. Când se procesează un pachet ESP în UDP-encapsulation, funcţia de decriptare scrie 4 octeţi din AAD (seqno_lo) imediat după zona criptată, în paginile page cache ale binarului ţintă. Repetând operaţiile *splice()* şi *sendmsg()* se pot scrie mai multe cuvinte până ce un payload complet acoperă prima pagină din /usr/bin/su.

CVE-2026-43500 – RxRPC

În RxRPC, rutina *crypto_krb5_decrypt*() decriptă pachetele DATA/RESPONSE direct pe scatterlist-ul de fragmente. Dacă skbuff-ul nu este clonat, dar conţine fragmente externe (setate prin *SKBFL_SHARED_FRAG sau skb_has_frag_list()),* acesta nu este copiat înainte de decriptare. Commitul upstream extinde condiţia de copiere, astfel încât pachetele cu fragmente partajate să fie copiate într-un buffer liniar înainte de decriptare. Fără patch, un atacator poate construi un token RxRPC (rxkad) cu o cheie determinată offline, astfel încât decriptarea PCBC/FCRYPT să scrie un bloc de 8 octeţi dorit; prin trei operaţii *splice()* se suprapune scrierea peste şirul root:x:0:0: din /etc/passwd, transformându-l în root::0:0:.

Versiuni afectate

Distribuție	CVE-2026-43284 (ESP)	CVE-2026-43500 (RxRPC)	Dirty CBC (RxGK decrypt-before-MAC)
Ubuntu (toate versiunile active)	Afectat	✓ Afectat	Rutina *rxgk_decrypt_skb()* din RxGK decriptează token-urile Kerberos „în locul” paginilor din page cache. Delphos Labs a demonstrat că toate kernelurile cu suport AF_RXRPC şi modul RxGK sunt vulnerabile până la commitul aa54b1d27fe0 din 8 mai 2026. Marea majoritate a distribuţiilor nu livrează modulul RxGK implicit, de aceea
RHEL 8 / CentOS Stream 8	Afectat	✓ Afectat	
RHEL 9 / CentOS Stream 9	Afectat	✗ Neafectat	
RHEL 10.1	Afectat	✗ Neafectat	
AlmaLinux 8	Afectat	✗ Neafectat	
AlmaLinux 9 / 10	Afectat	⚠ Doar cu kernel-modules-partner	
openSUSE Tumbleweed	Afectat	✓ Afectat	

Fedora 44	Afectat	✓ Afectat	impactul este restrâns la sisteme care compilează manual RxGK.
CloudLinux 7h / 8	Afectat	✓ Afectat	
OpenShift 4	Afectat	✗ Neafectat	

Consecinţe şi impact

- Escaladare de privilegii: Atacatorul local obţine acces root fără a modifica fişierele pe disc; schimbarea se află numai în page cache. La următoarea execuţie a binarului sau la verificarea parolei, se foloseşte pagina coruptă din RAM.
- Invizibilitate faţă de instrumente de integritate: File-hash-urile rămân neschimbate; doar conţinutul din RAM este modificat.
- Portabilitate: exploitul funcţionează pe mai multe distribuţii şi arhitecturi fără adaptări majore.
- Vector de escape container: Page cache-ul este partajat între containere; coruperea /usr/bin/su sau /etc/passwd într-un container poate permite evadarea în host.

Detectare şi monitorizare

- Monitorizarea încărcării modulelor: urmăriţi dacă modulele esp4, esp6 sau rxrpc sunt încărcate (lsmod | grep -E "esp4|esp6|rxrpc"). Absenta lor nu garantează siguranţa, deoarece pot fi încărcate la cerere.
- Detectarea apelurilor neobișnuite: monitorizaţi apeluri *splice()/vmsplice()* în procese neprivilegiate, în special în combinaţie cu sockets IPsec sau RxRPC.
- Integritate page cache vs. disc: analizaţi discrepanţele între hash-urile fişierelor din cache şi de pe disc; comportamente precum execuţia repetată a /usr/bin/su cu rezultat root fără parolă pot indica compromiterea.
- Jurnale de autentificare: semnalaţi obţinerea privilegiilor root fără autentificare normală (sudo/su), mai ales dacă apar erori de decriptare AEAD sau RxRPC în log-uri.

TTP-uri MITRE ATT&CK:

T1068	Exploitation for Privilege Escalation: scrierea în page cache a unui binar SUID şi execuţia ulterioară.
T1548.001	Abuse Elevation Control Mechanism (setuid): folosirea binarului setuid /usr/bin/su pentru a obţine UID 0
T1611	Escape to Host: compromiterea page cache-ului într-un container pentru a evada pe host.
T1055	Process Injection: injecţie de shellcode în page cache fără a scrie pe disc.
T1059.006	Command and Scripting Interpreter: Python: exploitul original Copy Fail foloseşte un script Python; variantele Dirty Frag/Dirty CBC pot fi implementate în C/Python.

Vulnerabilități Linux Kernel tip „DirtyFrag / Copy Fail class”

Măsuri de remediere şi atenuare

1. Identificarea sistemelor afectate:
- Rulaţi uname -r pentru a determina versiunea kernelului şi consultaţi avertizările distribuitorului.
- Verificaţi dacă modulele esp4, esp6 şi rxrpc sunt încărcate (lsmod | grep -E 'esp4|esp6|rxrpc').
2. Dezactivarea temporară a modulelor vulnerabile (mitigare recomandată de Delphos Labs şi de centrele naţionale):
- Creaţi un fişier de configurare dirtyfrag.conf în /etc/modprobe.d care să blocheze încărcarea modulelor ESP şi RxRPC:
  „ printf 'install esp4 /bin/false\ninstall esp6 /bin/false\ninstall rxrpc /bin/false\n' \
  | sudo tee /etc/modprobe.d/dirtyfrag.conf
  sudo rmmod esp4 esp6 rxrpc 2>/dev/null || true ”
  Această măsură previne exploatarea, dar poate întrerupe VPN-urile IPsec şi AFS; evaluaţi impactul operaţional înainte de aplicare.
3. Aplicarea patch-urilor oficiale:
- Urmăriţi anunţurile distribuitorului dvs. (Ubuntu, Red Hat, SUSE etc.) şi actualizaţi kernelul imediat ce versiunile corectate devin disponibile.
- Pentru Dirty CBC, kernelul upstream a inclus un patch în commitul aa54b1d27fe0 (10 mai 2026). Distribuţiile vor incorpora acest patch în release-urile stabile.
4. Restricţionarea accesului local:
- Menţineţi politicile SELinux/AppArmor în mod enforcing şi limitaţi capabilităţile, în special CAP_NET_ADMIN, astfel încât atacatorii să nu poată crea asociaţii IPsec.
- Evitaţi să acordaţi acces shell inutil utilizatorilor neprivilegiaţi; implementaţi mecanisme MFA şi monitorizaţi escaladările de privilegii.
5. Monitorizarea şi reacţia la posibile compromiteri:
- Dacă suspectaţi exploatarea, eliberaţi page cache-ul pentru a înlătura paginile corupte: echo 3 | sudo tee /proc/sys/vm/drop_caches.
- Investigaţi şi reinstalaţi binarele setuid afectate; analizaţi integritatea /etc/passwd şi resetaţi parolele în cazul în care s-a constatat modificarea.
6. Planificarea pe termen lung:
- Revizuiţi proiectarea serviciilor interne care folosesc criptografie în spaţiul kernel şi evitaţi operaţiile in-place pe date care pot proveni din spaţiul utilizator.
- Susţineţi adoptarea mecanismelor de cod review şi testare pentru noile patch-uri, astfel încât optimizările viitoare să nu introducă bug-uri similare.