الهيئة الوطنية للأمن السيبراني
National Cybersecurity Authority

تمت مشاركة هذه المعلومة بإشارة مشاركة ***أبيض*** حيث يسمح بتبادلها أو نشرها من خلال القنوات العامة.

في ضوء دور الهيئة الوطنية للأمن السيبراني للمساعدة في حماية الفضاء السيبراني الوطني، تود الهيئة مشاركتكم النشرة الأسبوعية للثغرات المسجلة من قبل the National Institute of Standards and Technology (NIST) National Vulnerability Database (NVD) للأسبوع من 19 أبريل إلى 25 أبريل. علماً أنه يتم تصنيف هذه الثغرات باستخدام معيار Common Vulnerability Scoring System (CVSS) حيث يتم تصنيف الثغرات بناء على التالي:

- عالي جدًا: النتيجة الأساسية لـCVSS 9.0-10.0
- عالي: النتيجة الأساسية لـCVSS 7.0-8.9
- متوسط: النتيجة الأساسية لـCVSS 4.0-6.9
- منخفض: النتيجة الأساسية لـ CVSS 0.0-3.9

Please note that this notification/advisory has been tagged as TLP ***WHITE*** where information can be shared or published on any public forums.

As part of NCA duties to help securing the cyberspace and protecting national interests, NCA provides the weekly summary of published vulnerabilities by the National Institute of Standards and Technology (NIST) National Vulnerability Database (NVD) for the week from 19th of April to 25th of April. Vulnerabilities are scored using the Common Vulnerability Scoring System (CVSS) standard as per the following severity:

- **Critical: CVSS base score of 9.0-10.0**
- **High: CVSS base score of 7.0-8.9**
- **Medium: CVSS base score 4.0-6.9**
- **Low: CVSS base score 0.0-3.9**

CVE ID & Source	Vendor - Product	Description	Publish Date	CVSS Score
CVE-2026-33819	microsoft - Microsoft Bing	Deserialization of untrusted data in Microsoft Bing allows an unauthorized attacker to execute code over a network.	2026-04-23	10
CVE-2026-35431	microsoft - entra_id	Server-side request forgery (ssrf) in Microsoft Entra ID Entitlement Management allows an unauthorized attacker to perform spoofing over a network.	2026-04-23	10
CVE-2026-21515	microsoft - azure_iot_central	Exposure of sensitive information to an unauthorized actor in Azure IOT Central allows an authorized attacker to elevate privileges over a network.	2026-04-24	9.9
CVE-2026-5450	gnu - glibc	Calling the scanf family of functions with a %mc (malloc'd character match) in the GNU C Library version 2.7 to version 2.43 with a format width specifier with an explicit width greater than 1024 could result in a one byte heap buffer overflow.	2026-04-20	9.8
CVE-2026-6748	mozilla - multiple products	Uninitialized memory in the Audio/Video: Web Codecs component. This vulnerability was fixed in Firefox 150, Firefox ESR 140.10, Thunderbird 150, and Thunderbird 140.10.	2026-04-21	9.8
CVE-2026-6760	mozilla - multiple products	Mitigation bypass in the Networking: Cookies component. This vulnerability was fixed in Firefox 150 and Thunderbird 150.	2026-04-21	9.8
CVE-2026-6768	mozilla - multiple products	Mitigation bypass in the Networking: Cookies component. This vulnerability was fixed in Firefox 150 and Thunderbird 150.	2026-04-21	9.8
CVE-2026-6771	mozilla - multiple products	Mitigation bypass in the DOM: Security component. This vulnerability was fixed in Firefox 150, Firefox ESR 140.10, Thunderbird 150, and Thunderbird 140.10.	2026-04-21	9.8
CVE-2026-34275	oracle - advanced_inbound_telephony	Vulnerability in the Oracle Advanced Inbound Telephony product of Oracle E-Business Suite (component: Setup and Administration). Supported versions that are affected are 12.2.3-12.2.15. Easily exploitable vulnerability allows unauthenticated attacker with network access via HTTP to compromise Oracle Advanced Inbound Telephony. Successful attacks of this vulnerability can result in takeover of Oracle Advanced Inbound Telephony. CVSS 3.1 Base Score 9.8 (Confidentiality, Integrity and Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H).	2026-04-21	9.8
CVE-2026-31436	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: dmaengine: idxd: fix possible wrong descriptor completion in llist_abort_desc() At the end of this function, d is the traversal cursor of flist, but the code completes found instead. This can lead to issues such as NULL pointer dereferences, double completion, or descriptor leaks. Fix this by completing d instead of found in the final list_for_each_entry_safe() loop.	2026-04-22	9.8
CVE-2026-31444	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ksmbd: fix use-after-free and NULL deref in smb_grant_oplock() smb_grant_oplock() has two issues in the oplock publication sequence: 1) opinfo is linked into ci->m_op_list (via opinfo_add) before add_lease_global_list() is called. If add_lease_global_list() fails (kmalloc returns NULL), the error path frees the opinfo via __free_opinfo() while it is still linked in ci->m_op_list. Concurrent m_op_list readers (opinfo_get_list, or direct iteration in smb_break_all_levII_oplock) dereference the freed node.	2026-04-22	9.8

		2) opinfo->o_fp is assigned after add_lease_global_list() publishes the opinfo on the global lease list. A concurrent find_same_lease_key() can walk the lease list and dereference opinfo->o_fp->f_ci while o_fp is still NULL. Fix by restructuring the publication sequence to eliminate post-publish failure: - Set opinfo->o_fp before any list publication (fixes NULL deref). - Preallocate lease_table via alloc_lease_table() before opinfo_add() so add_lease_global_list() becomes infallible after publication. - Keep the original m_op_list publication order (opinfo_add before lease list) so concurrent opens via same_client_has_lease() and opinfo_get_list() still see the in-flight grant. - Use opinfo_put() instead of __free_opinfo() on err_out so that the RCU-deferred free path is used. This also requires splitting add_lease_global_list() to take a preallocated lease_table and changing its return type from int to void, since it can no longer fail.		
CVE-2026-31463	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: iomap: fix invalid folio access when i_blkbits differs from I/O granularity Commit aa35dd5cbc06 ("iomap: fix invalid folio access after folio_end_read()") partially addressed invalid folio access for folios without an ifs attached, but it did not handle the case where 1 << inode->i_blkbits matches the folio size but is different from the granularity used for the IO, which means IO can be submitted for less than the full folio for the !ifs case. In this case, the condition: if (*bytes_submitted == folio_len) ctx->cur_folio = NULL; in iomap_read_folio_iter() will not invalidate ctx->cur_folio, and iomap_read_end() will still be called on the folio even though the IO helper owns it and will finish the read on it. Fix this by unconditionally invalidating ctx->cur_folio for the !ifs case.	2026-04-22	9.8
CVE-2026-31478	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ksmbd: replace hardcoded hdr2_len with offsetof() in smb2_calc_max_out_buf_len() After this commit (e2b76ab8b5c9 "ksmbd: add support for read compound"), response buffer management was changed to use dynamic iov array. In the new design, smb2_calc_max_out_buf_len() expects the second argument (hdr2_len) to be the offset of ->Buffer field in the response structure, not a hardcoded magic number. Fix the remaining call sites to use the correct offsetof() value.	2026-04-22	9.8
CVE-2026-31501	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net: ti: icssg-prueth: fix use-after-free of CPPI descriptor in RX path cppi5_hdesc_get_psdata() returns a pointer into the CPPI descriptor. In both emac_rx_packet() and emac_rx_packet_zc(), the descriptor is freed via k3_cppi_desc_pool_free() before the psdata pointer is used by emac_rx_timestamp(), which dereferences psdata[0] and psdata[1]. This constitutes a use-after-free on every received packet that goes through the timestamp path. Defer the descriptor free until after all accesses through the psdata pointer are complete. For emac_rx_packet(), move the free into the requeue label so both early-exit and success paths free the descriptor after all accesses are done. For emac_rx_packet_zc(), move the free to the end of the loop body after emac_dispatch_skb_zc() (which calls emac_rx_timestamp()) has returned.	2026-04-22	9.8
CVE-2026-31533	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net/tls: fix use-after-free in -EBUSY error path of tls_do_encryption The -EBUSY handling in tls_do_encryption(), introduced by commit 859054147318 ("net: tls: handle backlogging of crypto requests"), has a use-after-free due to double cleanup of encrypt_pending and the scatterlist entry. When crypto_aead_encrypt() returns -EBUSY, the request is enqueued to the cryptd backlog and the async callback tls_encrypt_done() will be	2026-04-23	9.8

		invoked upon completion. That callback unconditionally restores the scatterlist entry (sge->offset, sge->length) and decrements ctx->encrypt_pending. However, if tls_encrypt_async_wait() returns an error, the synchronous error path in tls_do_encryption() performs the same cleanup again, double-decrementing encrypt_pending and double-restoring the scatterlist. The double-decrement corrupts the encrypt_pending sentinel (initialized to 1), making tls_encrypt_async_wait() permanently skip the wait for pending async callbacks. A subsequent sendmsg can then free the tls_rec via bpf_exec_tx_verdict() while a cryptd callback is still pending, resulting in a use-after-free when the callback fires on the freed record. Fix this by skipping the synchronous cleanup when the -EBUSY async wait returns an error, since the callback has already handled encrypt_pending and sge restoration.		
CVE-2026-31536	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: smb: server: let send_done handle a completion without IB_SEND_SIGNALED With smbdirect_send_batch processing we likely have requests without IB_SEND_SIGNALED, which will be destroyed in the final request that has IB_SEND_SIGNALED set. If the connection is broken all requests are signaled even without explicit IB_SEND_SIGNALED.	2026-04-24	9.8
CVE-2026-31589	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: mm: call ->free_folio() directly in folio_unmap_invalidate() We can only call filemap_free_folio() if we have a reference to (or hold a lock on) the mapping. Otherwise, we've already removed the folio from the mapping so it no longer pins the mapping and the mapping can be removed, causing a use-after-free when accessing mapping->a_ops. Follow the same pattern as __remove_mapping() and load the free_folio function pointer before dropping the lock on the mapping. That lets us make filemap_free_folio() static as this was the only caller outside filemap.c.	2026-04-24	9.8
CVE-2026-31607	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: usbip: validate number_of_packets in usbip_pack_ret_submit() When a USB/IP client receives a RET_SUBMIT response, usbip_pack_ret_submit() unconditionally overwrites urb->number_of_packets from the network PDU. This value is subsequently used as the loop bound in usbip_recv_iso() and usbip_pad_iso() to iterate over urb->iso_frame_desc[], a flexible array whose size was fixed at URB allocation time based on the *original* number_of_packets from the CMD_SUBMIT. A malicious USB/IP server can set number_of_packets in the response to a value larger than what was originally submitted, causing a heap out-of-bounds write when usbip_recv_iso() writes to urb->iso_frame_desc[i] beyond the allocated region. KASAN confirmed this with kernel 7.0.0-rc5: BUG: KASAN: slab-out-of-bounds in usbip_recv_iso+0x46a/0x640 Write of size 4 at addr ffff888106351d40 by task vhci_rx/69 The buggy address is located 0 bytes to the right of allocated 320-byte region [ffff888106351c00, ffff888106351d40) The server side (stub_rx.c) and gadget side (vudc_rx.c) already validate number_of_packets in the CMD_SUBMIT path since commits c6688ef9f297 ("usbip: fix stub_rx: harden CMD_SUBMIT path to handle malicious input") and b78d830f0049 ("usbip: fix vudc_rx: harden CMD_SUBMIT path to handle malicious input"). The server side validates against USBIP_MAX_ISO_PACKETS because no URB exists yet at that point. On the client side we have the original URB, so we can use the tighter bound: the response must not exceed the original number_of_packets. This mirrors the existing validation of actual_length against transfer_buffer_length in usbip_recv_xbuff(), which checks the response value against the original allocation size. Kelvin Mbogo's series ("usb: usbip: fix integer overflow in usbip_recv_iso()", v2) hardens the receive-side functions themselves;	2026-04-24	9.8

		this patch complements that work by catching the bad value at its source -- in usbip_pack_ret_submit() before the overwrite -- and using the tighter per-URB allocation bound rather than the global USBIP_MAX_ISO_PACKETS limit. Fix this by checking rpdu->number_of_packets against urb->number_of_packets in usbip_pack_ret_submit() before the overwrite. On violation, clamp to zero so that usbip_recv_iso() and usbip_pad_iso() safely return early.		
CVE-2026-31608	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: smb: server: avoid double-free in smb_direct_free_sendmsg after smb_direct_flush_send_list() smb_direct_flush_send_list() already calls smb_direct_free_sendmsg(), so we should not call it again after post_sendmsg() moved it to the batch list.	2026-04-24	9.8
CVE-2026-31609	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: smb: client: avoid double-free in smbd_free_send_io() after smbd_send_batch_flush() smbd_send_batch_flush() already calls smbd_free_send_io(), so we should not call it again after smbd_post_send() moved it to the batch list.	2026-04-24	9.8
CVE-2026-31633	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: rxrpc: Fix integer overflow in rxgk_verify_response() In rxgk_verify_response(), there's a potential integer overflow due to rounding up token_len before checking it, thereby allowing the length check to be bypassed. Fix this by checking the unrounded value against len too (len is limited as the response must fit in a single UDP packet).	2026-04-24	9.8
CVE-2026-31637	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: rxrpc: reject undecryptable rxkad response tickets rxkad_decrypt_ticket() decrypts the RXKAD response ticket and then parses the buffer as plaintext without checking whether crypto_skcipher_decrypt() succeeded. A malformed RESPONSE can therefore use a non-block-aligned ticket length, make the decrypt operation fail, and still drive the ticket parser with attacker-controlled bytes. Check the decrypt result and abort the connection with RXKADBADTICKET when ticket decryption fails.	2026-04-24	9.8
CVE-2026-31649	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net: stmmac: fix integer underflow in chain mode The jumbo_frm() chain-mode implementation unconditionally computes len = nopaged_len - bmax; where nopaged_len = skb_headlen(skb) (linear bytes only) and bmax is BUF_SIZE_8KiB or BUF_SIZE_2KiB. However, the caller stmmac_xmit() decides to invoke jumbo_frm() based on skb->len (total length including page fragments): is_jumbo = stmmac_is_jumbo_frm(priv, skb->len, enh_desc); When a packet has a small linear portion (nopaged_len <= bmax) but a large total length due to page fragments (skb->len > bmax), the subtraction wraps as an unsigned integer, producing a huge len value (~0xFFFFxxxx). This causes the while (len != 0) loop to execute hundreds of thousands of iterations, passing skb->data + bmax * i pointers far beyond the skb buffer to dma_map_single(). On IOMMU-less SoCs (the typical deployment for stmmac), this maps arbitrary kernel memory to the DMA engine, constituting a kernel memory disclosure and potential memory corruption from hardware. Fix this by introducing a buf_len local variable clamped to min(nopaged_len, bmax). Computing len = nopaged_len - buf_len is then always safe: it is zero when the linear portion fits within a single descriptor, causing the while (len != 0) loop to be skipped naturally, and the fragment loop in stmmac_xmit() handles page fragments afterward.	2026-04-24	9.8
CVE-2026-31657	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: batman-adv: hold claim backbone gateways by reference	2026-04-24	9.8

		batadv_bla_add_claim() can replace claim->backbone_gw and drop the old gateway's last reference while readers still follow the pointer. The netlink claim dump path dereferences claim->backbone_gw->orig and takes claim->backbone_gw->crc_lock without pinning the underlying backbone gateway. batadv_bla_check_claim() still has the same naked pointer access pattern. Reuse batadv_bla_claim_get_backbone_gw() in both readers so they operate on a stable gateway reference until the read-side work is complete. This keeps the dump and claim-check paths aligned with the lifetime rules introduced for the other BLA claim readers.		
CVE-2026-31659	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: batman-adv: reject oversized global TT response buffers batadv_tt_prepare_tvlv_global_data() builds the allocation length for a global TT response in 16-bit temporaries. When a remote originator advertises a large enough global TT, the TT payload length plus the VLAN header offset can exceed 65535 and wrap before kmalloc(). The full-table response path still uses the original TT payload length when it fills tt_change, so the wrapped allocation is too small and batadv_tt_prepare_tvlv_global_data() writes past the end of the heap object before the later packet-size check runs. Fix this by rejecting TT responses whose TVLV value length cannot fit in the 16-bit TVLV payload length field.	2026-04-24	9.8
CVE-2026-31668	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: seg6: separate dst_cache for input and output paths in seg6 lwtunnel The seg6 lwtunnel uses a single dst_cache per encap route, shared between seg6_input_core() and seg6_output_core(). These two paths can perform the post-encap SID lookup in different routing contexts (e.g., ip rules matching on the ingress interface, or VRF table separation). Whichever path runs first populates the cache, and the other reuses it blindly, bypassing its own lookup. Fix this by splitting the cache into cache_input and cache_output, so each path maintains its own cached dst independently.	2026-04-24	9.8
CVE-2026-31669	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: mptcp: fix slab-use-after-free in __inet_lookup_established The ehash table lookups are lockless and rely on SLAB_TYPESAFE_BY_RCU to guarantee socket memory stability during RCU read-side critical sections. Both tcp_prot and tcpv6_prot have their slab caches created with this flag via proto_register(). However, MPTCP's mptcp_subflow_init() copies tcpv6_prot into tcpv6_prot_override during inet_init() (fs_initcall, level 5), before inet6_init() (module_init/device_initcall, level 6) has called proto_register(&tcpv6_prot). At that point, tcpv6_prot.slab is still NULL, so tcpv6_prot_override.slab remains NULL permanently. This causes MPTCP v6 subflow child sockets to be allocated via kmalloc (falling into kmalloc-4k) instead of the TCPv6 slab cache. The kmalloc-4k cache lacks SLAB_TYPESAFE_BY_RCU, so when these sockets are freed without SOCK_RCU_FREE (which is cleared for child sockets by design), the memory can be immediately reused. Concurrent ehash lookups under rcu_read_lock can then access freed memory, triggering a slab-use-after-free in __inet_lookup_established. Fix this by splitting the IPv6-specific initialization out of mptcp_subflow_init() into a new mptcp_subflow_v6_init(), called from mptcp_proto_v6_init() before protocol registration. This ensures tcpv6_prot_override.slab correctly inherits the SLAB_TYPESAFE_BY_RCU slab cache.	2026-04-24	9.8
CVE-2026-6919	google - chrome	Use after free in DevTools in Google Chrome prior to 147.0.7727.117 allowed a remote attacker who had compromised the renderer process to potentially perform a sandbox escape via a crafted HTML page. (Chromium security severity: High)	2026-04-23	9.6
CVE-2026-6920	google - chrome	Out of bounds read in GPU in Google Chrome on Android prior to 147.0.7727.117 allowed a remote attacker who had compromised the renderer process to potentially perform a sandbox escape via a crafted HTML page. (Chromium security severity: High)	2026-04-23	9.6

CVE-2026-24303	microsoft - partner_center	Improper access control in Microsoft Partner Center allows an authorized attacker to elevate privileges over a network.	2026-04-23	9.6
CVE-2026-21571	atlassian - Bamboo Data Center	This Critical severity OS Command Injection vulnerability was introduced in versions 9.6.0, 10.0.0, 10.1.0, 10.2.0, 11.0.0, 11.1.0, 12.0.0, and 12.1.0 of Bamboo Data Center. This RCE (Remote Code Execution) vulnerability, with a CVSS Score of 9.4 and a CVSS Vector of CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:H/SI:H/SA:H allows an authenticated attacker to execute commands on the remote system, which has high impact to confidentiality, high impact to integrity, high impact to availability, and requires no user interaction. Atlassian recommends that Bamboo Data Center customers upgrade to latest version, if you are unable to do so, upgrade your instance to one of the specified supported fixed versions: Bamboo Data Center 9.6.0: Upgrade to a release greater than or equal to 9.6.25 Bamboo Data Center 10.2: Upgrade to a release greater than or equal to 10.2.18 Bamboo Data Center 12.1: Upgrade to a release greater than or equal to 12.1.6 See the release notes ([https://confluence.atlassian.com/bambooreleases/bamboo-release-notes-1189793869.html]). You can download the latest version of Bamboo Data Center from the download center ([https://www.atlassian.com/software/bamboo/download-archives]).	2026-04-21	9.4
CVE-2026-31448	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ext4: avoid infinite loops caused by residual data On the mkdir/mknod path, when mapping logical blocks to physical blocks, if inserting a new extent into the extent tree fails (in this example, because the file system disabled the huge file feature when marking the inode as dirty), ext4_ext_map_blocks() only calls ext4_free_blocks() to reclaim the physical block without deleting the corresponding data in the extent tree. This causes subsequent mkdir operations to reference the previously reclaimed physical block number again, even though this physical block is already being used by the xattr block. Therefore, a situation arises where both the directory and xattr are using the same buffer head block in memory simultaneously. The above causes ext4_xattr_block_set() to enter an infinite loop about "inserted" and cannot release the inode lock, ultimately leading to the 143s blocking problem mentioned in [1]. If the metadata is corrupted, then trying to remove some extent space can do even more harm. Also in case EXT4_GET_BLOCKS_DELALLOC_RESERVE was passed, remove space wrongly update quota information. Jan Kara suggests distinguishing between two cases: 1) The error is ENOSPC or EDQUOT - in this case the filesystem is fully consistent and we must maintain its consistency including all the accounting. However these errors can happen only early before we've inserted the extent into the extent tree. So current code works correctly for this case. 2) Some other error - this means metadata is corrupted. We should strive to do as few modifications as possible to limit damage. So I'd just skip freeing of allocated blocks. [1] INFO: task syz.0.17:5995 blocked for more than 143 seconds. Call Trace: inode_lock_nested include/linux/fs.h:1073 [inline] __start_dirop fs/namei.c:2923 [inline] start_dirop fs/namei.c:2934 [inline]	2026-04-22	9.4
CVE-2026-31685	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: netfilter: ip6t_eui64: reject invalid MAC header for all packets `eui64_mt6()` derives a modified EUI-64 from the Ethernet source address and compares it with the low 64 bits of the IPv6 source address. The existing guard only rejects an invalid MAC header when `par->fragoff != 0`. For packets with `par->fragoff == 0`, `eui64_mt6()` can still reach `eth_hdr(skb)` even when the MAC header is not valid. Fix this by removing the `par->fragoff != 0` condition so that packets with an invalid MAC header are rejected before accessing `eth_hdr(skb)`.	2026-04-25	9.4
CVE-2026-32210	microsoft - Microsoft Dynamics 365 (online)	Server-side request forgery (ssrf) in Microsoft Dynamics 365 (Online) allows an unauthorized attacker to perform spoofing over a network.	2026-04-23	9.3

CVE-2026-33102	microsoft - 365_copilot	Url redirection to untrusted site ('open redirect') in M365 Copilot allows an unauthorized attacker to elevate privileges over a network.	2026-04-23	9.3
CVE-2026-33557	apache - kafka	A possible security vulnerability has been identified in Apache Kafka. By default, the broker property `sasl.oauthbearer.jwt.validator.class` is set to `org.apache.kafka.common.security.oauthbearer.DefaultJwtValidator`. It accepts any JWT token without validating its signature, issuer, or audience. An attacker can generate a JWT token from any issuer with the `preferred_username` set to any user, and the broker will accept it. We advise the Kafka users using kafka v4.1.0 or v4.1.1 to set the config `sasl.oauthbearer.jwt.validator.class` to `org.apache.kafka.common.security.oauthbearer.BrokerJwtValidator` explicitly to avoid this vulnerability. Since Kafka v4.1.2 and v4.2.0 and later, the issue is fixed and will correctly validate the JWT token.	2026-04-20	9.1
CVE-2026-40372	microsoft - asp.net_core	Improper verification of cryptographic signature in ASP.NET Core allows an unauthorized attacker to elevate privileges over a network.	2026-04-21	9.1
CVE-2026-34279	oracle - multiple products	Vulnerability in the Oracle Enterprise Manager Base Platform product of Oracle Enterprise Manager (component: Event Management). Supported versions that are affected are 13.5 and 24.1. Easily exploitable vulnerability allows high privileged attacker with network access via HTTP to compromise Oracle Enterprise Manager Base Platform. While the vulnerability is in Oracle Enterprise Manager Base Platform, attacks may significantly impact additional products (scope change). Successful attacks of this vulnerability can result in takeover of Oracle Enterprise Manager Base Platform. CVSS 3.1 Base Score 9.1 (Confidentiality, Integrity and Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:C/C:H/I:H/A:H).	2026-04-21	9.1
CVE-2026-34285	oracle - identity_manager _connector	Vulnerability in the Oracle Identity Manager Connector product of Oracle Fusion Middleware (component: Core). The supported version that is affected is 12.2.1.4.0. Easily exploitable vulnerability allows unauthenticated attacker with network access via HTTPS to compromise Oracle Identity Manager Connector. Successful attacks of this vulnerability can result in unauthorized creation, deletion or modification access to critical data or all Oracle Identity Manager Connector accessible data as well as unauthorized access to critical data or complete access to all Oracle Identity Manager Connector accessible data. CVSS 3.1 Base Score 9.1 (Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:N).	2026-04-21	9.1
CVE-2026-34286	oracle - identity_manager _connector	Vulnerability in the Oracle Identity Manager Connector product of Oracle Fusion Middleware (component: Core). The supported version that is affected is 12.2.1.4.0. Easily exploitable vulnerability allows unauthenticated attacker with network access via HTTPS to compromise Oracle Identity Manager Connector. Successful attacks of this vulnerability can result in unauthorized creation, deletion or modification access to critical data or all Oracle Identity Manager Connector accessible data as well as unauthorized access to critical data or complete access to all Oracle Identity Manager Connector accessible data. CVSS 3.1 Base Score 9.1 (Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:N).	2026-04-21	9.1
CVE-2026-34287	oracle - identity_manager _connector	Vulnerability in the Oracle Identity Manager Connector product of Oracle Fusion Middleware (component: Core). The supported version that is affected is 12.2.1.4.0. Easily exploitable vulnerability allows unauthenticated attacker with network access via HTTPS to compromise Oracle Identity Manager Connector. Successful attacks of this vulnerability can result in unauthorized creation, deletion or modification access to critical data or all Oracle Identity Manager Connector accessible data as well as unauthorized access to critical data or complete access to all Oracle Identity Manager Connector accessible data. CVSS 3.1 Base Score 9.1 (Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:N).	2026-04-21	9.1
CVE-2026-31636	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: rxrpc: fix RESPONSE authenticator parser OOB read rxgk_verify_authenticator() copies auth_len bytes into a temporary buffer and then passes p + auth_len as the parser limit to rxgk_do_verify_authenticator(). Since p is a __be32 *, that inflates the parser end pointer by a factor of four and lets malformed RESPONSE authenticators read past the kmalloc() buffer. Decoded from the original latest-net reproduction logs with scripts/decode_stacktrace.sh: BUG: KASAN: slab-out-of-bounds in rxgk_verify_response() Call Trace: dump_stack_lvl() [lib/dump_stack.c:123] print_report() [mm/kasan/report.c:379 mm/kasan/report.c:482] kasan_report() [mm/kasan/report.c:597] rxgk_verify_response() [net/rxrpc/rxgk.c:1103 net/rxrpc/rxgk.c:1167 net/rxrpc/rxgk.c:1274] rxrpc_process_connection() [net/rxrpc/conn_event.c:266 net/rxrpc/conn_event.c:364 net/rxrpc/conn_event.c:386] process_one_work() [kernel/workqueue.c:3281] worker_thread() [kernel/workqueue.c:3353 kernel/workqueue.c:3440] kthread() [kernel/kthread.c:436] ret_from_fork() [arch/x86/kernel/process.c:164] Allocated by task 54: rxgk_verify_response()	2026-04-24	9.1

		[include/linux/slab.h:954 net/rxrpc/rxgk.c:1155 net/rxrpc/rxgk.c:1274] rxrpc_process_connection() [net/rxrpc/conn_event.c:266 net/rxrpc/conn_event.c:364 net/rxrpc/conn_event.c:386] Convert the byte count to __be32 units before constructing the parser limit.		
CVE-2026-31682	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: bridge: br_nd_send: linearize skb before parsing ND options br_nd_send() parses neighbour discovery options from ns->opt[] and assumes that these options are in the linear part of request. Its callers only guarantee that the ICMPv6 header and target address are available, so the option area can still be non-linear. Parsing ns->opt[] in that case can access data past the linear buffer. Linearize request before option parsing and derive ns from the linear network header.	2026-04-25	9.1
CVE-2026-26944	dell - multiple products	Dell PowerProtect Data Domain, versions 7.7.1.0 through 8.6, LTS2025 release version 8.3.1.0 through 8.3.1.20, LTS2024 release versions 7.13.1.0 through 7.13.1.60 contain a missing authentication for critical function vulnerability. An unauthenticated attacker with remote access could potentially exploit this vulnerability, leading to arbitrary command execution with root privileges. Exploitation requires an authenticated user to perform a specific action.	2026-04-20	8.8
CVE-2026-6750	mozilla - multiple products	Privilege escalation in the Graphics: WebRender component. This vulnerability was fixed in Firefox 150, Firefox ESR 115.35, Firefox ESR 140.10, Thunderbird 150, and Thunderbird 140.10.	2026-04-21	8.8
CVE-2026-6761	mozilla - multiple products	Privilege escalation in the Networking component. This vulnerability was fixed in Firefox 150, Firefox ESR 140.10, Thunderbird 150, and Thunderbird 140.10.	2026-04-21	8.8
CVE-2026-6769	mozilla - multiple products	Privilege escalation in the Debugger component. This vulnerability was fixed in Firefox 150, Firefox ESR 140.10, Thunderbird 150, and Thunderbird 140.10.	2026-04-21	8.8
CVE-2026-31432	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ksmbd: fix OOB write in QUERY_INFO for compound requests When a compound request such as READ + QUERY_INFO(Security) is received, and the first command (READ) consumes most of the response buffer, ksmbd could write beyond the allocated buffer while building a security descriptor. The root cause was that smb2_get_info_sec() checked buffer space using ppntsd_size from xattr, while build_sec_desc() often synthesized a significantly larger descriptor from POSIX ACLs. This patch introduces smb_acl_sec_desc_scratch_len() to accurately compute the final descriptor size beforehand, performs proper buffer checking with smb2_calc_max_out_buf_len(), and uses exact-sized allocation + iov pinning.	2026-04-22	8.8
CVE-2026-31433	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ksmbd: fix potencial OOB in get_file_all_info() for compound requests When a compound request consists of QUERY_DIRECTORY + QUERY_INFO (FILE_ALL_INFORMATION) and the first command consumes nearly the entire max_trans_size, get_file_all_info() would blindly call smbConvertToUTF16() with PATH_MAX, causing out-of-bounds write beyond the response buffer. In get_file_all_info(), there was a missing validation check for the client-provided OutputBufferLength before copying the filename into FileName field of the smb2_file_all_info structure. If the filename length exceeds the available buffer space, it could lead to potential buffer overflows or memory corruption during smbConvertToUTF16 conversion. This calculating the actual free buffer size using smb2_calc_max_out_buf_len() and returning -EINVAL if the buffer is insufficient and updating smbConvertToUTF16 to use the actual filename length (clamped by PATH_MAX) to ensure a safe copy operation.	2026-04-22	8.8
CVE-2026-31435	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: netfs: Fix read abandonment during retry Under certain circumstances, all the remaining subrequests from a read request will get abandoned during retry. The abandonment process expects the 'subreq' variable to be set to the place to start abandonment from, but it doesn't always have a useful value (it will be uninitialised on the first pass through the loop and it may point to a deleted subrequest on later passes). Fix the first jump to "abandon:" to set subreq to the start of the first subrequest expected to need retry (which, in this abandonment case, turned out unexpectedly to no longer have NEED_RETRY set).	2026-04-22	8.8

		Also clear the subreq pointer after discarding superfluous retryable subrequests to cause an oops if we do try to access it.		
CVE-2026-31450	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ext4: publish jinode after initialization ext4_inode_attach_jinode() publishes ei->jinode to concurrent users. It used to set ei->jinode before jbd2_journal_init_jbd_inode(), allowing a reader to observe a non-NULL jinode with i_vfs_inode still unset. The fast commit flush path can then pass this jinode to jbd2_wait_inode_data(), which dereferences i_vfs_inode->i_mapping and may crash. Below is the crash I observe: ``` BUG: unable to handle page fault for address: 000000010beb47f4 PGD 110e51067 P4D 110e51067 PUD 0 Oops: Oops: 0000 [#1] SMP NOPTI CPU: 1 UID: 0 PID: 4850 Comm: fc_fsync_bench_ Not tainted 6.18.0-00764-g795a690c06a5 #1 PREEMPT(voluntary) Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS Arch Linux 1.17.0-2-2 04/01/2014 RIP: 0010:xas_find_marked+0x3d/0x2e0 Code: e0 03 48 83 f8 02 0f 84 f0 01 00 00 48 8b 47 08 48 89 c3 48 39 c6 0f 82 fd 01 00 00 48 85 c9 74 3d 48 83 f9 03 77 63 4c 8b 0f <49> 8b 71 08 48 c7 47 18 00 00 00 00 48 89 f1 83 e1 03 48 83 f9 02 RSP: 0018:ffffbbee806e7bf0 EFLAGS: 00010246 RAX: 000000000010beb4 RBX: 000000000010beb4 RCX: 0000000000000003 RDX: 0000000000000001 RSI: 0000002000300000 RDI: ffffbbee806e7c10 RBP: 0000000000000001 R08: 0000002000300000 R09: 000000010beb47ec R10: ffff9ea494590090 R11: 0000000000000000 R12: 0000002000300000 R13: ffffbbee806e7c90 R14: ffff9ea494513788 R15: ffffbbee806e7c88 FS: 00007fc2f9e3e6c0(0000) GS:ffff9ea6b1444000(0000) knlGS:0000000000000000 CS: 0010 DS: 0000 ES: 0000 CR0: 0000000080050033 CR2: 000000010beb47f4 CR3: 0000000119ac5000 CR4: 0000000000750ef0 PKRU: 55555554 Call Trace: <TASK> filemap_get_folios_tag+0x87/0x2a0 __filemap_fdatawait_range+0x5f/0xd0 ? srso_alias_return_thunk+0x5/0xfbef5 ? __schedule+0x3e7/0x10c0 ? srso_alias_return_thunk+0x5/0xfbef5 ? srso_alias_return_thunk+0x5/0xfbef5 ? srso_alias_return_thunk+0x5/0xfbef5 ? preempt_count_sub+0x5f/0x80 ? srso_alias_return_thunk+0x5/0xfbef5 ? cap_safe_nice+0x37/0x70 ? srso_alias_return_thunk+0x5/0xfbef5 ? preempt_count_sub+0x5f/0x80 ? srso_alias_return_thunk+0x5/0xfbef5 filemap_fdatawait_range_keep_errors+0x12/0x40 ext4_fc_commit+0x697/0x8b0 ? ext4_file_write_iter+0x64b/0x950 ? srso_alias_return_thunk+0x5/0xfbef5 ? preempt_count_sub+0x5f/0x80 ? srso_alias_return_thunk+0x5/0xfbef5 ? vfs_write+0x356/0x480 ? srso_alias_return_thunk+0x5/0xfbef5 ? preempt_count_sub+0x5f/0x80 ext4_sync_file+0xf7/0x370 do_fsync+0x3b/0x80 ? syscall_trace_enter+0x108/0x1d0 __x64_sys_fdatasync+0x16/0x20 do_syscall_64+0x62/0x2c0 entry_SYSCALL_64_after_hwframe+0x76/0x7e ... ``` Fix this by initializing the jbd2_inode first. Use smp_wmb() and WRITE_ONCE() to publish ei->jinode after initialization. Readers use READ_ONCE() to fetch the pointer.	2026-04-22	8.8
CVE-2026-6859	red hat - multiple products	A flaw was found in InstructLab. The `linux_train.py` script hardcodes `trust_remote_code=True` when loading models from HuggingFace. This allows a remote attacker to achieve arbitrary Python code execution by convincing a user to run `ilab train/download/generate` with a specially crafted malicious model from the HuggingFace Hub. This vulnerability can lead to complete system compromise.	2026-04-22	8.8

CVE-2026-40466	apache - multiple products	Improper Input Validation, Improper Control of Generation of Code ('Code Injection') vulnerability in Apache ActiveMQ Broker, Apache ActiveMQ All, Apache ActiveMQ. An authenticated attacker may bypass the fix in CVE-2026-34197 by adding a connector using an HTTP Discovery transport via BrokerView.addNetworkConnector or BrokerView.addConnector through Jolokia if the activemq-http module is on the classpath. A malicious HTTP endpoint can return a VM transport through the HTTP URI which will bypass the validation added in CVE-2026-34197. The attacker can then use the VM transport's brokerConfig parameter to load a remote Spring XML application context using ResourceXmlApplicationContext. Because Spring's ResourceXmlApplicationContext instantiates all singleton beans before the BrokerService validates the configuration, arbitrary code execution occurs on the broker's JVM through bean factory methods such as Runtime.exec(). This issue affects Apache ActiveMQ Broker: before 5.19.6, from 6.0.0 before 6.2.5; Apache ActiveMQ All: before 5.19.6, from 6.0.0 before 6.2.5; Apache ActiveMQ: before 5.19.6, from 6.0.0 before 6.2.5. Users are recommended to upgrade to version 5.19.6 or 6.2.5, which fixes the issue.	2026-04-24	8.8
CVE-2026-41044	apache - multiple products	Improper Input Validation, Improper Control of Generation of Code ('Code Injection') vulnerability in Apache ActiveMQ, Apache ActiveMQ Broker, Apache ActiveMQ All. An authenticated attacker can use the admin web console page to construct a malicious broker name that bypasses name validation to include an xbean binding that can be later used by a VM transport to load a remote Spring XML application. The attacker can then use the DestinationView mbean to send a message to trigger a VM transport creation that will reference this malicious broker name which can lead to loading the malicious Spring XML context file. Because Spring's ResourceXmlApplicationContext instantiates all singleton beans before the BrokerService validates the configuration, arbitrary code execution occurs on the broker's JVM through bean factory methods such as Runtime.exec(). This issue affects Apache ActiveMQ: before 5.19.6, from 6.0.0 before 6.2.5; Apache ActiveMQ Broker: before 5.19.6, from 6.0.0 before 6.2.5; Apache ActiveMQ All: before 5.19.6, from 6.0.0 before 6.2.5. Users are recommended to upgrade to version 6.2.5 or 5.19.6, which fixes the issue.	2026-04-24	8.8
CVE-2026-31553	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: KVM: arm64: Fix the descriptor address in __kvm_at_swap_desc() Using "(u64 __user *)hva + offset" to get the virtual addresses of S1/S2 descriptors looks really wrong, if offset is not zero. What we want to get for swapping is hva + offset, not hva + offset*8. ;-) Fix it.	2026-04-24	8.8
CVE-2026-31558	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: LoongArch: KVM: Make kvm_get_vcpu_by_cpuid() more robust kvm_get_vcpu_by_cpuid() takes a cpuid parameter whose type is int, so cpuid can be negative. Let kvm_get_vcpu_by_cpuid() return NULL for this case so as to make it more robust. This fix an out-of-bounds access to kvm_arch::phyid_map::phys_map[].	2026-04-24	8.8
CVE-2026-31570	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: can: gw: fix OOB heap access in cgw_csum_crc8_rel() cgw_csum_crc8_rel() correctly computes bounds-safe indices via calc_idx(): int from = calc_idx(crc8->from_idx, cf->len); int to = calc_idx(crc8->to_idx, cf->len); int res = calc_idx(crc8->result_idx, cf->len); if (from < 0 || to < 0 || res < 0) return; However, the loop and the result write then use the raw s8 fields directly instead of the computed variables: for (i = crc8->from_idx; ...) /* BUG: raw negative index */ cf->data[crc8->result_idx] = ...; /* BUG: raw negative index */ With from_idx = to_idx = result_idx = -64 on a 64-byte CAN FD frame, calc_idx(-64, 64) = 0 so the guard passes, but the loop iterates with	2026-04-24	8.8

		i = -64, reading cf->data[-64], and the write goes to cf->data[-64]. This write might end up to 56 (7.0-rc) or 40 (<= 6.19) bytes before the start of the canfd_frame on the heap. The companion function cgw_csum_xor_rel() uses `from`/`to`/`res` correctly throughout; fix cgw_csum_crc8_rel() to match. Confirmed with KASAN on linux-7.0-rc2: BUG: KASAN: slab-out-of-bounds in cgw_csum_crc8_rel+0x515/0x5b0 Read of size 1 at addr ffff8880076619c8 by task poc_cgw_oob/62 To configure the can-gw crc8 checksums CAP_NET_ADMIN is needed.		
CVE-2026-31588	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: KVM: x86: Use scratch field in MMIO fragment to hold small write values When exiting to userspace to service an emulated MMIO write, copy the to-be-written value to a scratch field in the MMIO fragment if the size of the data payload is 8 bytes or less, i.e. can fit in a single chunk, instead of pointing the fragment directly at the source value. This fixes a class of use-after-free bugs that occur when the emulator initiates a write using an on-stack, local variable as the source, the write splits a page boundary, *and* both pages are MMIO pages. Because KVM's ABI only allows for physically contiguous MMIO requests, accesses that split MMIO pages are separated into two fragments, and are sent to userspace one at a time. When KVM attempts to complete userspace MMIO in response to KVM_RUN after the first fragment, KVM will detect the second fragment and generate a second userspace exit, and reference the on-stack variable. The issue is most visible if the second KVM_RUN is performed by a separate task, in which case the stack of the initiating task can show up as truly freed data. ========== BUG: KASAN: use-after-free in complete_emulated_mmio+0x305/0x420 Read of size 1 at addr ffff888009c378d1 by task syz-executor417/984 CPU: 1 PID: 984 Comm: syz-executor417 Not tainted 5.10.0-182.0.0.95.h2627.eulerosv2r13.x86_64 #3 Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS rel-1.15.0-0-g2dd4b9b3f840- prebuilt.qemu.org 04/01/2014 Call Trace: dump_stack+0xbe/0xfd print_address_description.constprop.0+0x19/0x170 __kasan_report.cold+0x6c/0x84 kasan_report+0x3a/0x50 check_memory_region+0xfd/0x1f0 memcpy+0x20/0x60 complete_emulated_mmio+0x305/0x420 kvm_arch_vcpu_ioctl_run+0x63f/0x6d0 kvm_vcpu_ioctl+0x413/0xb20 __se_sys_ioctl+0x111/0x160 do_syscall_64+0x30/0x40 entry_SYSCALL_64_after_hwframe+0x67/0xd1 RIP: 0033:0x42477d Code: <48> 3d 01 f0 ff ff 73 01 c3 48 c7 c1 b0 ff ff ff f7 d8 64 89 01 48 RSP: 002b:00007faa8e6890e8 EFLAGS: 00000246 ORIG_RAX: 0000000000000010 RAX: fffffffffffffda RBX: 00000000004d7338 RCX: 000000000042477d RDX: 0000000000000000 RSI: 000000000000ae80 RDI: 0000000000000005 RBP: 00000000004d7330 R08: 00007fff28d546df R09: 0000000000000000 R10: 0000000000000000 R11: 0000000000000246 R12: 00000000004d733c R13: 0000000000000000 R14: 000000000040a200 R15: 00007fff28d54720 The buggy address belongs to the page: page:0000000029f6a428 refcount:0 mapcount:0 mapping:0000000000000000 index:0x0 pfn:0x9c37 flags: 0xfffffc0000000(node=0|zone=1|lastcpupid=0x1fffff) raw: 000fffffc0000000 0000000000000000 ffffea0000270dc8 0000000000000000 raw: 0000000000000000 0000000000000000 00000000ffffffff 0000000000000000 page dumped because: kasan: bad access detected Memory state around the buggy address: ffff888009c37780: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ffff888009c37800: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff >ffff888009c37880: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ^ ffff888009c37900: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ffff888009c37980: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ==========	2026-04-24	8.8

		The bug can also be reproduced with a targeted KVM-Unit-Test by hacking KVM to fill a large on-stack variable in complete_emulated_mmio(), i.e. by overwrite the data value with garbage. Limit the use of the scratch fields to 8-byte or smaller accesses, and to just writes, as larger accesses and reads are not affected thanks to implementation details in the emulator, but add a sanity check to ensure those details don't change in the future. Specifically, KVM never uses on-stack variables for accesses larger that 8 bytes, e.g. uses an operand in the emulator context, and *al ---truncated---		
CVE-2026-31622	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: NFC: digital: Bounds check NFC-A cascade depth in SDD response handler The NFC-A anti-collision cascade in digital_in_recv_sdd_res() appends 3 or 4 bytes to target->nfcid1 on each round, but the number of cascade rounds is controlled entirely by the peer device. The peer sets the cascade tag in the SDD_RES (deciding 3 vs 4 bytes) and the cascade-incomplete bit in the SEL_RES (deciding whether another round follows). ISO 14443-3 limits NFC-A to three cascade levels and target->nfcid1 is sized accordingly (NFC_NFCID1_MAXSIZE = 10), but nothing in the driver actually enforces this. This means a malicious peer can keep the cascade running, writing past the heap-allocated nfc_target with each round. Fix this by rejecting the response when the accumulated UID would exceed the buffer. Commit e329e71013c9 ("NFC: nci: Bounds check struct nfc_target arrays") fixed similar missing checks against the same field on the NCI path.	2026-04-24	8.8
CVE-2026-31629	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: nfc: llcp: add missing return after LLCP_CLOSED checks In nfc_llcp_recv_hdlc() and nfc_llcp_recv_disc(), when the socket state is LLCP_CLOSED, the code correctly calls release_sock() and nfc_llcp_sock_put() but fails to return. Execution falls through to the remainder of the function, which calls release_sock() and nfc_llcp_sock_put() again. This results in a double release_sock() and a refcount underflow via double nfc_llcp_sock_put(), leading to a use-after-free. Add the missing return statements after the LLCP_CLOSED branches in both functions to prevent the fall-through.	2026-04-24	8.8
CVE-2026-34291	oracle - multiple products	Vulnerability in the Oracle HTTP Server product of Oracle Fusion Middleware (component: Core). Supported versions that are affected are 12.2.1.4.0 and 14.1.2.0.0. Difficult to exploit vulnerability allows unauthenticated attacker with network access via HTTP to compromise Oracle HTTP Server. While the vulnerability is in Oracle HTTP Server, attacks may significantly impact additional products (scope change). Successful attacks of this vulnerability can result in unauthorized creation, deletion or modification access to critical data or all Oracle HTTP Server accessible data as well as unauthorized access to critical data or complete access to all Oracle HTTP Server accessible data. CVSS 3.1 Base Score 8.7 (Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:C/C:H/I:H/A:N).	2026-04-21	8.7
CVE-2026-6947	d-link - DWM-222W	DWM-222W USB Wi-Fi Adapter developed by D-Link has a Brute-Force Protection Bypass vulnerability, allowing unauthenticated adjacent network attackers to bypass login attempt limits to perform brute-force attacks to gain control over the device.	2026-04-24	8.7
CVE-2026-26150	microsoft - purview_ediscovery	Server-side request forgery (ssrf) in Microsoft Purview allows an unauthorized attacker to elevate privileges over a network.	2026-04-23	8.6
CVE-2026-5367	red hat - multiple products	A flaw was found in OVN (Open Virtual Network). A remote attacker, by sending crafted DHCPv6 (Dynamic Host Configuration Protocol for IPv6) SOLICIT packets with an inflated Client ID length, could cause the ovn-controller to read beyond the bounds of a packet. This out-of-bounds read can lead to the disclosure of sensitive information stored in heap memory, which is then returned to the attacker's virtual machine port.	2026-04-24	8.6
CVE-2026-31611	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ksmbd: require 3 sub-authorities before reading sub_auth[2] parse_dacl() compares each ACE SID against sid_unix_NFS_mode and on match reads sid.sub_auth[2] as the file mode. If sid_unix_NFS_mode is the prefix S-1-5-88-3 with num_subauth = 2 then compare_sids() compares only min(num_subauth, 2) sub-authorities so a client SID with num_subauth = 2 and sub_auth = {88, 3} will match. If num_subauth = 2 and the ACE is placed at the very end of the security descriptor, sub_auth[2] will be 4 bytes past end_of_acl. The	2026-04-24	8.6

		out-of-band bytes will then be masked to the low 9 bits and applied as the file's POSIX mode, probably not something that is good to have happen. Fix this up by forcing the SID to actually carry a third sub-authority before reading it at all.		
CVE-2026-21997	oracle - life_sciences_empirica_signal	Vulnerability in the Oracle Life Sciences Empirica Signal product of Oracle Life Science Applications (component: Common Core). Supported versions that are affected are 9.2.1-9.2.3. Easily exploitable vulnerability allows low privileged attacker with network access via HTTP to compromise Oracle Life Sciences Empirica Signal. While the vulnerability is in Oracle Life Sciences Empirica Signal, attacks may significantly impact additional products (scope change). Successful attacks of this vulnerability can result in unauthorized creation, deletion or modification access to critical data or all Oracle Life Sciences Empirica Signal accessible data as well as unauthorized read access to a subset of Oracle Life Sciences Empirica Signal accessible data. CVSS 3.1 Base Score 8.5 (Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:C/C:L/I:H/A:N).	2026-04-21	8.5
CVE-2026-6921	google - chrome	Race in GPU in Google Chrome on Windows prior to 147.0.7727.117 allowed a remote attacker to potentially perform a sandbox escape via a crafted video file. (Chromium security severity: Medium)	2026-04-23	8.3
CVE-2026-31476	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ksmbd: do not expire session on binding failure When a multichannel session binding request fails (e.g. wrong password), the error path unconditionally sets sess->state = SMB2_SESSION_EXPIRED. However, during binding, sess points to the target session looked up via ksmbd_session_lookup_slowpath() -- which belongs to another connection's user. This allows a remote attacker to invalidate any active session by simply sending a binding request with a wrong password (DoS). Fix this by skipping session expiration when the failed request was a binding attempt, since the session does not belong to the current connection. The reference taken by ksmbd_session_lookup_slowpath() is still correctly released via ksmbd_user_session_put().	2026-04-22	8.2
CVE-2026-31631	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: rxrpc: Fix buffer overread in rxgk_do_verify_authenticator() Fix rxgk_do_verify_authenticator() to check the buffer size before checking the nonce.	2026-04-24	8.2
CVE-2026-34309	oracle - multiple products	Vulnerability in the PeopleSoft Enterprise PeopleTools product of Oracle PeopleSoft (component: Security). Supported versions that are affected are 8.61-8.62. Easily exploitable vulnerability allows low privileged attacker with network access via HTTP to compromise PeopleSoft Enterprise PeopleTools. Successful attacks of this vulnerability can result in unauthorized creation, deletion or modification access to critical data or all PeopleSoft Enterprise PeopleTools accessible data as well as unauthorized access to critical data or complete access to all PeopleSoft Enterprise PeopleTools accessible data. CVSS 3.1 Base Score 8.1 (Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:N).	2026-04-21	8.1
CVE-2026-31464	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: scsi: ibmvfc: Fix OOB access in ibmvfc_discover_targets_done() A malicious or compromised VIO server can return a num_written value in the discover targets MAD response that exceeds max_targets. This value is stored directly in vhost->num_targets without validation, and is then used as the loop bound in ibmvfc_alloc_targets() to index into disc_buf[], which is only allocated for max_targets entries. Indices at or beyond max_targets access kernel memory outside the DMA-coherent allocation. The out-of-bounds data is subsequently embedded in Implicit Logout and PLOGI MADs that are sent back to the VIO server, leaking kernel memory. Fix by clamping num_written to max_targets before storing it.	2026-04-22	8.1
CVE-2026-31513	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: Bluetooth: L2CAP: Fix stack-out-of-bounds read in l2cap_ecred_conn_req Syzbot reported a KASAN stack-out-of-bounds read in l2cap_build_cmd() that is triggered by a malformed Enhanced Credit Based Connection Request. The vulnerability stems from l2cap_ecred_conn_req(). The function allocates a local stack buffer (`pdu`) designed to hold a maximum of 5 Source Channel IDs (SCIDs), totaling 18 bytes. When an attacker sends a request with more than 5 SCIDs, the function calculates `rsp_len` based on this unvalidated `cmd_len` before checking if the number of SCIDs exceeds L2CAP_ECRED_MAX_CID. If the SCID count is too high, the function correctly jumps to the `response` label to reject the packet, but `rsp_len` retains the attacker's oversized value. Consequently, l2cap_send_cmd() is instructed to read past the end of the 18-byte `pdu` buffer, triggering a KASAN panic.	2026-04-22	8.1

		Fix this by moving the assignment of `rsp_len` to after the `num_scid` boundary check. If the packet is rejected, `rsp_len` will safely remain 0, and the error response will only read the 8-byte base header from the stack.		
CVE-2026-26354	dell - multiple products	Dell PowerProtect Data Domain with Domain Operating System (DD OS) of Feature Release versions 7.7.1.0 through 8.6, LTS2025 release version 8.3.1.0 through 8.3.1.10, LTS2024 release versions 7.13.1.0 through 7.13.1.60, contain a stack-based Buffer Overflow vulnerability. An unauthenticated attacker with remote access could potentially exploit this vulnerability, leading to arbitrary command execution.	2026-04-22	8.1
CVE-2026-23902	apache - dolphinscheduler	Incorrect Authorization vulnerability in Apache DolphinScheduler allows authenticated users with system login permissions to use tenants that are not defined on the platform during workflow execution. This issue affects Apache DolphinScheduler versions prior to 3.4.1. Users are recommended to upgrade to version 3.4.1, which fixes this issue.	2026-04-24	8.1
CVE-2026-31613	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: smb: client: fix OOB reads parsing symlink error response When a CREATE returns STATUS_STOPPED_ON_SYMLINK, smb2_check_message() returns success without any length validation, leaving the symlink parsers as the only defense against an untrusted server. symlink_data() walks SMB 3.1.1 error contexts with the loop test "p < end", but reads p->ErrorId at offset 4 and p->ErrorDataLength at offset 0. When the server-controlled ErrorDataLength advances p to within 1-7 bytes of end, the next iteration will read past it. When the matching context is found, sym->SymLinkErrorTag is read at offset 4 from p->ErrorContextData with no check that the symlink header itself fits. smb2_parse_symlink_response() then bounds-checks the substitute name using SMB2_SYMLINK_STRUCT_SIZE as the offset of PathBuffer from iov_base. That value is computed as sizeof(smb2_err_rsp) + sizeof(smb2_symlink_err_rsp), which is correct only when ErrorContextCount == 0. With at least one error context the symlink data sits 8 bytes deeper, and each skipped non-matching context shifts it further by 8 + ALIGN(ErrorDataLength, 8). The check is too short, allowing the substitute name read to run past iov_len. The out-of-bound heap bytes are UTF-16-decoded into the symlink target and returned to userspace via readlink(2). Fix this all up by making the loops test require the full context header to fit, rejecting sym if its header runs past end, and bound the substitute name against the actual position of sym->PathBuffer rather than a fixed offset. Because sub_offs and sub_len are 16bits, the pointer math will not overflow here with the new greater-than.	2026-04-24	8.1
CVE-2026-32172	microsoft - power_apps	Uncontrolled search path element in Microsoft Power Apps allows an unauthorized attacker to execute code over a network.	2026-04-23	8
CVE-2026-6776	mozilla - multiple products	Incorrect boundary conditions in the WebRTC: Networking component. This vulnerability was fixed in Firefox 150, Firefox ESR 140.10, Thunderbird 150, and Thunderbird 140.10.	2026-04-21	7.8
CVE-2026-35243	oracle - multiple products	Vulnerability in the Oracle Application Development Framework (ADF) product of Oracle Fusion Middleware (component: ADF Faces). Supported versions that are affected are 12.2.1.4.0 and 14.1.2.0.0. Easily exploitable vulnerability allows low privileged attacker with logon to the infrastructure where Oracle Application Development Framework (ADF) executes to compromise Oracle Application Development Framework (ADF). Successful attacks of this vulnerability can result in takeover of Oracle Application Development Framework (ADF). CVSS 3.1 Base Score 7.8 (Confidentiality, Integrity and Availability impacts). CVSS Vector: (CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H).	2026-04-21	7.8
CVE-2026-31431	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: crypto: algif_aead - Revert to operating out-of-place This mostly reverts commit 72548b093ee3 except for the copying of the associated data. There is no benefit in operating in-place in algif_aead since the source and destination come from different mappings. Get rid of all the complexity added for in-place operation and just copy the AD directly.	2026-04-22	7.8
CVE-2026-6846	red hat - multiple products	A flaw was found in binutils. A heap-buffer-overflow vulnerability exists when processing a specially crafted XCOFF (Extended Common Object File Format) object file during linking. A local attacker could trick a user into processing this malicious file, which could lead to arbitrary code execution, allowing the attacker to run unauthorized commands, or cause a denial of service, making the system unavailable.	2026-04-22	7.8

CVE-2026-31442	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: dmaengine: idxd: Fix possible invalid memory access after FLR In the case that the first Function Level Reset (FLR) concludes correctly, but in the second FLR the scratch area for the saved configuration cannot be allocated, it's possible for a invalid memory access to happen. Always set the deallocated scratch area to NULL after FLR completes.	2026-04-22	7.8
CVE-2026-31446	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ext4: fix use-after-free in update_super_work when racing with umount Commit b98535d09179 ("ext4: fix bug_on in start_this_handle during umount filesystem") moved ext4_unregister_sysfs() before flushing s_sb_upd_work to prevent new error work from being queued via /proc/fs/ext4/xx/mb_groups reads during unmount. However, this introduced a use-after-free because update_super_work calls ext4_notify_error_sysfs() -> sysfs_notify() which accesses the kobject's kernfs_node after it has been freed by kobject_del() in ext4_unregister_sysfs(): update_super_work ext4_put_super ---------- ---------- ext4_unregister_sysfs(sb) kobject_del(&sbi->s_kobj) __kobject_del() sysfs_remove_dir() kobj->sd = NULL sysfs_put(sd) kernfs_put() // RCU free ext4_notify_error_sysfs(sbi) sysfs_notify(&sbi->s_kobj) kn = kobj->sd // stale pointer kernfs_get(kn) // UAF on freed kernfs_node ext4_journal_destroy() flush_work(&sbi->s_sb_upd_work) Instead of reordering the teardown sequence, fix this by making ext4_notify_error_sysfs() detect that sysfs has already been torn down by checking s_kobj.state_in_sysfs, and skipping the sysfs_notify() call in that case. A dedicated mutex (s_error_notify_mutex) serializes ext4_notify_error_sysfs() against kobject_del() in ext4_unregister_sysfs() to prevent TOCTOU races where the kobject could be deleted between the state_in_sysfs check and the sysfs_notify() call.	2026-04-22	7.8
CVE-2026-31447	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ext4: reject mount if bigalloc with s_first_data_block != 0 bigalloc with s_first_data_block != 0 is not supported, reject mounting it.	2026-04-22	7.8
CVE-2026-31449	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ext4: validate p_idx bounds in ext4_ext_correct_indexes ext4_ext_correct_indexes() walks up the extent tree correcting index entries when the first extent in a leaf is modified. Before accessing path[k].p_idx->ei_block, there is no validation that p_idx falls within the valid range of index entries for that level. If the on-disk extent header contains a corrupted or crafted eh_entries value, p_idx can point past the end of the allocated buffer, causing a slab-out-of-bounds read. Fix this by validating path[k].p_idx against EXT_LAST_INDEX() at both access sites: before the while loop and inside it. Return -EFSCORRUPTED if the index pointer is out of range, consistent with how other bounds violations are handled in the ext4 extent tree code.	2026-04-22	7.8
CVE-2026-31453	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: xfs: avoid dereferencing log items after push callbacks After xfsaild_push_item() calls iop_push(), the log item may have been freed if the AIL lock was dropped during the push. Background inode reclaim or the dquot shrinker can free the log item while the AIL lock is not held, and the tracepoints in the switch statement dereference the log item after iop_push() returns.	2026-04-22	7.8

		Fix this by capturing the log item type, flags, and LSN before calling xfsaild_push_item(), and introducing a new xfs_ail_push_class trace event class that takes these pre-captured values and the ailp pointer instead of the log item pointer.		
CVE-2026-31454	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: xfs: save ailp before dropping the AIL lock in push callbacks In xfs_inode_item_push() and xfs_qm_dquot_logitem_push(), the AIL lock is dropped to perform buffer IO. Once the cluster buffer no longer protects the log item from reclaim, the log item may be freed by background reclaim or the dquot shrinker. The subsequent spin_lock() call dereferences lip->li_ailp, which is a use-after-free. Fix this by saving the ailp pointer in a local variable while the AIL lock is held and the log item is guaranteed to be valid.	2026-04-22	7.8
CVE-2026-31468	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: vfio/pci: Fix double free in dma-buf feature The error path through vfio_pci_core_feature_dma_buf() ignores its own advice to only use dma_buf_put() after dma_buf_export(), instead falling through the entire unwind chain. In the unlikely event that we encounter file descriptor exhaustion, this can result in an unbalanced refcount on the vfio device and double free of allocated objects. Avoid this by moving the "put" directly into the error path and return the errno rather than entering the unwind chain.	2026-04-22	7.8
CVE-2026-31469	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: virtio_net: Fix UAF on dst_ops when IFF_XMIT_DST_RELEASE is cleared and napi_tx is false A UAF issue occurs when the virtio_net driver is configured with napi_tx=N and the device's IFF_XMIT_DST_RELEASE flag is cleared (e.g., during the configuration of tc route filter rules). When IFF_XMIT_DST_RELEASE is removed from the net_device, the network stack expects the driver to hold the reference to skb->dst until the packet is fully transmitted and freed. In virtio_net with napi_tx=N, skbs may remain in the virtio transmit ring for an extended period. If the network namespace is destroyed while these skbs are still pending, the corresponding dst_ops structure has freed. When a subsequent packet is transmitted, free_old_xmit() is triggered to clean up old skbs. It then calls dst_release() on the skb associated with the stale dst_entry. Since the dst_ops (referenced by the dst_entry) has already been freed, a UAF kernel paging request occurs. fix it by adds skb_dst_drop(skb) in start_xmit to explicitly release the dst reference before the skb is queued in virtio_net. Call Trace: Unable to handle kernel paging request at virtual address ffff80007e150000 CPU: 2 UID: 0 PID: 6236 Comm: ping Kdump: loaded Not tainted 7.0.0-rc1+ #6 PREEMPT ... percpu_counter_add_batch+0x3c/0x158 lib/percpu_counter.c:98 (P) dst_release+0xe0/0x110 net/core/dst.c:177 skb_release_head_state+0xe8/0x108 net/core/skbuff.c:1177 sk_skb_reason_drop+0x54/0x2d8 net/core/skbuff.c:1255 dev_kfree_skb_any_reason+0x64/0x78 net/core/dev.c:3469 napi_consume_skb+0x1c4/0x3a0 net/core/skbuff.c:1527 __free_old_xmit+0x164/0x230 drivers/net/virtio_net.c:611 [virtio_net] free_old_xmit drivers/net/virtio_net.c:1081 [virtio_net] start_xmit+0x7c/0x530 drivers/net/virtio_net.c:3329 [virtio_net] ... Reproduction Steps: NETDEV="enp3s0" config_qdisc_route_filter() { tc qdisc del dev $NETDEV root tc qdisc add dev $NETDEV root handle 1: prio tc filter add dev $NETDEV parent 1:0 \ protocol ip prio 100 route to 100 flowid 1:1 ip route add 192.168.1.100/32 dev $NETDEV realm 100 } test_ns() { ip netns add testns	2026-04-22	7.8

		ip link set $NETDEV netns testns ip netns exec testns ifconfig $NETDEV 10.0.32.46/24 ip netns exec testns ping -c 1 10.0.32.1 ip netns del testns } config_qdisc_route_filter test_ns sleep 2 test_ns		
CVE-2026-31471	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: xfrm: iptfs: only publish mode_data after clone setup iptfs_clone_state() stores x->mode_data before allocating the reorder window. If that allocation fails, the code frees the cloned state and returns -ENOMEM, leaving x->mode_data pointing at freed memory. The xfrm clone unwind later runs destroy_state() through x->mode_data, so the failed clone path tears down IPTFS state that clone_state() already freed. Keep the cloned IPTFS state private until all allocations succeed so failed clones leave x->mode_data unset. The destroy path already handles a NULL mode_data pointer.	2026-04-22	7.8
CVE-2026-31473	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: media: mc, v4l2: serialize REINIT and REQBUFS with req_queue_mutex MEDIA_REQUEST_IOC_REINIT can run concurrently with VIDIOC_REQBUFS(0) queue teardown paths. This can race request object cleanup against vb2 queue cancellation and lead to use-after-free reports. We already serialize request queueing against STREAMON/OFF with req_queue_mutex. Extend that serialization to REQBUFS, and also take the same mutex in media_request_ioctl_reinit() so REINIT is in the same exclusion domain. This keeps request cleanup and queue cancellation from running in parallel for request-capable devices.	2026-04-22	7.8
CVE-2026-31474	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: can: isotp: fix tx.buf use-after-free in isotp_sendmsg() isotp_sendmsg() uses only cmpxchg() on so->tx.state to serialize access to so->tx.buf. isotp_release() waits for ISOTP_IDLE via wait_event_interruptible() and then calls kfree(so->tx.buf). If a signal interrupts the wait_event_interruptible() inside close() while tx.state is ISOTP_SENDING, the loop exits early and release proceeds to force ISOTP_SHUTDOWN and continues to kfree(so->tx.buf) while sendmsg may still be reading so->tx.buf for the final CAN frame in isotp_fill_dataframe(). The so->tx.buf can be allocated once when the standard tx.buf length needs to be extended. Move the kfree() of this potentially extended tx.buf to sk_destruct time when either isotp_sendmsg() and isotp_release() are done.	2026-04-22	7.8
CVE-2026-31475	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ASoC: sma1307: fix double free of devm_kzalloc() memory A previous change added NULL checks and cleanup for allocation failures in sma1307_setting_loaded(). However, the cleanup for mode_set entries is wrong. Those entries are allocated with devm_kzalloc(), so they are device-managed resources and must not be freed with kfree(). Manually freeing them in the error path can lead to a double free when devres later releases the same memory. Drop the manual kfree() loop and let devres handle the cleanup.	2026-04-22	7.8
CVE-2026-31479	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: drm/xe: always keep track of remap prev/next During 3D workload, user is reporting hitting: [413.361679] WARNING: drivers/gpu/drm/xe/xe_vm.c:1217 at vm_bind_ioctl_ops_unwind+0x1e2/0x2e0 [xe], CPU#7: vkd3d_queue/9925 [413.361944] CPU: 7 UID: 1000 PID: 9925 Comm: vkd3d_queue Kdump: loaded Not tainted 7.0.0-	2026-04-22	7.8

		070000rc3-generic #202603090038 PREEMPT(lazy) [413.361949] RIP: 0010:vm_bind_ioctl_ops_unwind+0x1e2/0x2e0 [xe] [413.362074] RSP: 0018:ffffd4c25c3df930 EFLAGS: 00010282 [413.362077] RAX: 0000000000000000 RBX: ffff8f3ee817ed10 RCX: 0000000000000000 [413.362078] RDX: 0000000000000000 RSI: 0000000000000000 RDI: 0000000000000000 [413.362079] RBP: ffffd4c25c3df980 R08: 0000000000000000 R09: 0000000000000000 [413.362081] R10: 0000000000000000 R11: 0000000000000000 R12: ffff8f41fbf99380 [413.362082] R13: ffff8f3ee817e968 R14: 00000000ffffffef R15: ffff8f43d00bd380 [413.362083] FS: 00000001040ff6c0(0000) GS:ffff8f4696d89000(0000) knlGS:00000000330b0000 [413.362085] CS: 0010 DS: 002b ES: 002b CR0: 0000000080050033 [413.362086] CR2: 00007ddfc4747000 CR3: 00000002e6262005 CR4: 0000000000f72ef0 [413.362088] PKRU: 55555554 [413.362089] Call Trace: [413.362092] <TASK> [413.362096] xe_vm_bind_ioctl+0xa9a/0xc60 [xe] Which seems to hint that the vma we are re-inserting for the ops unwind is either invalid or overlapping with something already inserted in the vm. It shouldn't be invalid since this is a re-insertion, so must have worked before. Leaving the likely culprit as something already placed where we want to insert the vma. Following from that, for the case where we do something like a rebind in the middle of a vma, and one or both mapped ends are already compatible, we skip doing the rebind of those vma and set next/prev to NULL. As well as then adjust the original unmap va range, to avoid unmapping the ends. However, if we trigger the unwind path, we end up with three va, with the two ends never being removed and the original va range in the middle still being the shrunken size. If this occurs, one failure mode is when another unwind op needs to interact with that range, which can happen with a vector of binds. For example, if we need to re-insert something in place of the original va. In this case the va is still the shrunken version, so when removing it and then doing a re-insert it can overlap with the ends, which were never removed, triggering a warning like above, plus leaving the vm in a bad state. With that, we need two things here: 1) Stop nuking the prev/next tracking for the skip cases. Instead relying on checking for skip prev/next, where needed. That way on the unwind path, we now correctly remove both ends. 2) Undo the unmap va shrinkage, on the unwind path. With the two ends now removed the unmap va should expand back to the original size again, before re-insertion. v2: - Update the explanation in the commit message, based on an actual IGT of triggering this issue, rather than conjecture. - Also undo the unmap shrinkage, for the skip case. With the two ends now removed, the original unmap va range should expand back to the original range. v3: - Track the old start/range separately. vma_size/start() uses the va info directly. (cherry picked from commit aec6969f75afbf4e01fd5fb5850ed3e9c27043ac)		
CVE-2026-31485	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: spi: spi-fsl-lpspi: fix teardown order issue (UAF) There is a teardown order issue in the driver. The SPI controller is registered using devm_spi_register_controller(), which delays unregistration of the SPI controller until after the fsl_lpspi_remove() function returns. As the fsl_lpspi_remove() function synchronously tears down the DMA channels, a running SPI transfer triggers the following NULL pointer dereference due to use after free: | fsl_lpspi 42550000.spi: I/O Error in DMA RX | Unable to handle kernel NULL pointer dereference at virtual address 0000000000000000 [...] | Call trace: | fsl_lpspi_dma_transfer+0x260/0x340 [spi_fsl_lpspi] | fsl_lpspi_transfer_one+0x198/0x448 [spi_fsl_lpspi] | spi_transfer_one_message+0x49c/0x7c8 | __spi_pump_transfer_message+0x120/0x420	2026-04-22	7.8

		| __spi_sync+0x2c4/0x520 | spi_sync+0x34/0x60 | spidev_message+0x20c/0x378 [spidev] | spidev_ioctl+0x398/0x750 [spidev] [...] Switch from devm_spi_register_controller() to spi_register_controller() in fsl_lpspi_probe() and add the corresponding spi_unregister_controller() in fsl_lpspi_remove().		
CVE-2026-31488	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: drm/amd/display: Do not skip unrelated mode changes in DSC validation Starting with commit 17ce8a6907f7 ("drm/amd/display: Add dsc pre-validation in atomic check"), amdgpu resets the CRTC state mode_changed flag to false when recomputing the DSC configuration results in no timing change for a particular stream. However, this is incorrect in scenarios where a change in MST/DSC configuration happens in the same KMS commit as another (unrelated) mode change. For example, the integrated panel of a laptop may be configured differently (e.g., HDR enabled/disabled) depending on whether external screens are attached. In this case, plugging in external DP-MST screens may result in the mode_changed flag being dropped incorrectly for the integrated panel if its DSC configuration did not change during precomputation in pre_validate_dsc(). At this point, however, dm_update_crtc_state() has already created new streams for CRTCs with DSC-independent mode changes. In turn, amdgpu_dm_commit_streams() will never release the old stream, resulting in a memory leak. amdgpu_dm_atomic_commit_tail() will never acquire a reference to the new stream either, which manifests as a use-after-free when the stream gets disabled later on: BUG: KASAN: use-after-free in dc_stream_release+0x25/0x90 [amdgpu] Write of size 4 at addr ffff88813d836524 by task kworker/9:9/29977 Workqueue: events drm_mode_rmfb_work_fn Call Trace: <TASK> dump_stack_lvl+0x6e/0xa0 print_address_description.constprop.0+0x88/0x320 ? dc_stream_release+0x25/0x90 [amdgpu] print_report+0xfc/0x1ff ? srso_alias_return_thunk+0x5/0xfbef5 ? __virt_addr_valid+0x225/0x4e0 ? dc_stream_release+0x25/0x90 [amdgpu] kasan_report+0xe1/0x180 ? dc_stream_release+0x25/0x90 [amdgpu] kasan_check_range+0x125/0x200 dc_stream_release+0x25/0x90 [amdgpu] dc_state_destruct+0x14d/0x5c0 [amdgpu] dc_state_release.part.0+0x4e/0x130 [amdgpu] dm_atomic_destroy_state+0x3f/0x70 [amdgpu] drm_atomic_state_default_clear+0x8ee/0xf30 ? drm_mode_object_put.part.0+0xb1/0x130 __drm_atomic_state_free+0x15c/0x2d0 atomic_remove_fb+0x67e/0x980 Since there is no reliable way of figuring out whether a CRTC has unrelated mode changes pending at the time of DSC validation, remember the value of the mode_changed flag from before the point where a CRTC was marked as potentially affected by a change in DSC configuration. Reset the mode_changed flag to this earlier value instead in pre_validate_dsc(). (cherry picked from commit cc7c7121ae082b7b82891baa7280f1ff2608f22b)	2026-04-22	7.8
CVE-2026-31489	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: spi: meson-spicc: Fix double-put in remove path meson_spicc_probe() registers the controller with devm_spi_register_controller(), so teardown already drops the controller reference via devm cleanup. Calling spi_controller_put() again in meson_spicc_remove() causes a double-put.	2026-04-22	7.8
CVE-2026-31490	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: drm/xe/pf: Fix use-after-free in migration restore When an error is returned from xe_sriov_pf_migration_restore_produce(),		

		the data pointer is not set to NULL, which can trigger use-after-free in subsequent .write() calls. Set the pointer to NULL upon error to fix the problem. (cherry picked from commit 4f53d8c6d23527d734fe3531d08e15cb170a0819)		
CVE-2026-31493	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: RDMA/efa: Fix use of completion ctx after free On admin queue completion handling, if the admin command completed with error we print data from the completion context. The issue is that we already freed the completion context in polling/interrupts handler which means we print data from context in an unknown state (it might be already used again). Change the admin submission flow so alloc/dealloc of the context will be symmetric and dealloc will be called after any potential use of the context.	2026-04-22	7.8
CVE-2026-31494	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net: macb: use the current queue number for stats There's a potential mismatch between the memory reserved for statistics and the amount of memory written. gem_get_sset_count() correctly computes the number of stats based on the active queues, whereas gem_get_ethtool_stats() indiscriminately copies data using the maximum number of queues, and in the case the number of active queues is less than MACB_MAX_QUEUES, this results in a OOB write as observed in the KASAN splat. ========== BUG: KASAN: vmalloc-out-of-bounds in gem_get_ethtool_stats+0x54/0x78 [macb] Write of size 760 at addr ffff80008080b000 by task ethtool/1027 CPU: [...] Tainted: [E]=UNSIGNED_MODULE Hardware name: raspberrypi rpi/rpi, BIOS 2025.10 10/01/2025 Call trace: show_stack+0x20/0x38 (C) dump_stack_lvl+0x80/0xf8 print_report+0x384/0x5e0 kasan_report+0xa0/0xf0 kasan_check_range+0xe8/0x190 __asan_memcpy+0x54/0x98 gem_get_ethtool_stats+0x54/0x78 [macb 926c13f3af83b0c6fe64badb21ec87d5e93fcf65] dev_ethtool+0x1220/0x38c0 dev_ioctl+0x4ac/0xca8 sock_do_ioctl+0x170/0x1d8 sock_ioctl+0x484/0x5d8 __arm64_sys_ioctl+0x12c/0x1b8 invoke_syscall+0xd4/0x258 el0_svc_common.constprop.0+0xb4/0x240 do_el0_svc+0x48/0x68 el0_svc+0x40/0xf8 el0t_64_sync_handler+0xa0/0xe8 el0t_64_sync+0x1b0/0x1b8 The buggy address belongs to a 1-page vmalloc region starting at 0xffff80008080b000 allocated at dev_ethtool+0x11f0/0x38c0 The buggy address belongs to the physical page: page: refcount:1 mapcount:0 mapping:0000000000000000 index:0xffff00000a333000 pfn:0xa333 flags: 0x7fffc000000000(node=0|zone=0|lastcpupid=0x1ffff) raw: 007fffc000000000 0000000000000000 dead000000000122 0000000000000000 raw: ffff00000a333000 0000000000000000 00000001ffffffff 0000000000000000 page dumped because: kasan: bad access detected Memory state around the buggy address: ffff80008080b080: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ffff80008080b100: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 >ffff80008080b180: 00 00 00 00 00 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 ^ ffff80008080b200: f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 ffff80008080b280: f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 ========== Fix it by making sure the copied size only considers the active number of queues.	2026-04-22	7.8

CVE-2026-31500	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: Bluetooth: btintel: serialize btintel_hw_error() with hci_req_sync_lock btintel_hw_error() issues two __hci_cmd_sync() calls (HCI_OP_RESET and Intel exception-info retrieval) without holding hci_req_sync_lock(). This lets it race against hci_dev_do_close() -> btintel_shutdown_combined(), which also runs __hci_cmd_sync() under the same lock. When both paths manipulate hdev->req_status/req_rsp concurrently, the close path may free the response skb first, and the still-running hw_error path hits a slab-use-after-free in kfree_skb(). Wrap the whole recovery sequence in hci_req_sync_lock/unlock so it is serialized with every other synchronous HCI command issuer. Below is the data race report and the kasan report: BUG: data-race in __hci_cmd_sync_sk / btintel_shutdown_combined read of hdev->req_rsp at net/bluetooth/hci_sync.c:199 by task kworker/u17:1/83: __hci_cmd_sync_sk+0x12f2/0x1c30 net/bluetooth/hci_sync.c:200 __hci_cmd_sync+0x55/0x80 net/bluetooth/hci_sync.c:223 btintel_hw_error+0x114/0x670 drivers/bluetooth/btintel.c:254 hci_error_reset+0x348/0xa30 net/bluetooth/hci_core.c:1030 write/free by task ioctl/22580: btintel_shutdown_combined+0xd0/0x360 drivers/bluetooth/btintel.c:3648 hci_dev_close_sync+0x9ae/0x2c10 net/bluetooth/hci_sync.c:5246 hci_dev_do_close+0x232/0x460 net/bluetooth/hci_core.c:526 BUG: KASAN: slab-use-after-free in sk_skb_reason_drop+0x43/0x380 net/core/skbuff.c:1202 Read of size 4 at addr ffff888144a738dc by task kworker/u17:1/83: __hci_cmd_sync_sk+0x12f2/0x1c30 net/bluetooth/hci_sync.c:200 __hci_cmd_sync+0x55/0x80 net/bluetooth/hci_sync.c:223 btintel_hw_error+0x186/0x670 drivers/bluetooth/btintel.c:260	2026-04-22	7.8
CVE-2026-31502	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: team: fix header_ops type confusion with non-Ethernet ports Similar to commit 950803f72547 ("bonding: fix type confusion in bond_setup_by_slave()") team has the same class of header_ops type confusion. For non-Ethernet ports, team_setup_by_port() copies port_dev->header_ops directly. When the team device later calls dev_hard_header() or dev_parse_header(), these callbacks can run with the team net_device instead of the real lower device, so netdev_priv(dev) is interpreted as the wrong private type and can crash. The syzbot report shows a crash in bond_header_create(), but the root cause is in team: the topology is gre -> bond -> team, and team calls the inherited header_ops with its own net_device instead of the lower device, so bond_header_create() receives a team device and interprets netdev_priv() as bonding private data, causing a type confusion crash. Fix this by introducing team header_ops wrappers for create/parse, selecting a team port under RCU, and calling the lower device callbacks with port->dev, so each callback always sees the correct net_device context. Also pass the selected lower device to the lower parse callback, so recursion is bounded in stacked non-Ethernet topologies and parse callbacks always run with the correct device context.	2026-04-22	7.8
CVE-2026-31504	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net: fix fanout UAF in packet_release() via NETDEV_UP race `packet_release()` has a race window where `NETDEV_UP` can re-register a socket into a fanout group's `arr[]` array. The re-registration is not cleaned up by `fanout_release()`, leaving a dangling pointer in the fanout array. `packet_release()` does NOT zero `po->num` in its `bind_lock` section. After releasing `bind_lock`, `po->num` is still non-zero and `po->ifindex` still matches the bound device. A concurrent `packet_notifier(NETDEV_UP)` that already found the socket in `sklist` can re-register the hook.	2026-04-22	7.8

		For fanout sockets, this re-registration calls `__fanout_link(sk, po)` which adds the socket back into `f->arr[]` and increments `f->num_members`, but does NOT increment `f->sk_ref`. The fix sets `po->num` to zero in `packet_release` while `bind_lock` is held to prevent NETDEV_UP from linking, preventing the race window. This bug was found following an additional audit with Claude Code based on CVE-2025-38617.		
CVE-2026-31505	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: iavf: fix out-of-bounds writes in iavf_get_ethtool_stats() iavf incorrectly uses real_num_tx_queues for ETH_SS_STATS. Since the value could change in runtime, we should use num_tx_queues instead. Moreover iavf_get_ethtool_stats() uses num_active_queues while iavf_get_sset_count() and iavf_get_stat_strings() use real_num_tx_queues, which triggers out-of-bounds writes when we do "ethtool -L" and "ethtool -S" simultaneously [1]. For example when we change channels from 1 to 8, Thread 3 could be scheduled before Thread 2, and out-of-bounds writes could be triggered in Thread 3: Thread 1 (ethtool -L) Thread 2 (work) Thread 3 (ethtool -S) iavf_set_channels() ... iavf_alloc_queues() -> num_active_queues = 8 iavf_schedule_finish_config() iavf_get_sset_count() real_num_tx_queues: 1 -> buffer for 1 queue iavf_get_ethtool_stats() num_active_queues: 8 -> out-of-bounds! iavf_finish_config() -> real_num_tx_queues = 8 Use immutable num_tx_queues in all related functions to avoid the issue. [1] BUG: KASAN: vmalloc-out-of-bounds in iavf_add_one_ethtool_stat+0x200/0x270 Write of size 8 at addr ffffc900031c9080 by task ethtool/5800 CPU: 1 UID: 0 PID: 5800 Comm: ethtool Not tainted 6.19.0-enjuk-08403-g8137e3db7f1c #241 PREEMPT(full) Hardware name: QEMU Standard PC (Q35 + ICH9, 2009), BIOS 1.16.3-debian-1.16.3-2 04/01/2014 Call Trace: <TASK> dump_stack_lvl+0x6f/0xb0 print_report+0x170/0x4f3 kasan_report+0xe1/0x180 iavf_add_one_ethtool_stat+0x200/0x270 iavf_get_ethtool_stats+0x14c/0x2e0 __dev_ethtool+0x3d0c/0x5830 dev_ethtool+0x12d/0x270 dev_ioctl+0x53c/0xe30 sock_do_ioctl+0x1a9/0x270 sock_ioctl+0x3d4/0x5e0 __x64_sys_ioctl+0x137/0x1c0 do_syscall_64+0xf3/0x690 entry_SYSCALL_64_after_hwframe+0x77/0x7f RIP: 0033:0x7f7da0e6e36d ... </TASK> The buggy address belongs to a 1-page vmalloc region starting at 0xffffc900031c9000 allocated at __dev_ethtool+0x3cc9/0x5830 The buggy address belongs to the physical page: page: refcount:1 mapcount:0 mapping:0000000000000000 index:0xffff88813a013de0 pfn:0x13a013 flags: 0x200000000000000(node=0|zone=2) raw: 0200000000000000 0000000000000000 dead000000000122 0000000000000000 raw: ffff88813a013de0 0000000000000000 00000001ffffffff 0000000000000000 page dumped because: kasan: bad access detected Memory state around the buggy address: ffffc900031c8f80: f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8	2026-04-22	7.8

		ffffc900031c9000: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 >ffffc900031c9080: f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 ^ ffffc900031c9100: f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 ffffc900031c9180: f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8 f8		
CVE-2026-31506	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net: bcmasp: fix double free of WoL irq We do not need to free wol_irq since it was instantiated with devm_request_irq(). So devres will free for us.	2026-04-22	7.8
CVE-2026-31507	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net/smc: fix double-free of smc_spd_priv when tee() duplicates splice pipe buffer smc_rx_splice() allocates one smc_spd_priv per pipe_buffer and stores the pointer in pipe_buffer.private. The pipe_buf_operations for these buffers used .get = generic_pipe_buf_get, which only increments the page reference count when tee(2) duplicates a pipe buffer. The smc_spd_priv pointer itself was not handled, so after tee() both the original and the cloned pipe_buffer share the same smc_spd_priv *. When both pipes are subsequently released, smc_rx_pipe_buf_release() is called twice against the same object: 1st call: kfree(priv) sock_put(sk) smc_rx_update_cons() [correct] 2nd call: kfree(priv) sock_put(sk) smc_rx_update_cons() [UAF] KASAN reports a slab-use-after-free in smc_rx_pipe_buf_release(), which then escalates to a NULL-pointer dereference and kernel panic via smc_rx_update_consumer() when it chases the freed priv->smc pointer: BUG: KASAN: slab-use-after-free in smc_rx_pipe_buf_release+0x78/0x2a0 Read of size 8 at addr ffff888004a45740 by task smc_splice_tee_/74 Call Trace: <TASK> dump_stack_lvl+0x53/0x70 print_report+0xce/0x650 kasan_report+0xc6/0x100 smc_rx_pipe_buf_release+0x78/0x2a0 free_pipe_info+0xd4/0x130 pipe_release+0x142/0x160 __fput+0x1c6/0x490 __x64_sys_close+0x4f/0x90 do_syscall_64+0xa6/0x1a0 entry_SYSCALL_64_after_hwframe+0x77/0x7f </TASK> BUG: kernel NULL pointer dereference, address: 0000000000000020 RIP: 0010:smc_rx_update_consumer+0x8d/0x350 Call Trace: <TASK> smc_rx_pipe_buf_release+0x121/0x2a0 free_pipe_info+0xd4/0x130 pipe_release+0x142/0x160 __fput+0x1c6/0x490 __x64_sys_close+0x4f/0x90 do_syscall_64+0xa6/0x1a0 entry_SYSCALL_64_after_hwframe+0x77/0x7f </TASK> Kernel panic - not syncing: Fatal exception Beyond the memory-safety problem, duplicating an SMC splice buffer is semantically questionable: smc_rx_update_cons() would advance the consumer cursor twice for the same data, corrupting receive-window accounting. A refcount on smc_spd_priv could fix the double-free, but the cursor-accounting issue would still need to be addressed separately. The .get callback is invoked by both tee(2) and splice_pipe_to_pipe() for partial transfers; both will now return -EFAULT. Users who need to duplicate SMC socket data must use a copy-based read path.	2026-04-22	7.8
CVE-2026-31508	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net: openvswitch: Avoid releasing netdev before teardown completes The patch cited in the Fixes tag below changed the teardown code for OVS ports to no longer unconditionally take the RTNL. After this change, the netdev_destroy() callback can proceed immediately to the call_rcu() invocation if the IFF_OVS_DATAPATH flag is already cleared on the netdev.	2026-04-22	7.8

		The ovs_netdev_detach_dev() function clears the flag before completing the unregistration, and if it gets preempted after clearing the flag (as can happen on an -rt kernel), netdev_destroy() can complete and the device can be freed before the unregistration completes. This leads to a splat like: [998.393867] Oops: general protection fault, probably for non-canonical address 0xff00000001000239: 0000 [#1] SMP PTI [998.393877] CPU: 42 UID: 0 PID: 55177 Comm: ip Kdump: loaded Not tainted 6.12.0-211.1.1.el10_2.x86_64+rt #1 PREEMPT_RT [998.393886] Hardware name: Dell Inc. PowerEdge R740/0JMK61, BIOS 2.24.0 03/27/2025 [998.393889] RIP: 0010:dev_set_promiscuity+0x8d/0xa0 [998.393901] Code: 00 00 75 d8 48 8b 53 08 48 83 ba b0 02 00 00 00 75 ca 48 83 c4 08 5b c3 cc cc cc cc 48 83 bf 48 09 00 00 00 75 91 48 8b 47 08 <48> 83 b8 b0 02 00 00 00 74 97 eb 81 0f 1f 80 00 00 00 00 90 90 90 [998.393906] RSP: 0018:ffffce5864a5f6a0 EFLAGS: 00010246 [998.393912] RAX: ff00000000ffff89 RBX: ffff894d0adf5a05 RCX: 0000000000000000 [998.393917] RDX: 0000000000000000 RSI: 00000000ffffffff RDI: ffff894d0adf5a05 [998.393921] RBP: ffff894d19252000 R08: ffff894d19252000 R09: 0000000000000000 [998.393924] R10: ffff894d19252000 R11: ffff894d192521b8 R12: 0000000000000006 [998.393927] R13: ffffce5864a5f738 R14: 00000000ffffffe2 R15: 0000000000000000 [998.393931] FS: 00007fad61971800(0000) GS:ffff894cc0140000(0000) knlGS:0000000000000000 [998.393936] CS: 0010 DS: 0000 ES: 0000 CR0: 0000000080050033 [998.393940] CR2: 000055df0a2a6e40 CR3: 000000011c7fe003 CR4: 00000000007726f0 [998.393944] PKRU: 55555554 [998.393946] Call Trace: [998.393949] <TASK> [998.393952] ? show_trace_log_lvl+0x1b0/0x2f0 [998.393961] ? show_trace_log_lvl+0x1b0/0x2f0 [998.393975] ? dp_device_event+0x41/0x80 [openvswitch] [998.394009] ? __die_body.cold+0x8/0x12 [998.394016] ? die_addr+0x3c/0x60 [998.394027] ? exc_general_protection+0x16d/0x390 [998.394042] ? asm_exc_general_protection+0x26/0x30 [998.394058] ? dev_set_promiscuity+0x8d/0xa0 [998.394066] ? ovs_netdev_detach_dev+0x3a/0x80 [openvswitch] [998.394092] dp_device_event+0x41/0x80 [openvswitch] [998.394102] notifier_call_chain+0x5a/0xd0 [998.394106] unregister_netdevice_many_notify+0x51b/0xa60 [998.394110] rtnl_dellink+0x169/0x3e0 [998.394121] ? rt_mutex_slowlock.constprop.0+0x95/0xd0 [998.394125] rtnetlink_rcv_msg+0x142/0x3f0 [998.394128] ? avc_has_perm_noaudit+0x69/0xf0 [998.394130] ? __pfx_rtnetlink_rcv_msg+0x10/0x10 [998.394132] netlink_rcv_skb+0x50/0x100 [998.394138] netlink_unicast+0x292/0x3f0 [998.394141] netlink_sendmsg+0x21b/0x470 [998.394145] ____sys_sendmsg+0x39d/0x3d0 [998.394149] ___sys_sendmsg+0x9a/0xe0 [998.394156] __sys_sendmsg+0x7a/0xd0 [998.394160] do_syscall_64+0x7f/0x170 [998.394162] entry_SYSCALL_64_after_hwframe+0x76/0x7e [998.394165] RIP: 0033:0x7fad61bf4724 [998.394188] Code: 89 02 b8 ff ff ff ff eb bb 66 2e 0f 1f 84 00 00 00 00 00 0f 1f 00 f3 0f 1e fa 80 3d c5 e9 0c 00 00 74 13 b8 2e 00 00 00 0f 05 <48> 3d 00 f0 ff ff 77 54 c3 0f 1f 00 48 83 ec 28 89 54 24 1c 48 89 [998.394189] RSP: 002b:00007ffd7e2f7cb8 EFLAGS: 00000202 ORIG_RAX: 000000000000002e [998.394191] RAX: ffffffffffffffda RBX: 0000000000000001 RCX: 00007fad61bf4724 [998.394193] RDX: 0000000000000000 RSI: 00007ffd7e2f7d20 RDI: 0000000000000003 [998.394194] RBP: 00007ffd7e2f7d90 R08: 0000000000000010 R09: 000000000000003f [998.394195] R10: 000055df11558010 R11: 0000000000000202 R12: 00007ffd7e2 ---truncated---		
CVE-2026-31511	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: Bluetooth: MGMT: Fix dangling pointer on mgmt_add_adv_patterns_monitor_complete This fixes the condition checking so mgmt_pending_valid is executed whenever status != -ECANCELED otherwise calling mgmt_pending_free(cmd) would kfree(cmd) without unlinking it from the list first, leaving a dangling pointer. Any subsequent list traversal (e.g., mgmt_pending_foreach during __mgmt_power_off, or another mgmt_pending_valid call) would dereference freed memory.	2026-04-22	7.8
CVE-2026-31516	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: xfrm: prevent policy_hthresh.work from racing with netns teardown A XFRM_MSG_NEWSPDINFO request can queue the per-net work item policy_hthresh.work onto the system workqueue.	2026-04-22	7.8

		The queued callback, xfrm_hash_rebuild(), retrieves the enclosing struct net via container_of(). If the net namespace is torn down before that work runs, the associated struct net may already have been freed, and xfrm_hash_rebuild() may then dereference stale memory. xfrm_policy_fini() already flushes policy_hash_work during teardown, but it does not synchronize policy_hthresh.work. Synchronize policy_hthresh.work in xfrm_policy_fini() as well, so the queued work cannot outlive the net namespace teardown and access a freed struct net.		
CVE-2026-31525	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: bpf: Fix undefined behavior in interpreter sdiv/smod for INT_MIN The BPF interpreter's signed 32-bit division and modulo handlers use the kernel abs() macro on s32 operands. The abs() macro documentation (include/linux/math.h) explicitly states the result is undefined when the input is the type minimum. When DST contains S32_MIN (0x80000000), abs((s32)DST) triggers undefined behavior and returns S32_MIN unchanged on arm64/x86. This value is then sign-extended to u64 as 0xFFFFFFFF80000000, causing do_div() to compute the wrong result. The verifier's abstract interpretation (scalar32_min_max_sdiv) computes the mathematically correct result for range tracking, creating a verifier/interpreter mismatch that can be exploited for out-of-bounds map value access. Introduce abs_s32() which handles S32_MIN correctly by casting to u32 before negating, avoiding signed overflow entirely. Replace all 8 abs((s32)...) call sites in the interpreter's sdiv32/smod32 handlers. s32 is the only affected case -- the s64 division/modulo handlers do not use abs().	2026-04-22	7.8
CVE-2026-31527	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: driver core: platform: use generic driver_override infrastructure When a driver is probed through __driver_attach(), the bus' match() callback is called without the device lock held, thus accessing the driver_override field without a lock, which can cause a UAF. Fix this by using the driver-core driver_override infrastructure taking care of proper locking internally. Note that calling match() from __driver_attach() without the device lock held is intentional. [1]	2026-04-22	7.8
CVE-2026-31528	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: perf: Make sure to use pmu_ctx->pmu for groups Oliver reported that x86_pmu_del() ended up doing an out-of-bound memory access when group_sched_in() fails and needs to roll back. This *should* be handled by the transaction callbacks, but he found that when the group leader is a software event, the transaction handlers of the wrong PMU are used. Despite the move_group case in perf_event_open() and group_sched_in() using pmu_ctx->pmu. Turns out, inherit uses event->pmu to clone the events, effectively undoing the move_group case for all inherited contexts. Fix this by also making inherit use pmu_ctx->pmu, ensuring all inherited counters end up in the same pmu context. Similarly, __perf_event_read() should use equally use pmu_ctx->pmu for the group case.	2026-04-22	7.8
CVE-2026-31530	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: cxl/port: Fix use after free of parent_port in cxl_detach_ep() cxl_detach_ep() is called during bottom-up removal when all CXL memory devices beneath a switch port have been removed. For each port in the hierarchy it locks both the port and its parent, removes the endpoint, and if the port is now empty, marks it dead and unregisters the port by calling delete_switch_port(). There are two places during this work where the parent_port may be used after freeing: First, a concurrent detach may have already processed a port by the time a second worker finds it via bus_find_device(). Without pinning parent_port, it may already be freed when we discover port->dead and attempt to unlock the parent_port. In a production kernel that's a	2026-04-22	7.8

		silent memory corruption, with lock debug, it looks like this: []DEBUG_LOCKS_WARN_ON(__owner_task(owner) != get_current()) []WARNING: kernel/locking/mutex.c:949 at __mutex_unlock_slowpath+0x1ee/0x310 []Call Trace: []mutex_unlock+0xd/0x20 []cxl_detach_ep+0x180/0x400 [cxl_core] []devm_action_release+0x10/0x20 []devres_release_all+0xa8/0xe0 []device_unbind_cleanup+0xd/0xa0 []really_probe+0x1a6/0x3e0 Second, delete_switch_port() releases three devm actions registered against parent_port. The last of those is unregister_port() and it calls device_unregister() on the child port, which can cascade. If parent_port is now also empty the device core may unregister and free it too. So by the time delete_switch_port() returns, parent_port may be free, and the subsequent device_unlock(&parent_port->dev) operates on freed memory. The kernel log looks same as above, with a different offset in cxl_detach_ep(). Both of these issues stem from the absence of a lifetime guarantee between a child port and its parent port. Establish a lifetime rule for ports: child ports hold a reference to their parent device until release. Take the reference when the port is allocated and drop it when released. This ensures the parent is valid for the full lifetime of the child and eliminates the use after free window in cxl_detach_ep(). This is easily reproduced with a reload of cxl_acpi in QEMU with CXL devices present.		
CVE-2026-31532	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: can: raw: fix ro->uniq use-after-free in raw_rcv() raw_release() unregisters raw CAN receive filters via can_rx_unregister(), but receiver deletion is deferred with call_rcu(). This leaves a window where raw_rcv() may still be running in an RCU read-side critical section after raw_release() frees ro->uniq, leading to a use-after-free of the percpu uniq storage. Move free_percpu(ro->uniq) out of raw_release() and into a raw-specific socket destructor. can_rx_unregister() takes an extra reference to the socket and only drops it from the RCU callback, so freeing uniq from sk_destruct ensures the percpu area is not released until the relevant callbacks have drained. [mkl: applied manually]	2026-04-23	7.8
CVE-2026-33999	red hat - multiple products	A flaw was found in the X.Org X server. This integer underflow vulnerability, specifically in the XKB compatibility map handling, allows an attacker with local or remote X11 server access to trigger a buffer read overrun. This can lead to memory-safety violations and potentially a denial of service (DoS) or other severe impacts.	2026-04-23	7.8
CVE-2026-34001	red hat - multiple products	A flaw was found in the X.Org X server. This use-after-free vulnerability occurs in the XSYNC fence triggering logic, specifically within the miSyncTriggerFence() function. An attacker with access to the X11 server can exploit this without user interaction, leading to a server crash and potentially enabling memory corruption. This could result in a denial of service or further compromise of the system.	2026-04-23	7.8
CVE-2026-34003	red hat - multiple products	A flaw was found in the X.Org X server's XKB key types request validation. A local attacker could send a specially crafted request to the X server, leading to an out-of-bounds memory access vulnerability. This could result in the disclosure of sensitive information or cause the server to crash, leading to a Denial of Service (DoS). In certain configurations, higher impact outcomes may be possible.	2026-04-23	7.8
CVE-2026-31541	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: tracing: Fix trace_marker copy link list updates When the "copy_trace_marker" option is enabled for an instance, anything written into /sys/kernel/tracing/trace_marker is also copied into that instances buffer. When the option is set, that instance's trace_array descriptor is added to the marker_copies link list. This list is protected by RCU, as all iterations uses an RCU protected list traversal. When the instance is deleted, all the flags that were enabled are cleared. This also clears the copy_trace_marker flag and removes the trace_array descriptor from the list. The issue is after the flags are called, a direct call to update_marker_trace() is performed to clear the flag. This function returns true if the state of the flag changed and false otherwise. If it	2026-04-24	7.8

		returns true here, synchronize_rcu() is called to make sure all readers see that its removed from the list. But since the flag was already cleared, the state does not change and the synchronization is never called, leaving a possible UAF bug. Move the clearing of all flags below the updating of the copy_trace_marker option which then makes sure the synchronization is performed. Also use the flag for checking the state in update_marker_trace() instead of looking at if the list is empty.		
CVE-2026-31548	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: wifi: cfg80211: cancel pmsr_free_wk in cfg80211_pmsr_wdev_down When the nl80211 socket that originated a PMSR request is closed, cfg80211_release_pmsr() sets the request's nl_portid to zero and schedules pmsr_free_wk to process the abort asynchronously. If the interface is concurrently torn down before that work runs, cfg80211_pmsr_wdev_down() calls cfg80211_pmsr_process_abort() directly. However, the already-scheduled pmsr_free_wk work item remains pending and may run after the interface has been removed from the driver. This could cause the driver's abort_pmsr callback to operate on a torn-down interface, leading to undefined behavior and potential crashes. Cancel pmsr_free_wk synchronously in cfg80211_pmsr_wdev_down() before calling cfg80211_pmsr_process_abort(). This ensures any pending or in-progress work is drained before interface teardown proceeds, preventing the work from invoking the driver abort callback after the interface is gone.	2026-04-24	7.8
CVE-2026-31554	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: futex: Require sys_futex_requeue() to have identical flags Nicholas reported that his LLM found it was possible to create a UaF when sys_futex_requeue() is used with different flags. The initial motivation for allowing different flags was the variable sized futex, but since that hasn't been merged (yet), simply mandate the flags are identical, as is the case for the old style sys_futex() requeue operations.	2026-04-24	7.8
CVE-2026-31566	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: drm/amdgpu: Fix fence put before wait in amdgpu_amdkfd_submit_ib amdgpu_amdkfd_submit_ib() submits a GPU job and gets a fence from amdgpu_ib_schedule(). This fence is used to wait for job completion. Currently, the code drops the fence reference using dma_fence_put() before calling dma_fence_wait(). If dma_fence_put() releases the last reference, the fence may be freed before dma_fence_wait() is called. This can lead to a use-after-free. Fix this by waiting on the fence first and releasing the reference only after dma_fence_wait() completes. Fixes the below: drivers/gpu/drm/amd/amdgpu/amdgpu_amdkfd.c:697 amdgpu_amdkfd_submit_ib() warn: passing freed memory 'f' (line 696) (cherry picked from commit 8b9e5259adc385b61a6590a13b82ae0ac2bd3482)	2026-04-24	7.8
CVE-2026-31576	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: media: hackrf: fix to not free memory after the device is registered in hackrf_probe() In hackrf driver, the following race condition occurs: ``` CPU0 CPU1 hackrf_probe() kzalloc(); // alloc hackrf_dev .... v4l2_device_register(); .... fd = sys_open("/path/to/dev"); // open hackrf fd .... v4l2_device_unregister();	2026-04-24	7.8

		.... kfree(); // free hackrf_dev .... sys_ioctl(fd, ...); v4l2_ioctl(); video_is_registered() // UAF!! .... sys_close(fd); v4l2_release() // UAF!! hackrf_video_release() kfree(); // DFB!! ``` When a V4L2 or video device is unregistered, the device node is removed so new open() calls are blocked. However, file descriptors that are already open-and any in-flight I/O-do not terminate immediately; they remain valid until the last reference is dropped and the driver's release() is invoked. Therefore, freeing device memory on the error path after hackrf_probe() has registered dev it will lead to a race to use-after-free vuln, since those already-open handles haven't been released yet. And since release() free memory too, race to use-after-free and double-free vuln occur. To prevent this, if device is registered from probe(), it should be modified to free memory only through release() rather than calling kfree() directly.		
CVE-2026-31578	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: media: as102: fix to not free memory after the device is registered in as102_usb_probe() In as102_usb driver, the following race condition occurs: ``` CPU0 CPU1 as102_usb_probe() kzalloc(); // alloc as102_dev_t .... usb_register_dev(); fd = sys_open("/path/to/dev"); // open as102 fd .... usb_deregister_dev(); .... kfree(); // free as102_dev_t .... sys_close(fd); as102_release() // UAF!! as102_usb_release() kfree(); // DFB!! ``` When a USB character device registered with usb_register_dev() is later unregistered (via usb_deregister_dev() or disconnect), the device node is removed so new open() calls fail. However, file descriptors that are already open do not go away immediately: they remain valid until the last reference is dropped and the driver's .release() is invoked. In as102, as102_usb_probe() calls usb_register_dev() and then, on an error path, does usb_deregister_dev() and frees as102_dev_t right away. If userspace raced a successful open() before the deregistration, that open FD will later hit as102_release() --> as102_usb_release() and access or free as102_dev_t again, occur a race to use-after-free and double-free vuln. The fix is to never kfree(as102_dev_t) directly once usb_register_dev() has succeeded. After deregistration, defer freeing memory to .release(). In other words, let release() perform the last kfree when the final open FD is closed.	2026-04-24	7.8
CVE-2026-31580	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: bcache: fix cached_dev.sb_bio use-after-free and crash In our production environment, we have received multiple crash reports regarding libceph, which have caught our attention: ``` [6888366.280350] Call Trace:	2026-04-24	7.8

		[6888366.280452] blk_update_request+0x14e/0x370 [6888366.280561] blk_mq_end_request+0x1a/0x130 [6888366.280671] rbd_img_handle_request+0x1a0/0x1b0 [rbd] [6888366.280792] rbd_obj_handle_request+0x32/0x40 [rbd] [6888366.280903] __complete_request+0x22/0x70 [libceph] [6888366.281032] osd_dispatch+0x15e/0xb40 [libceph] [6888366.281164] ? inet_recvmsg+0x5b/0xd0 [6888366.281272] ? ceph_tcp_recvmsg+0x6f/0xa0 [libceph] [6888366.281405] ceph_con_process_message+0x79/0x140 [libceph] [6888366.281534] ceph_con_v1_try_read+0x5d7/0xf30 [libceph] [6888366.281661] ceph_con_workfn+0x329/0x680 [libceph] ``` After analyzing the coredump file, we found that the address of dc->sb_bio has been freed. We know that cached_dev is only freed when it is stopped. Since sb_bio is a part of struct cached_dev, rather than an alloc every time. If the device is stopped while writing to the superblock, the released address will be accessed at endio. This patch hopes to wait for sb_write to complete in cached_dev_free. It should be noted that we analyzed the cause of the problem, then tell all details to the QWEN and adopted the modifications it made.		
CVE-2026-31581	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ALSA: 6fire: fix use-after-free on disconnect In usb6fire_chip_abort(), the chip struct is allocated as the card's private data (via snd_card_new with sizeof(struct sfire_chip)). When snd_card_free_when_closed() is called and no file handles are open, the card and embedded chip are freed synchronously. The subsequent chip->card = NULL write then hits freed slab memory. Call trace: usb6fire_chip_abort sound/usb/6fire/chip.c:59 [inline] usb6fire_chip_disconnect+0x348/0x358 sound/usb/6fire/chip.c:182 usb_unbind_interface+0x1a8/0x88c drivers/usb/core/driver.c:458 ... hub_event+0x1a04/0x4518 drivers/usb/core/hub.c:5953 Fix by moving the card lifecycle out of usb6fire_chip_abort() and into usb6fire_chip_disconnect(). The card pointer is saved in a local before any teardown, snd_card_disconnect() is called first to prevent new opens, URBs are aborted while chip is still valid, and snd_card_free_when_closed() is called last so chip is never accessed after the card may be freed.	2026-04-24	7.8
CVE-2026-31582	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: hwmon: (powerz) Fix use-after-free on USB disconnect After powerz_disconnect() frees the URB and releases the mutex, a subsequent powerz_read() call can acquire the mutex and call powerz_read_data(), which dereferences the freed URB pointer. Fix by: - Setting priv->urb to NULL in powerz_disconnect() so that powerz_read_data() can detect the disconnected state. - Adding a !priv->urb check at the start of powerz_read_data() to return -ENODEV on a disconnected device. - Moving usb_set_intfdata() before hwmon registration so the disconnect handler can always find the priv pointer.	2026-04-24	7.8
CVE-2026-31583	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: media: em28xx: fix use-after-free in em28xx_v4l2_open() em28xx_v4l2_open() reads dev->v4l2 without holding dev->lock, creating a race with em28xx_v4l2_init()'s error path and em28xx_v4l2_fini(), both of which free the em28xx_v4l2 struct and set dev->v4l2 to NULL under dev->lock. This race leads to two issues: - use-after-free in v4l2_fh_init() when accessing vdev->ctrl_handler, since the video_device is embedded in the freed em28xx_v4l2 struct. - NULL pointer dereference in em28xx_resolution_set() when accessing v4l2->norm, since dev->v4l2 has been set to NULL. Fix this by moving the mutex_lock() before the dev->v4l2 read and adding a NULL check for dev->v4l2 under the lock.	2026-04-24	7.8

CVE-2026-31584	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: media: mediatek: vcodec: fix use-after-free in encoder release path The fops_vcodec_release() function frees the context structure (ctx) without first cancelling any pending or running work in ctx->encode_work. This creates a race window where the workqueue handler (mtk_venc_worker) may still be accessing the context memory after it has been freed. Race condition: CPU 0 (release path) CPU 1 (workqueue) ---------- ---------- fops_vcodec_release() v4l2_m2m_ctx_release() v4l2_m2m_cancel_job() // waits for m2m job "done" mtk_venc_worker() v4l2_m2m_job_finish() // m2m job "done" // BUT worker still running! // post-job_finish access: other ctx dereferences // UAF if ctx already freed // returns (job "done") kfree(ctx) // ctx freed Root cause: The v4l2_m2m_ctx_release() only waits for the m2m job lifecycle (via TRANS_RUNNING flag), not the workqueue lifecycle. After v4l2_m2m_job_finish() is called, the m2m framework considers the job complete and v4l2_m2m_ctx_release() returns, but the worker function continues executing and may still access ctx. The work is queued during encode operations via: queue_work(ctx->dev->encode_workqueue, &ctx->encode_work) The worker function accesses ctx->m2m_ctx, ctx->dev, and other ctx fields even after calling v4l2_m2m_job_finish(). This vulnerability was confirmed with KASAN by running an instrumented test module that widens the post-job_finish race window. KASAN detected: BUG: KASAN: slab-use-after-free in mtk_venc_worker+0x159/0x180 Read of size 4 at addr ffff88800326e000 by task kworker/u8:0/12 Workqueue: mtk_vcodec_enc_wq mtk_venc_worker Allocated by task 47: __kasan_kmalloc+0x7f/0x90 fops_vcodec_open+0x85/0x1a0 Freed by task 47: __kasan_slab_free+0x43/0x70 kfree+0xee/0x3a0 fops_vcodec_release+0xb7/0x190 Fix this by calling cancel_work_sync(&ctx->encode_work) before kfree(ctx). This ensures the workqueue handler is both cancelled (if pending) and synchronized (waits for any running handler to complete) before the context is freed. Placement rationale: The fix is placed after v4l2_ctrl_handler_free() and before list_del_init(&ctx->list). At this point, all m2m operations are done (v4l2_m2m_ctx_release() has returned), and we need to ensure the workqueue is synchronized before removing ctx from the list and freeing it. Note: The open error path does NOT need cancel_work_sync() because INIT_WORK() only initializes the work structure - it does not schedule it. Work is only scheduled later during device_run() operations.	2026-04-24	7.8
CVE-2026-31586	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: mm: blk-cgroup: fix use-after-free in cgwb_release_workfn() cgwb_release_workfn() calls css_put(wb->blkcg_css) and then later accesses wb->blkcg_css again via blkcg_unpin_online(). If css_put() drops the last reference, the blkcg can be freed asynchronously (css_free_rwork_fn -> blkcg_css_free -> kfree) before blkcg_unpin_online() dereferences the pointer to access blkcg->online_pin, resulting in a use-after-free: BUG: KASAN: slab-use-after-free in blkcg_unpin_online (./include/linux/instrumented.h:112	2026-04-24	7.8

		./include/linux/atomic/atomic-instrumented.h:400 ./include/linux/refcount.h:389 ./include/linux/refcount.h:432 ./include/linux/refcount.h:450 block/blk-cgroup.c:1367) Write of size 4 at addr ff11000117aa6160 by task kworker/71:1/531 Workqueue: cgwb_release cgwb_release_workfn Call Trace: <TASK> blkcg_unpin_online (./include/linux/instrumented.h:112 ./include/linux/atomic/atomic-instrumented.h:400 ./include/linux/refcount.h:389 ./include/linux/refcount.h:432 ./include/linux/refcount.h:450 block/blk-cgroup.c:1367) cgwb_release_workfn (mm/backing-dev.c:629) process_scheduled_works (kernel/workqueue.c:3278 kernel/workqueue.c:3385) Freed by task 1016: kfree (./include/linux/kasan.h:235 mm/slub.c:2689 mm/slub.c:6246 mm/slub.c:6561) css_free_rwork_fn (kernel/cgroup/cgroup.c:5542) process_scheduled_works (kernel/workqueue.c:3302 kernel/workqueue.c:3385) ** Stack based on commit 66672af7a095 ("Add linux-next specific files for 20260410") I am seeing this crash sporadically in Meta fleet across multiple kernel versions. A full reproducer is available at: https://github.com/leitao/debug/blob/main/reproducers/repro_blkcg_uaf.sh (The race window is narrow. To make it easily reproducible, inject a msleep(100) between css_put() and blkcg_unpin_online() in cgwb_release_workfn(). With that delay and a KASAN-enabled kernel, the reproducer triggers the splat reliably in less than a second.) Fix this by moving blkcg_unpin_online() before css_put(), so the cgwb's CSS reference keeps the blkcg alive while blkcg_unpin_online() accesses it.		
CVE-2026-31587	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ASoC: qcom: q6apm: move component registration to unmanaged version q6apm component registers dais dynamically from ASoC toplology, which are allocated using device managed version apis. Allocating both component and dynamic dais using managed version could lead to incorrect free ordering, dai will be freed while component still holding references to it. Fix this issue by moving component to unmanged version so that the dai pointers are only freeded after the component is removed. ========== BUG: KASAN: slab-use-after-free in snd_soc_del_component_unlocked+0x3d4/0x400 [snd_soc_core] Read of size 8 at addr ffff00084493a6e8 by task kworker/u48:0/3426 Tainted: [W]=WARN Hardware name: LENOVO 21N2ZC5PUS/21N2ZC5PUS, BIOS N42ET57W (1.31) 08/08/2024 Workqueue: pdr_notifier_wq pdr_notifier_work [pdr_interface] Call trace: show_stack+0x28/0x7c (C) dump_stack_lvl+0x60/0x80 print_report+0x160/0x4b4 kasan_report+0xac/0xfc __asan_report_load8_noabort+0x20/0x34 snd_soc_del_component_unlocked+0x3d4/0x400 [snd_soc_core] snd_soc_unregister_component_by_driver+0x50/0x88 [snd_soc_core] devm_component_release+0x30/0x5c [snd_soc_core] devres_release_all+0x13c/0x210 device_unbind_cleanup+0x20/0x190 device_release_driver_internal+0x350/0x468 device_release_driver+0x18/0x30 bus_remove_device+0x1a0/0x35c device_del+0x314/0x7f0 device_unregister+0x20/0xbc apr_remove_device+0x5c/0x7c [apr] device_for_each_child+0xd8/0x160 apr_pd_status+0x7c/0xa8 [apr] pdr_notifier_work+0x114/0x240 [pdr_interface] process_one_work+0x500/0xb70 worker_thread+0x630/0xfb0 kthread+0x370/0x6c0 ret_from_fork+0x10/0x20 Allocated by task 77: kasan_save_stack+0x40/0x68 kasan_save_track+0x20/0x40	2026-04-24	7.8

		kasan_save_alloc_info+0x44/0x58 __kasan_kmalloc+0xbc/0xdc __kmalloc_node_track_caller_noprof+0x1f4/0x620 devm_kmalloc+0x7c/0x1c8 snd_soc_register_dai+0x50/0x4f0 [snd_soc_core] soc_tplg_pcm_elems_load+0x55c/0x1eb8 [snd_soc_core] snd_soc_tplg_component_load+0x4f8/0xb60 [snd_soc_core] audioreach_tplg_init+0x124/0x1fc [snd_q6apm] q6apm_audio_probe+0x10/0x1c [snd_q6apm] snd_soc_component_probe+0x5c/0x118 [snd_soc_core] soc_probe_component+0x44c/0xaf0 [snd_soc_core] snd_soc_bind_card+0xad0/0x2370 [snd_soc_core] snd_soc_register_card+0x3b0/0x4c0 [snd_soc_core] devm_snd_soc_register_card+0x50/0xc8 [snd_soc_core] x1e80100_platform_probe+0x208/0x368 [snd_soc_x1e80100] platform_probe+0xc0/0x188 really_probe+0x188/0x804 __driver_probe_device+0x158/0x358 driver_probe_device+0x60/0x190 __device_attach_driver+0x16c/0x2a8 bus_for_each_drv+0x100/0x194 __device_attach+0x174/0x380 device_initial_probe+0x14/0x20 bus_probe_device+0x124/0x154 deferred_probe_work_func+0x140/0x220 process_one_work+0x500/0xb70 worker_thread+0x630/0xfb0 kthread+0x370/0x6c0 ret_from_fork+0x10/0x20 Freed by task 3426: kasan_save_stack+0x40/0x68 kasan_save_track+0x20/0x40 __kasan_save_free_info+0x4c/0x80 __kasan_slab_free+0x78/0xa0 kfree+0x100/0x4a4 devres_release_all+0x144/0x210 device_unbind_cleanup+0x20/0x190 device_release_driver_internal+0x350/0x468 device_release_driver+0x18/0x30 bus_remove_device+0x1a0/0x35c device_del+0x314/0x7f0 device_unregister+0x20/0xbc apr_remove_device+0x5c/0x7c [apr] device_for_each_child+0xd8/0x160 apr_pd_status+0x7c/0xa8 [apr] pdr_notifier_work+0x114/0x240 [pdr_interface] process_one_work+0x500/0xb70 worker_thread+0x630/0xfb0 kthread+0x370/0x6c0 ret_from_fork+0x10/0x20		
CVE-2026-31597	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ocfs2: fix use-after-free in ocfs2_fault() when VM_FAULT_RETRY filemap_fault() may drop the mmap_lock before returning VM_FAULT_RETRY, as documented in mm/filemap.c: "If our return value has VM_FAULT_RETRY set, it's because the mmap_lock may be dropped before doing I/O or by lock_folio_maybe_drop_mmap()." When this happens, a concurrent munmap() can call remove_vma() and free the vm_area_struct via RCU. The saved 'vma' pointer in ocfs2_fault() then becomes a dangling pointer, and the subsequent trace_ocfs2_fault() call dereferences it -- a use-after-free. Fix this by saving ip_blkno as a plain integer before calling filemap_fault(), and removing vma from the trace event. Since ip_blkno is copied by value before the lock can be dropped, it remains valid regardless of what happens to the vma or inode afterward.	2026-04-24	7.8
CVE-2026-31602	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ALSA: ctxfi: Limit PTP to a single page Commit 391e69143d0a increased CT_PTP_NUM from 1 to 4 to support 256 playback streams, but the additional pages are not used by the card correctly. The CT20K2 hardware already has multiple VMEM_PTPAL registers, but using them separately would require refactoring the entire virtual memory allocation logic.	2026-04-24	7.8

		ct_vm_map() always uses PTEs in vm->ptp[0].area regardless of CT_PTP_NUM. On AMD64 systems, a single PTP covers 512 PTEs (2M). When aggregate memory allocations exceed this limit, ct_vm_map() tries to access beyond the allocated space and causes a page fault: BUG: unable to handle page fault for address: ffffd4ae8a10a000 Oops: Oops: 0002 [#1] SMP PTI RIP: 0010:ct_vm_map+0x17c/0x280 [snd_ctxfi] Call Trace: atc_pcm_playback_prepare+0x225/0x3b0 ct_pcm_playback_prepare+0x38/0x60 snd_pcm_do_prepare+0x2f/0x50 snd_pcm_action_single+0x36/0x90 snd_pcm_action_nonatomic+0xbf/0xd0 snd_pcm_ioctl+0x28/0x40 __x64_sys_ioctl+0x97/0xe0 do_syscall_64+0x81/0x610 entry_SYSCALL_64_after_hwframe+0x76/0x7e Revert CT_PTP_NUM to 1. The 256 SRC_RESOURCE_NUM and playback_count remain unchanged.		
CVE-2026-31627	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: i2c: s3c24xx: check the size of the SMBUS message before using it The first byte of an i2c SMBUS message is the size, and it should be verified to ensure that it is in the range of 0..I2C_SMBUS_BLOCK_MAX before processing it. This is the same logic that was added in commit a6e04f05ce0b ("i2c: tegra: check msg length in SMBUS block read") to the i2c tegra driver.	2026-04-24	7.8
CVE-2026-31630	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: rxrpc: proc: size address buffers for %pISpc output The AF_RXRPC procfs helpers format local and remote socket addresses into fixed 50-byte stack buffers with "%pISpc". That is too small for the longest current-tree IPv6-with-port form the formatter can produce. In lib/vsprintf.c, the compressed IPv6 path uses a dotted-quad tail not only for v4mapped addresses, but also for ISATAP addresses via ipv6_addr_is_isatap(). As a result, a case such as [ffff:ffff:ffff:ffff:0:5efe:255.255.255.255]:65535 is possible with the current formatter. That is 50 visible characters, so 51 bytes including the trailing NUL, which does not fit in the existing char[50] buffers used by net/rxrpc/proc.c. Size the buffers from the formatter's maximum textual form and switch the call sites to scnprintf(). Changes since v1: - correct the changelog to cite the actual maximum current-tree case explicitly - frame the proof around the ISATAP formatting path instead of the earlier mapped-v4 example	2026-04-24	7.8
CVE-2026-31641	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: rxrpc: Fix RxGK token loading to check bounds rxrpc_preparse_xdr_yfs_rxgk() reads the raw key length and ticket length from the XDR token as u32 values and passes each through round_up(x, 4) before using the rounded value for validation and allocation. When the raw length is >= 0xfffffffd, round_up() wraps to 0, so the bounds check and kzalloc both use 0 while the subsequent memcpy still copies the original ~4 GiB value, producing a heap buffer overflow reachable from an unprivileged add_key() call. Fix this by: (1) Rejecting raw key lengths above AFSTOKEN_GK_KEY_MAX and raw ticket lengths above AFSTOKEN_GK_TOKEN_MAX before rounding, consistent with the caps that the RxKAD path already enforces via AFSTOKEN_RK_TIX_MAX. (2) Sizing the flexible-array allocation from the validated raw key length via struct_size_t() instead of the rounded value.	2026-04-24	7.8

		(3) Caching the raw lengths so that the later field assignments and memcpy calls do not re-read from the token, eliminating a class of TOCTOU re-parse. The control path (valid token with lengths within bounds) is unaffected.		
CVE-2026-31644	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net: lan966x: fix use-after-free and leak in lan966x_fdma_reload() When lan966x_fdma_reload() fails to allocate new RX buffers, the restore path restarts DMA using old descriptors whose pages were already freed via lan966x_fdma_rx_free_pages(). Since page_pool_put_full_page() can release pages back to the buddy allocator, the hardware may DMA into memory now owned by other kernel subsystems. Additionally, on the restore path, the newly created page pool (if allocation partially succeeded) is overwritten without being destroyed, leaking it. Fix both issues by deferring the release of old pages until after the new allocation succeeds. Save the old page array before the allocation so old pages can be freed on the success path. On the failure path, the old descriptors, pages and page pool are all still valid, making the restore safe. Also ensure the restore path re-enables NAPI and wakes the netdev, matching the success path.	2026-04-24	7.8
CVE-2026-31648	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: mm: filemap: fix nr_pages calculation overflow in filemap_map_pages() When running stress-ng on my Arm64 machine with v7.0-rc3 kernel, I encountered some very strange crash issues showing up as "Bad page state": " [734.496287] BUG: Bad page state in process stress-ng-env pfn:415735fb [734.496427] page: refcount:0 mapcount:1 mapping:0000000000000000 index:0x4cf316 pfn:0x415735fb [734.496434] flags: 0x57fffe000000800(owner_2|node=1|zone=2|lastcpupid=0x3ffff) [734.496439] raw: 057fffe000000800 0000000000000000 dead000000000122 0000000000000000 [734.496440] raw: 00000000004cf316 0000000000000000 0000000000000000 0000000000000000 [734.496442] page dumped because: nonzero mapcount " After analyzing this page's state, it is hard to understand why the mapcount is not 0 while the refcount is 0, since this page is not where the issue first occurred. By enabling the CONFIG_DEBUG_VM config, I can reproduce the crash as well and captured the first warning where the issue appears: " [734.469226] page: refcount:33 mapcount:0 mapping:00000000bef2d187 index:0x81a0 pfn:0x415735c0 [734.469304] head: order:5 mapcount:0 entire_mapcount:0 nr_pages_mapped:0 pincount:0 [734.469315] memcg:ffff000807a8ec00 [734.469320] aops:ext4_da_aops ino:100b6f dentry name(?):"stress-ng-mmaptorture-9397-0-2736200540" [734.469335] flags: 0x57fffe400000069(locked|uptodate|lru|head|node=1|zone=2|lastcpupid=0x3ffff) ..... [734.469364] page dumped because: VM_WARN_ON_FOLIO((_Generic((page + nr_pages - 1), const struct page *: (const struct folio *)_compound_head(page + nr_pages - 1), struct page *: (struct folio *)_compound_head(page + nr_pages - 1))) != folio) [734.469390] ------------[cut here]------------ [734.469393] WARNING: ./include/linux/rmap.h:351 at folio_add_file_rmap_ptes+0x3b8/0x468, CPU#90: stress-ng-mlock/9430 [734.469551] folio_add_file_rmap_ptes+0x3b8/0x468 (P) [734.469555] set_pte_range+0xd8/0x2f8 [734.469566] filemap_map_folio_range+0x190/0x400 [734.469579] filemap_map_pages+0x348/0x638 [734.469583] do_fault_around+0x140/0x198 ..... [734.469640] el0t_64_sync+0x184/0x188 " The code that triggers the warning is: "VM_WARN_ON_FOLIO(page_folio(page + nr_pages - 1) != folio, folio)", which indicates that set_pte_range() tried to map beyond the large folio's size. By adding more debug information, I found that 'nr_pages' had overflowed	2026-04-24	7.8

		in filemap_map_pages(), causing set_pte_range() to establish mappings for a range exceeding the folio size, potentially corrupting fields of pages that do not belong to this folio (e.g., page->_mapcount). After above analysis, I think the possible race is as follows: CPU 0 CPU 1 filemap_map_pages() ext4_setattr() //get and lock folio with old inode->i_size next_uptodate_folio() ..... //shrink the inode->i_size i_size_write(inode, attr->ia_size); //calculate the end_pgoff with the new inode->i_size file_end = DIV_ROUND_UP(i_size_read(mapping->host), PAGE_SIZE) - 1; end_pgoff = min(end_pgoff, file_end); ..... //nr_pages can be overflowed, cause xas.xa_index > end_pgoff end = folio_next_index(folio) - 1; nr_pages = min(end, end_pgoff) - xas.xa_index + 1; ..... //map large folio filemap_map_folio_range() ..... //truncate folios truncate_pagecache(inode, inode->i_size); To fix this issue, move the 'end_pgoff' calculation before next_uptodate_folio(), so the retrieved folio stays consistent with the file end to avoid ---truncated---		
CVE-2026-31650	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: mmc: vub300: fix use-after-free on disconnect The vub300 driver maintains an explicit reference count for the controller and its driver data and the last reference can in theory be dropped after the driver has been unbound. This specifically means that the controller allocation must not be device managed as that can lead to use-after-free. Note that the lifetime is currently also incorrectly tied the parent USB device rather than interface, which can lead to memory leaks if the driver is unbound without its device being physically disconnected (e.g. on probe deferral). Fix both issues by reverting to non-managed allocation of the controller.	2026-04-24	7.8
CVE-2026-31652	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: mm/damon/stat: deallocate damon_call() failure leaking damon_ctx damon_stat_start() always allocates the module's damon_ctx object (damon_stat_context). Meanwhile, if damon_call() in the function fails, the damon_ctx object is not deallocated. Hence, if the damon_call() is failed, and the user writes Y to “enabled” again, the previously allocated damon_ctx object is leaked. This cannot simply be fixed by deallocating the damon_ctx object when damon_call() fails. That's because damon_call() failure doesn't guarantee the kdamond main function, which accesses the damon_ctx object, is completely finished. In other words, if damon_stat_start() deallocates the damon_ctx object after damon_call() failure, the not-yet-terminated kdamond could access the freed memory (use-after-free). Fix the leak while avoiding the use-after-free by keeping returning damon_stat_start() without deallocating the damon_ctx object after damon_call() failure, but deallocating it when the function is invoked again and the kdamond is completely terminated. If the kdamond is not yet terminated, simply return -EAGAIN, as the kdamond will soon be terminated. The issue was discovered [1] by sashiko.	2026-04-24	7.8
CVE-2026-31656	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: drm/i915/gt: fix refcount underflow in intel_engine_park_heartbeat	2026-04-24	7.8

		A use-after-free / refcount underflow is possible when the heartbeat worker and intel_engine_park_heartbeat() race to release the same engine->heartbeat.systole request. The heartbeat worker reads engine->heartbeat.systole and calls i915_request_put() on it when the request is complete, but clears the pointer in a separate, non-atomic step. Concurrently, a request retirement on another CPU can drop the engine wakeref to zero, triggering __engine_park() -> intel_engine_park_heartbeat(). If the heartbeat timer is pending at that point, cancel_delayed_work() returns true and intel_engine_park_heartbeat() reads the stale non-NULL systole pointer and calls i915_request_put() on it again, causing a refcount underflow: ``` <4> [487.221889] Workqueue: i915-unordered engine_retire [i915] <4> [487.222640] RIP: 0010:refcount_warn_saturate+0x68/0xb0 ... <4> [487.222707] Call Trace: <4> [487.222711] <TASK> <4> [487.222716] intel_engine_park_heartbeat.part.0+0x6f/0x80 [i915] <4> [487.223115] intel_engine_park_heartbeat+0x25/0x40 [i915] <4> [487.223566] __engine_park+0xb9/0x650 [i915] <4> [487.223973] ____intel_wakeref_put_last+0x2e/0xb0 [i915] <4> [487.224408] __intel_wakeref_put_last+0x72/0x90 [i915] <4> [487.224797] intel_context_exit_engine+0x7c/0x80 [i915] <4> [487.225238] intel_context_exit+0xf1/0x1b0 [i915] <4> [487.225695] i915_request_retire.part.0+0x1b9/0x530 [i915] <4> [487.226178] i915_request_retire+0x1c/0x40 [i915] <4> [487.226625] engine_retire+0x122/0x180 [i915] <4> [487.227037] process_one_work+0x239/0x760 <4> [487.227060] worker_thread+0x200/0x3f0 <4> [487.227068] ? __pfx_worker_thread+0x10/0x10 <4> [487.227075] kthread+0x10d/0x150 <4> [487.227083] ? __pfx_kthread+0x10/0x10 <4> [487.227092] ret_from_fork+0x3d4/0x480 <4> [487.227099] ? __pfx_kthread+0x10/0x10 <4> [487.227107] ret_from_fork_asm+0x1a/0x30 <4> [487.227141] </TASK> ``` Fix this by replacing the non-atomic pointer read + separate clear with xchg() in both racing paths. xchg() is a single indivisible hardware instruction that atomically reads the old pointer and writes NULL. This guarantees only one of the two concurrent callers obtains the non-NULL pointer and performs the put, the other gets NULL and skips it. (cherry picked from commit 13238dc0ee4f9ab8dafa2cca7295736191ae2f42)		
CVE-2026-31663	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: xfrm: hold dev ref until after transport_finish NF_HOOK After async crypto completes, xfrm_input_resume() calls dev_put() immediately on re-entry before the skb reaches transport_finish. The skb->dev pointer is then used inside NF_HOOK and its okfn, which can race with device teardown. Remove the dev_put from the async resumption entry and instead drop the reference after the NF_HOOK call in transport_finish, using a saved device pointer since NF_HOOK may consume the skb. This covers NF_DROP, NF_QUEUE and NF_STOLEN paths that skip the okfn. For non-transport exits (decaps, gro, drop) and secondary async return points, release the reference inline when async is set.	2026-04-24	7.8
CVE-2026-31665	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: netfilter: nft_ct: fix use-after-free in timeout object destroy nft_ct_timeout_obj_destroy() frees the timeout object with kfree() immediately after nf_ct_untimeout(), without waiting for an RCU grace period. Concurrent packet processing on other CPUs may still hold RCU-protected references to the timeout object obtained via rcu_dereference() in nf_ct_timeout_data(). Add an rcu_head to struct nf_ct_timeout and use kfree_rcu() to defer freeing until after an RCU grace period, matching the approach already used in nfnetlink_cttimeout.c. KASAN report:	2026-04-24	7.8

		BUG: KASAN: slab-use-after-free in nf_conntrack_tcp_packet+0x1381/0x29d0 Read of size 4 at addr ffff8881035fe19c by task exploit/80 Call Trace: nf_conntrack_tcp_packet+0x1381/0x29d0 nf_conntrack_in+0x612/0x8b0 nf_hook_slow+0x70/0x100 __ip_local_out+0x1b2/0x210 tcp_sendmsg_locked+0x722/0x1580 __sys_sendto+0x2d8/0x320 Allocated by task 75: nft_ct_timeout_obj_init+0xf6/0x290 nft_obj_init+0x107/0x1b0 nf_tables_newobj+0x680/0x9c0 nfnetlink_rcv_batch+0xc29/0xe00 Freed by task 26: nft_obj_destroy+0x3f/0xa0 nf_tables_trans_destroy_work+0x51c/0x5c0 process_one_work+0x2c4/0x5a0		
CVE-2026-31666	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: btrfs: fix incorrect return value after changing leaf in lookup_extent_data_ref() After commit 1618aa3c2e01 ("btrfs: simplify return variables in lookup_extent_data_ref()"), the err and ret variables were merged into a single ret variable. However, when btrfs_next_leaf() returns 0 (success), ret is overwritten from -ENOENT to 0. If the first key in the next leaf does not match (different objectid or type), the function returns 0 instead of -ENOENT, making the caller believe the lookup succeeded when it did not. This can lead to operations on the wrong extent tree item, potentially causing extent tree corruption. Fix this by returning -ENOENT directly when the key does not match, instead of relying on the ret variable.	2026-04-24	7.8
CVE-2026-31667	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: Input: uinput - fix circular locking dependency with ff-core A lockdep circular locking dependency warning can be triggered reproducibly when using a force-feedback gamepad with uinput (for example, playing ELDEN RING under Wine with a Flydigi Vader 5 controller): ff->mutex -> udev->mutex -> input_mutex -> dev->mutex -> ff->mutex The cycle is caused by four lock acquisition paths: 1. ff upload: input_ff_upload() holds ff->mutex and calls uinput_dev_upload_effect() -> uinput_request_submit() -> uinput_request_send(), which acquires udev->mutex. 2. device create: uinput_ioctl_handler() holds udev->mutex and calls uinput_create_device() -> input_register_device(), which acquires input_mutex. 3. device register: input_register_device() holds input_mutex and calls kbd_connect() -> input_register_handle(), which acquires dev->mutex. 4. evdev release: evdev_release() calls input_flush_device() under dev->mutex, which calls input_ff_flush() acquiring ff->mutex. Fix this by introducing a new state_lock spinlock to protect udev->state and udev->dev access in uinput_request_send() instead of acquiring udev->mutex. The function only needs to atomically check device state and queue an input event into the ring buffer via uinput_dev_event() -- both operations are safe under a spinlock (ktime_get_ts64() and wake_up_interruptible() do not sleep). This breaks the ff->mutex -> udev->mutex link since a spinlock is a leaf in the lock ordering and cannot form cycles with mutexes. To keep state transitions visible to uinput_request_send(), protect writes to udev->state in uinput_create_device() and uinput_destroy_device() with the same state_lock spinlock. Additionally, move init_completion(&request->done) from uinput_request_send() to uinput_request_submit() before uinput_request_reserve_slot(). Once the slot is allocated,	2026-04-24	7.8

		uinput_flush_requests() may call complete() on it at any time from the destroy path, so the completion must be initialised before the request becomes visible. Lock ordering after the fix: ff->mutex -> state_lock (spinlock, leaf) udev->mutex -> state_lock (spinlock, leaf) udev->mutex -> input_mutex -> dev->mutex -> ff->mutex (no back-edge)		
CVE-2026-31673	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: af_unix: read UNIX_DIAG_VFS data under unix_state_lock Exact UNIX diag lookups hold a reference to the socket, but not to u->path. Meanwhile, unix_release_sock() clears u->path under unix_state_lock() and drops the path reference after unlocking. Read the inode and device numbers for UNIX_DIAG_VFS while holding unix_state_lock(), then emit the netlink attribute after dropping the lock. This keeps the VFS data stable while the reply is being built.	2026-04-25	7.8
CVE-2026-31675	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net/sched: sch_netem: fix out-of-bounds access in packet corruption In netem_enqueue(), the packet corruption logic uses get_random_u32_below(skb_headlen(skb)) to select an index for modifying skb->data. When an AF_PACKET TX_RING sends fully non-linear packets over an IPIP tunnel, skb_headlen(skb) evaluates to 0. Passing 0 to get_random_u32_below() takes the variable-ceil slow path which returns an unconstrained 32-bit random integer. Using this unconstrained value as an offset into skb->data results in an out-of-bounds memory access. Fix this by verifying skb_headlen(skb) is non-zero before attempting to corrupt the linear data area. Fully non-linear packets will silently bypass the corruption logic.	2026-04-25	7.8
CVE-2026-31678	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: openvswitch: defer tunnel netdev_put to RCU release ovs_netdev_tunnel_destroy() may run after NETDEV_UNREGISTER already detached the device. Dropping the netdev reference in destroy can race with concurrent readers that still observe vport->dev. Do not release vport->dev in ovs_netdev_tunnel_destroy(). Instead, let vport_netdev_free() drop the reference from the RCU callback, matching the non-tunnel destroy path and avoiding additional synchronization under RTNL.	2026-04-25	7.8
CVE-2026-31680	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net: ipv6: flowlabel: defer exclusive option free until RCU teardown `ip6fl_seq_show()` walks the global flowlabel hash under the seq-file RCU read-side lock and prints `fl->opt->opt_nflen` when an option block is present. Exclusive flowlabels currently free `fl->opt` as soon as `fl->users` drops to zero in `fl_release()`. However, the surrounding `struct ip6_flowlabel` remains visible in the global hash table until later garbage collection removes it and `fl_free_rcu()` finally tears it down. A concurrent `/proc/net/ip6_flowlabel` reader can therefore race that early `kfree()` and dereference freed option state, triggering a crash in `ip6fl_seq_show()`. Fix this by keeping `fl->opt` alive until `fl_free_rcu()`. That matches the lifetime already required for the enclosing flowlabel while readers can still reach it under RCU.	2026-04-25	7.8
CVE-2026-31683	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: batman-adv: avoid OGM aggregation when skb tailroom is insufficient When OGM aggregation state is toggled at runtime, an existing forwarded packet may have been allocated with only packet_len bytes, while a later packet can still be selected for aggregation. Appending in this case can hit skb_put overflow conditions.	2026-04-25	7.8

		Reject aggregation when the target skb tailroom cannot accommodate the new packet. The caller then falls back to creating a new forward packet instead of appending.		
CVE-2026-22011	oracle - applications_dba	Vulnerability in the Oracle Applications DBA product of Oracle E-Business Suite (component: ADPatch). Supported versions that are affected are 12.2.3-12.2.15. Difficult to exploit vulnerability allows high privileged attacker with network access via HTTP to compromise Oracle Applications DBA. Successful attacks require human interaction from a person other than the attacker and while the vulnerability is in Oracle Applications DBA, attacks may significantly impact additional products (scope change). Successful attacks of this vulnerability can result in takeover of Oracle Applications DBA. CVSS 3.1 Base Score 7.6 (Confidentiality, Integrity and Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:H/PR:H/UI:R/S:C/C:H/I:H/A:H).	2026-04-21	7.6
CVE-2026-5928	gnu - glibc	Calling the ungetwc function on a FILE stream with wide characters encoded in a character set that has overlaps between its single byte and multi-byte character encodings, in the GNU C Library version 2.43 or earlier, may result in an attempt to read bytes before an allocated buffer, potentially resulting in unintentional disclosure of neighboring data in the heap, or a program crash. A bug in the wide character pushback implementation (_IO_wdefault_pbackfail in libio/wgenops.c) causes ungetwc() to operate on the regular character buffer (fp->_IO_read_ptr) instead of the actual wide-stream read pointer (fp->_wide_data->_IO_read_ptr). The program crash may happen in cases where fp->_IO_read_ptr is not initialized and hence points to NULL. The buffer under-read requires a special situation where the input character encoding is such that there are overlaps between single byte representations and multibyte representations in that encoding, resulting in spurious matches. The spurious match case is not possible in the standard Unicode character sets.	2026-04-20	7.5
CVE-2026-6746	mozilla - multiple products	Use-after-free in the DOM: Core & HTML component. This vulnerability was fixed in Firefox 150, Firefox ESR 115.35, Firefox ESR 140.10, Thunderbird 150, and Thunderbird 140.10.	2026-04-21	7.5
CVE-2026-6747	mozilla - multiple products	Use-after-free in the WebRTC component. This vulnerability was fixed in Firefox 150, Firefox ESR 140.10, Thunderbird 150, and Thunderbird 140.10.	2026-04-21	7.5
CVE-2026-6749	mozilla - multiple products	Information disclosure due to uninitialized memory in the Graphics: Canvas2D component. This vulnerability was fixed in Firefox 150, Firefox ESR 115.35, Firefox ESR 140.10, Thunderbird 150, and Thunderbird 140.10.	2026-04-21	7.5
CVE-2026-6754	mozilla - multiple products	Use-after-free in the JavaScript Engine component. This vulnerability was fixed in Firefox 150, Firefox ESR 115.35, Firefox ESR 140.10, Thunderbird 150, and Thunderbird 140.10.	2026-04-21	7.5
CVE-2026-6756	mozilla - firefox	Mitigation bypass in Firefox for Android. This vulnerability was fixed in Firefox 150.	2026-04-21	7.5
CVE-2026-6758	mozilla - multiple products	Use-after-free in the JavaScript: WebAssembly component. This vulnerability was fixed in Firefox 150 and Thunderbird 150.	2026-04-21	7.5
CVE-2026-6759	mozilla - multiple products	Use-after-free in the Widget: Cocoa component. This vulnerability was fixed in Firefox 150, Firefox ESR 140.10, Thunderbird 150, and Thunderbird 140.10.	2026-04-21	7.5
CVE-2026-6766	mozilla - multiple products	Incorrect boundary conditions in the Libraries component in NSS. This vulnerability was fixed in Firefox 150, Firefox ESR 140.10, Thunderbird 150, and Thunderbird 140.10.	2026-04-21	7.5
CVE-2026-6772	mozilla - multiple products	Incorrect boundary conditions in the Libraries component in NSS. This vulnerability was fixed in Firefox 150, Firefox ESR 115.35, Firefox ESR 140.10, Thunderbird 150, and Thunderbird 140.10.	2026-04-21	7.5
CVE-2026-6773	mozilla - multiple products	Denial-of-service due to integer overflow in the Graphics: WebGPU component. This vulnerability was fixed in Firefox 150 and Thunderbird 150.	2026-04-21	7.5
CVE-2026-6780	mozilla - multiple products	Denial-of-service in the Audio/Video: Playback component. This vulnerability was fixed in Firefox 150 and Thunderbird 150.	2026-04-21	7.5
CVE-2026-6781	mozilla - multiple products	Denial-of-service in the Audio/Video: Playback component. This vulnerability was fixed in Firefox 150 and Thunderbird 150.	2026-04-21	7.5
CVE-2026-6782	mozilla - multiple products	Information disclosure in the IP Protection component. This vulnerability was fixed in Firefox 150 and Thunderbird 150.	2026-04-21	7.5
CVE-2026-6784	mozilla - multiple products	Memory safety bugs present in Firefox 149 and Thunderbird 149. Some of these bugs showed evidence of memory corruption and we presume that with enough effort some of these could have been exploited to run arbitrary code. This vulnerability was fixed in Firefox 150 and Thunderbird 150.	2026-04-21	7.5
CVE-2026-22010	oracle - multiple products	Vulnerability in the Oracle Financial Services Analytical Applications Infrastructure product of Oracle Financial Services Applications (component: Platform). Supported versions that are affected are 8.0.7.9, 8.0.8.7 and 8.1.2.5. Easily exploitable vulnerability allows unauthenticated attacker with network access via HTTP to compromise Oracle Financial Services Analytical Applications Infrastructure. Successful attacks of this vulnerability can result in unauthorized access to critical data or complete access to all Oracle Financial Services Analytical Applications Infrastructure accessible data. CVSS 3.1 Base Score 7.5 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N).	2026-04-21	7.5
CVE-2026-22016	oracle - multiple products	Vulnerability in the Oracle Java SE, Oracle GraalVM for JDK, Oracle GraalVM Enterprise Edition product of Oracle Java SE (component: JAXP). Supported versions that are affected are Oracle Java SE: 8u481, 8u481-b50, 8u481-perf, 11.0.30, 17.0.18, 21.0.10, 25.0.2, 26; Oracle GraalVM for JDK: 17.0.18 and 21.0.10; Oracle GraalVM Enterprise Edition: 21.3.17. Easily exploitable vulnerability allows unauthenticated attacker with network access via multiple protocols to compromise Oracle Java SE, Oracle GraalVM for JDK, Oracle GraalVM Enterprise Edition. Successful attacks of this vulnerability can result in unauthorized access to critical data or complete access to all Oracle Java SE, Oracle GraalVM for JDK, Oracle GraalVM Enterprise Edition accessible data. Note: This vulnerability can be exploited by using APIs in the specified Component, e.g., through a web service which supplies data to the APIs. This vulnerability also applies to Java deployments, typically in clients running sandboxed Java Web Start applications or sandboxed Java applets, that load and run untrusted code (e.g., code that comes from the internet) and rely on the Java sandbox for security. CVSS 3.1 Base Score 7.5 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N).	2026-04-21	7.5
CVE-2026-34282	oracle - multiple products	Vulnerability in the Oracle Java SE, Oracle GraalVM for JDK, Oracle GraalVM Enterprise Edition product of Oracle Java SE (component: Networking). Supported versions that are affected are Oracle Java SE: 8u481-perf, 11.0.30, 17.0.18, 21.0.10, 25.0.2, 26; Oracle GraalVM for JDK: 17.0.18	2026-04-21	7.5

		and 21.0.10; Oracle GraalVM Enterprise Edition: 21.3.17. Easily exploitable vulnerability allows unauthenticated attacker with network access via multiple protocols to compromise Oracle Java SE, Oracle GraalVM for JDK, Oracle GraalVM Enterprise Edition. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of Oracle Java SE, Oracle GraalVM for JDK, Oracle GraalVM Enterprise Edition. Note: This vulnerability can be exploited by using APIs in the specified Component, e.g., through a web service which supplies data to the APIs. This vulnerability also applies to Java deployments, typically in clients running sandboxed Java Web Start applications or sandboxed Java applets, that load and run untrusted code (e.g., code that comes from the internet) and rely on the Java sandbox for security. CVSS 3.1 Base Score 7.5 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:H).		
CVE-2026-34290	oracle - identity_manager _connector	Vulnerability in the Oracle Identity Manager Connector product of Oracle Fusion Middleware (component: Core). The supported version that is affected is 12.2.1.4.0. Easily exploitable vulnerability allows unauthenticated attacker with network access via TCP to compromise Oracle Identity Manager Connector. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of Oracle Identity Manager Connector. CVSS 3.1 Base Score 7.5 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:H).	2026-04-21	7.5
CVE-2026-34297	oracle - hcm_common_arc hitecture	Vulnerability in the Oracle HCM Common Architecture product of Oracle E-Business Suite (component: Knowledge Integration). Supported versions that are affected are 12.2.3-12.2.15. Easily exploitable vulnerability allows unauthenticated attacker with network access via HTTP to compromise Oracle HCM Common Architecture. Successful attacks of this vulnerability can result in unauthorized access to critical data or complete access to all Oracle HCM Common Architecture accessible data. CVSS 3.1 Base Score 7.5 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N).	2026-04-21	7.5
CVE-2026-34305	oracle - multiple products	Vulnerability in the Oracle WebLogic Server product of Oracle Fusion Middleware (component: Web Services). Supported versions that are affected are 12.2.1.4.0, 14.1.1.0.0, 14.1.2.0.0 and 15.1.1.0.0. Easily exploitable vulnerability allows unauthenticated attacker with network access via HTTP to compromise Oracle WebLogic Server. Successful attacks of this vulnerability can result in unauthorized access to critical data or complete access to all Oracle WebLogic Server accessible data. CVSS 3.1 Base Score 7.5 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N).	2026-04-21	7.5
CVE-2026-34310	oracle - multiple products	Vulnerability in the Oracle Financial Services Analytical Applications Infrastructure product of Oracle Financial Services Applications (component: Platform). Supported versions that are affected are 8.0.7.9, 8.0.8.7 and 8.1.2.5. Easily exploitable vulnerability allows unauthenticated attacker with network access via HTTP to compromise Oracle Financial Services Analytical Applications Infrastructure. Successful attacks of this vulnerability can result in unauthorized access to critical data or complete access to all Oracle Financial Services Analytical Applications Infrastructure accessible data. CVSS 3.1 Base Score 7.5 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N).	2026-04-21	7.5
CVE-2026-34320	oracle - financial_services _customer_screen ing	Vulnerability in the Oracle Financial Services Customer Screening product of Oracle Financial Services Applications (component: User Interface). The supported version that is affected is 8.1.2.8.0. Easily exploitable vulnerability allows unauthenticated attacker with network access via HTTP to compromise Oracle Financial Services Customer Screening. Successful attacks of this vulnerability can result in unauthorized access to critical data or complete access to all Oracle Financial Services Customer Screening accessible data. CVSS 3.1 Base Score 7.5 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N).	2026-04-21	7.5
CVE-2026-35230	oracle - vm_virtualbox	Vulnerability in the Oracle VM VirtualBox product of Oracle Virtualization (component: Core). The supported version that is affected is 7.2.6. Difficult to exploit vulnerability allows high privileged attacker with logon to the infrastructure where Oracle VM VirtualBox executes to compromise Oracle VM VirtualBox. While the vulnerability is in Oracle VM VirtualBox, attacks may significantly impact additional products (scope change). Successful attacks of this vulnerability can result in takeover of Oracle VM VirtualBox. CVSS 3.1 Base Score 7.5 (Confidentiality, Integrity and Availability impacts). CVSS Vector: (CVSS:3.1/AV:L/AC:H/PR:H/UI:N/S:C/C:H/I:H/A:H).	2026-04-21	7.5
CVE-2026-35231	oracle - financial_services _transaction_filter ing	Vulnerability in the Oracle Financial Services Transaction Filtering product of Oracle Financial Services Applications (component: User Interface). The supported version that is affected is 8.1.2.8.0. Easily exploitable vulnerability allows unauthenticated attacker with network access via HTTP to compromise Oracle Financial Services Transaction Filtering. Successful attacks of this vulnerability can result in unauthorized access to critical data or complete access to all Oracle Financial Services Transaction Filtering accessible data. CVSS 3.1 Base Score 7.5 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N).	2026-04-21	7.5
CVE-2026-35242	oracle - vm_virtualbox	Vulnerability in the Oracle VM VirtualBox product of Oracle Virtualization (component: Core). The supported version that is affected is 7.2.6. Difficult to exploit vulnerability allows high privileged attacker with logon to the infrastructure where Oracle VM VirtualBox executes to compromise Oracle VM VirtualBox. While the vulnerability is in Oracle VM VirtualBox, attacks may significantly impact additional products (scope change). Successful attacks of this vulnerability can result in takeover of Oracle VM VirtualBox. CVSS 3.1 Base Score 7.5 (Confidentiality, Integrity and Availability impacts). CVSS Vector: (CVSS:3.1/AV:L/AC:H/PR:H/UI:N/S:C/C:H/I:H/A:H).	2026-04-21	7.5
CVE-2026-35245	oracle - vm_virtualbox	Vulnerability in the Oracle VM VirtualBox product of Oracle Virtualization (component: Core). The supported version that is affected is 7.2.6. Easily exploitable vulnerability allows unauthenticated attacker with network access via RDP to compromise Oracle VM VirtualBox. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of Oracle VM VirtualBox. CVSS 3.1 Base Score 7.5 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:H).	2026-04-21	7.5
CVE-2026-35246	oracle - vm_virtualbox	Vulnerability in the Oracle VM VirtualBox product of Oracle Virtualization (component: Core). The supported version that is affected is 7.2.6. Difficult to exploit vulnerability allows high privileged attacker with logon to the infrastructure where Oracle VM VirtualBox executes to compromise Oracle VM VirtualBox. While the vulnerability is in Oracle VM VirtualBox, attacks may significantly impact additional products (scope change). Successful attacks of this vulnerability can result in		

		takeover of Oracle VM VirtualBox. CVSS 3.1 Base Score 7.5 (Confidentiality, Integrity and Availability impacts). CVSS Vector: (CVSS:3.1/AV:L/AC:H/PR:H/UI:N/S:C/C:H/I:H/A:H).		
CVE-2026-35251	oracle - vm_virtualbox	Vulnerability in the Oracle VM VirtualBox product of Oracle Virtualization (component: Core). The supported version that is affected is 7.2.6. Difficult to exploit vulnerability allows high privileged attacker with logon to the infrastructure where Oracle VM VirtualBox executes to compromise Oracle VM VirtualBox. While the vulnerability is in Oracle VM VirtualBox, attacks may significantly impact additional products (scope change). Successful attacks of this vulnerability can result in takeover of Oracle VM VirtualBox. CVSS 3.1 Base Score 7.5 (Confidentiality, Integrity and Availability impacts). CVSS Vector: (CVSS:3.1/AV:L/AC:H/PR:H/UI:N/S:C/C:H/I:H/A:H).	2026-04-21	7.5
CVE-2026-22753	vmware - spring_security	Vulnerability in Spring Spring Security. If an application is using securityMatchers(String) and a PathPatternRequestMatcher.Builder bean to prepend a servlet path, matching requests to that filter chain may fail and its related security components will not be exercised as intended by the application. This can lead to the authentication, authorization, and other security controls being rendered inactive on intended requests.This issue affects Spring Security: from 7.0.0 through 7.0.4.	2026-04-22	7.5
CVE-2026-22754	vmware - spring_security	Vulnerability in Spring Spring Security. If an application uses <sec:intercept-url servlet-path="/servlet-path" pattern="/endpoint/**"/> to define the servlet path for computing a path matcher, then the servlet path is not included and the related authorization rules are not exercised. This can lead to an authorization bypass.This issue affects Spring Security: from 7.0.0 through 7.0.4.	2026-04-22	7.5
CVE-2026-6857	red hat - multiple products	A flaw was found in camel-infinispan. This vulnerability involves unsafe deserialization in the ProtoStream remote aggregation repository. A remote attacker with low privileges could exploit this by sending specially crafted data, leading to arbitrary code execution. This allows the attacker to gain full control over the affected system, impacting its confidentiality, integrity, and availability.	2026-04-22	7.5
CVE-2026-31467	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: erofs: add GFP_NOIO in the bio completion if needed The bio completion path in the process context (e.g. dm-verity) will directly call into decompression rather than trigger another workqueue context for minimal scheduling latencies, which can then call vm_map_ram() with GFP_KERNEL. Due to insufficient memory, vm_map_ram() may generate memory swapping I/O, which can cause submit_bio_wait to deadlock in some scenarios. Trimmed down the call stack, as follows: f2fs_submit_read_io submit_bio //bio_list is initialized. mmc_blk_mq_recovery z_erofs_endio vm_map_ram __pte_alloc_kernel __alloc_pages_direct_reclaim shrink_folio_list __swap_writepage submit_bio_wait //bio_list is non-NULL, hang!!! Use memalloc_noio_{save,restore}() to wrap up this path.	2026-04-22	7.5
CVE-2026-31477	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ksmbd: fix memory leaks and NULL deref in smb2_lock() smb2_lock() has three error handling issues after list_del() detaches smb_lock from lock_list at no_check_cl: 1) If vfs_lock_file() returns an unexpected error in the non-UNLOCK path, goto out leaks smb_lock and its flock because the out: handler only iterates lock_list and rollback_list, neither of which contains the detached smb_lock. 2) If vfs_lock_file() returns -ENOENT in the UNLOCK path, goto out leaks smb_lock and flock for the same reason. The error code returned to the dispatcher is also stale. 3) In the rollback path, smb_flock_init() can return NULL on allocation failure. The result is dereferenced unconditionally, causing a kernel NULL pointer dereference. Add a NULL check to prevent the crash and clean up the bookkeeping; the VFS lock itself cannot be rolled back without the allocation and will be released at file or connection teardown. Fix cases 1 and 2 by hoisting the locks_free_lock()/kfree() to before the if(!rc) check in the UNLOCK branch so all exit paths share one free site, and by freeing smb_lock and flock before goto out in the non-UNLOCK branch. Propagate the correct error code in both cases. Fix case 3 by wrapping the VFS unlock in an if(rlock) guard and adding a NULL check for locks_free_lock(rlock) in the shared cleanup. Found via call-graph analysis using sqry.	2026-04-22	7.5

CVE-2026-3621	ibm - WebSphere Application Server - Liberty	IBM WebSphere Application Server - Liberty 17.0.0.3 through 26.0.0.4 IBM WebSphere Application Server Liberty is vulnerable to identity spoofing under limited conditions when an application is deployed without authentication and authorization configured.	2026-04-23	7.5
CVE-2026-21728	grafana - Tempo	Tempo queries with large limits can cause large memory allocations which can impact the availability of the service, depending on its deployment strategy. Mitigation can be done by setting max_result_limit in the search config, e.g. to 262144 (2^18).	2026-04-24	7.5
CVE-2026-31538	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: smb: server: make use of smbdirect_socket.recv_io.credits.available The logic off managing recv credits by counting posted recv_io and granted credits is racy. That's because the peer might already consumed a credit, but between receiving the incoming recv at the hardware and processing the completion in the 'recv_done' functions we likely have a window where we grant credits, which don't really exist. So we better have a decicated counter for the available credits, which will be incremented when we posted new recv buffers and drained when we grant the credits to the peer. This fixes regression Namjae reported with the 6.18 release.	2026-04-24	7.5
CVE-2026-31539	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: smb: smbdirect: introduce smbdirect_socket.recv_io.credits.available The logic off managing recv credits by counting posted recv_io and granted credits is racy. That's because the peer might already consumed a credit, but between receiving the incoming recv at the hardware and processing the completion in the 'recv_done' functions we likely have a window where we grant credits, which don't really exist. So we better have a decicated counter for the available credits, which will be incremented when we posted new recv buffers and drained when we grant the credits to the peer.	2026-04-24	7.5
CVE-2026-31552	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: wifi: wlcore: Return -ENOMEM instead of -EAGAIN if there is not enough headroom Since upstream commit e75665dd0968 ("wifi: wlcore: ensure skb headroom before skb_push"), wl1271_tx_allocate() and with it wl1271_prepare_tx_frame() returns -EAGAIN if pskb_expand_head() fails. However, in wlcore_tx_work_locked(), a return value of -EAGAIN from wl1271_prepare_tx_frame() is interpreted as the aggregation buffer being full. This causes the code to flush the buffer, put the skb back at the head of the queue, and immediately retry the same skb in a tight while loop. Because wlcore_tx_work_locked() holds wl->mutex, and the retry happens immediately with GFP_ATOMIC, this will result in an infinite loop and a CPU soft lockup. Return -ENOMEM instead so the packet is dropped and the loop terminates. The problem was found by an experimental code review agent based on gemini-3.1-pro while reviewing backports into v6.18.y.	2026-04-24	7.5
CVE-2026-31557	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: nvmet: move async event work off nvmet-wq For target nvmet_ctrl_free() flushes ctrl->async_event_work. If nvmet_ctrl_free() runs on nvmet-wq, the flush re-enters workqueue completion for the same worker:- A. Async event work queued on nvmet-wq (prior to disconnect): nvmet_execute_async_event() queue_work(nvmet_wq, &ctrl->async_event_work) nvmet_add_async_event() queue_work(nvmet_wq, &ctrl->async_event_work) B. Full pre-work chain (RDMA CM path):	2026-04-24	7.5

		nvmet_rdma_cm_handler() nvmet_rdma_queue_disconnect() __nvmet_rdma_queue_disconnect() queue_work(nvmet_wq, &queue->release_work) process_one_work() lock((wq_completion)nvmet-wq) <--------- 1st nvmet_rdma_release_queue_work() C. Recursive path (same worker): nvmet_rdma_release_queue_work() nvmet_rdma_free_queue() nvmet_sq_destroy() nvmet_ctrl_put() nvmet_ctrl_free() flush_work(&ctrl->async_event_work) __flush_work() touch_wq_lockdep_map() lock((wq_completion)nvmet-wq) <--------- 2nd Lockdep splat: == WARNING: possible recursive locking detected 6.19.0-rc3nvme+ #14 Tainted: G N -- kworker/u192:42/44933 is trying to acquire lock: ffff888118a00948 ((wq_completion)nvmet-wq){+.+.}-{0:0}, at: touch_wq_lockdep_map+0x26/0x90 but task is already holding lock: ffff888118a00948 ((wq_completion)nvmet-wq){+.+.}-{0:0}, at: process_one_work+0x53e/0x660 3 locks held by kworker/u192:42/44933: #0: ffff888118a00948 ((wq_completion)nvmet-wq){+.+.}-{0:0}, at: process_one_work+0x53e/0x660 #1: ffffc9000e6cbe28 ((work_completion)(&queue->release_work)){+.+.}-{0:0}, at: process_one_work+0x1c5/0x660 #2: ffffffff82d4db60 (rcu_read_lock){....}-{1:3}, at: __flush_work+0x62/0x530 Workqueue: nvmet-wq nvmet_rdma_release_queue_work [nvmet_rdma] Call Trace: __flush_work+0x268/0x530 nvmet_ctrl_free+0x140/0x310 [nvmet] nvmet_cq_put+0x74/0x90 [nvmet] nvmet_rdma_free_queue+0x23/0xe0 [nvmet_rdma] nvmet_rdma_release_queue_work+0x19/0x50 [nvmet_rdma] process_one_work+0x206/0x660 worker_thread+0x184/0x320 kthread+0x10c/0x240 ret_from_fork+0x319/0x390 Move async event work to a dedicated nvmet-aen-wq to avoid reentrant flush on nvmet-wq.		
[CVE-2026-31563](#)	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net: macb: Use dev_consume_skb_any() to free TX SKBs The napi_consume_skb() function is not intended to be called in an IRQ disabled context. However, after commit 6bc8a5098bf4 ("net: macb: Fix tx_ptr_lock locking"), the freeing of TX SKBs is performed with IRQs disabled. To resolve the following call trace, use dev_consume_skb_any() for freeing TX SKBs: WARNING: kernel/softirq.c:430 at __local_bh_enable_ip+0x174/0x188, CPU#0: ksoftirqd/0/15 Modules linked in: CPU: 0 UID: 0 PID: 15 Comm: ksoftirqd/0 Not tainted 7.0.0-rc4-next-20260319-yocto-standard- dirty #37 PREEMPT Hardware name: ZynqMP ZCU102 Rev1.1 (DT) pstate: 200000c5 (nzCv daIF -PAN -UAO -TCO -DIT -SSBS BTYPE=--) pc : __local_bh_enable_ip+0x174/0x188 lr : local_bh_enable+0x24/0x38 sp : ffff800082b3bb10 x29: ffff800082b3bb10 x28: ffff0008031f3c00 x27: 000000000011ede0 x26: ffff000800a7ff00 x25: ffff800083937ce8 x24: 0000000000017a80 x23: ffff000803243a78 x22: 0000000000000040 x21: 0000000000000000 x20: ffff000800394c80 x19: 0000000000000200 x18: 0000000000000001 x17: 0000000000000001 x16: ffff000803240000 x15: 0000000000000000 x14: ffffffffffffffff x13: 0000000000000028 x12: ffff000800395650 x11: ffff8000821d1528 x10: ffff800081c2bc08 x9 : ffff800081c1e258 x8 : 0000000100000301 x7 : ffff8000810426ec x6 : 0000000000000000 x5 : 0000000000000001 x4 : 0000000000000001 x3 : 0000000000000000 x2 : 0000000000000008 x1 : 0000000000000200 x0 : ffff8000810428dc	2026-04-24	7.5

		Call trace: __local_bh_enable_ip+0x174/0x188 (P) local_bh_enable+0x24/0x38 skb_attempt_defer_free+0x190/0x1d8 napi_consume_skb+0x58/0x108 macb_tx_poll+0x1a4/0x558 __napi_poll+0x50/0x198 net_rx_action+0x1f4/0x3d8 handle_softirqs+0x16c/0x560 run_ksoftirqd+0x44/0x80 smpboot_thread_fn+0x1d8/0x338 kthread+0x120/0x150 ret_from_fork+0x10/0x20 irq event stamp: 29751 hardirqs last enabled at (29750): [<ffff8000813be184>] _raw_spin_unlock_irqrestore+0x44/0x88 hardirqs last disabled at (29751): [<ffff8000813bdf60>] _raw_spin_lock_irqsave+0x38/0x98 softirqs last enabled at (29150): [<ffff8000800f1aec>] handle_softirqs+0x504/0x560 softirqs last disabled at (29153): [<ffff8000800f2fec>] run_ksoftirqd+0x44/0x80		
CVE-2026-31598	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ocfs2: fix possible deadlock between unlink and dio_end_io_write ocfs2_unlink takes orphan dir inode_lock first and then ip_alloc_sem, while in ocfs2_dio_end_io_write, it acquires these locks in reverse order. This creates an ABBA lock ordering violation on lock classes ocfs2_sysfile_lock_key[ORPHAN_DIR_SYSTEM_INODE] and ocfs2_file_ip_alloc_sem_key. Lock Chain #0 (orphan dir inode_lock -> ip_alloc_sem): ocfs2_unlink ocfs2_prepare_orphan_dir ocfs2_lookup_lock_orphan_dir inode_lock(orphan_dir_inode) <- lock A __ocfs2_prepare_orphan_dir ocfs2_prepare_dir_for_insert ocfs2_extend_dir ocfs2_expand_inline_dir down_write(&oi->ip_alloc_sem) <- Lock B Lock Chain #1 (ip_alloc_sem -> orphan dir inode_lock): ocfs2_dio_end_io_write down_write(&oi->ip_alloc_sem) <- Lock B ocfs2_del_inode_from_orphan() inode_lock(orphan_dir_inode) <- Lock A Deadlock Scenario: CPU0 (unlink) CPU1 (dio_end_io_write) ------ ------ inode_lock(orphan_dir_inode) down_write(ip_alloc_sem) down_write(ip_alloc_sem) inode_lock(orphan_dir_inode) Since ip_alloc_sem is to protect allocation changes, which is unrelated with operations in ocfs2_del_inode_from_orphan. So move ocfs2_del_inode_from_orphan out of ip_alloc_sem to fix the deadlock.	2026-04-24	7.5
CVE-2026-31600	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: arm64: mm: Handle invalid large leaf mappings correctly It has been possible for a long time to mark ptes in the linear map as invalid. This is done for secretmem, kfence, realm dma memory un/share, and others, by simply clearing the PTE_VALID bit. But until commit a166563e7ec37 ("arm64: mm: support large block mapping when rodata=full") large leaf mappings were never made invalid in this way. It turns out various parts of the code base are not equipped to handle invalid large leaf mappings (in the way they are currently encoded) and I've observed a kernel panic while booting a realm guest on a BBML2_NOABORT system as a result: [15.432706] software IO TLB: Memory encryption is active and system is using DMA bounce buffers [15.476896] Unable to handle kernel paging request at virtual address ffff000019600000 [15.513762] Mem abort info: [15.527245] ESR = 0x0000000096000046 [15.548553] EC = 0x25: DABT (current EL), IL = 32 bits [15.572146] SET = 0, FnV = 0 [15.592141] EA = 0, S1PTW = 0 [15.612694] FSC = 0x06: level 2 translation fault	2026-04-24	7.5

		[15.640644] Data abort info: [15.661983] ISV = 0, ISS = 0x00000046, ISS2 = 0x00000000 [15.694875] CM = 0, WnR = 1, TnD = 0, TagAccess = 0 [15.723740] GCS = 0, Overlay = 0, DirtyBit = 0, Xs = 0 [15.755776] swapper pgtable: 4k pages, 48-bit VAs, pgdp=0000000081f3f000 [15.800410] [ffff000019600000] pgd=0000000000000000, p4d=180000009ffff403, pud=180000009fffe403, pmd=00e8000199600704 [15.855046] Internal error: Oops: 0000000096000046 [#1] SMP [15.886394] Modules linked in: [15.900029] CPU: 0 UID: 0 PID: 1 Comm: swapper/0 Not tainted 7.0.0-rc4-dirty #4 PREEMPT [15.935258] Hardware name: linux,dummy-virt (DT) [15.955612] pstate: 21400005 (nzCv daif +PAN -UAO -TCO +DIT -SSBS BTYPE=--) [15.986009] pc : __pi_memcpy_generic+0x128/0x22c [16.006163] lr : swiotlb_bounce+0xf4/0x158 [16.024145] sp : ffff80008000b8f0 [16.038896] x29: ffff80008000b8f0 x28: 0000000000000000 x27: 0000000000000000 [16.069953] x26: ffffb3976d261ba8 x25: 0000000000000000 x24: ffff000019600000 [16.100876] x23: 0000000000000001 x22: ffff0000043430d0 x21: 0000000000007ff0 [16.131946] x20: 0000000084570010 x19: 0000000000000000 x18: ffff00001ffe3fcc [16.163073] x17: 0000000000000000 x16: 00000000003fffff x15: 646e612065766974 [16.194131] x14: 0000000000000000 x13: 0000000000000000 x12: 0000000000000000 [16.225059] x11: 0000000000000000 x10: 0000000000000010 x9 : 0000000000000018 [16.256113] x8 : 0000000000000018 x7 : 0000000000000000 x6 : 0000000000000000 [16.287203] x5 : ffff000019607ff0 x4 : ffff000004578000 x3 : ffff000019600000 [16.318145] x2 : 0000000000007ff0 x1 : ffff000004570010 x0 : ffff000019600000 [16.349071] Call trace: [16.360143] __pi_memcpy_generic+0x128/0x22c (P) [16.380310] swiotlb_tbl_map_single+0x154/0x2b4 [16.400282] swiotlb_map+0x5c/0x228 [16.415984] dma_map_phys+0x244/0x2b8 [16.432199] dma_map_page_attrs+0x44/0x58 [16.449782] virtqueue_map_page_attrs+0x38/0x44 [16.469596] virtqueue_map_single_attrs+0xc0/0x130 [16.490509] virtnet_rq_alloc.isra.0+0xa4/0x1fc [16.510355] try_fill_recv+0x2a4/0x584 [16.526989] virtnet_open+0xd4/0x238 [16.542775] __dev_open+0x110/0x24c [16.558280] __dev_change_flags+0x194/0x20c [16.576879] netif_change_flags+0x24/0x6c [16.594489] dev_change_flags+0x48/0x7c [16.611462] ip_auto_config+0x258/0x1114 [16.628727] do_one_initcall+0x80/0x1c8 [16.645590] kernel_init_freeable+0x208/0x2f0 [16.664917] kernel_init+0x24/0x1e0 [16.680295] ret_from_fork+0x10/0x20 [16.696369] Code: 927cec03 cb0e0021 8b0e0042 a9411c26 (a900340c) [16.723106] ---[end trace 0000000000000000]--- [16.752866] Kernel panic - not syncing: Attempted to kill init! exitcode=0x0000000b [16.792556] Kernel Offset: 0x3396ea200000 from 0xffff8000800000 ---truncated---		
[CVE-2026-31612](#)	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ksmbd: validate EaNameLength in smb2_get_ea() smb2_get_ea() reads ea_req->EaNameLength from the client request and passes it directly to strncmp() as the comparison length without verifying that the length of the name really is the size of the input buffer received. Fix this up by properly checking the size of the name based on the value received and the overall size of the request, to prevent a later strncmp() call to use the length as a "trusted" size of the buffer. Without this check, uninitialized heap values might be slowly leaked to the client.	2026-04-24	7.5
[CVE-2026-31635](#)	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: rxrpc: fix oversized RESPONSE authenticator length check rxgk_verify_response() decodes auth_len from the packet and is supposed to verify that it fits in the remaining bytes. The existing check is inverted, so oversized RESPONSE authenticators are accepted and passed to rxgk_decrypt_skb(), which can later reach skb_to_sgvec() with an impossible length and hit BUG_ON(len). Decoded from the original latest-net reproduction logs with scripts/decode_stacktrace.sh: RIP: __skb_to_sgvec() [net/core/skbuff.c:5285 (discriminator 1)] Call Trace:	2026-04-24	7.5

		skb_to_sgvec() [net/core/skbuff.c:5305] rxgk_decrypt_skb() [net/rxrpc/rxgk_common.h:81] rxgk_verify_response() [net/rxrpc/rxgk.c:1268] rxrpc_process_connection() [net/rxrpc/conn_event.c:266 net/rxrpc/conn_event.c:364 net/rxrpc/conn_event.c:386] process_one_work() [kernel/workqueue.c:3281] worker_thread() [kernel/workqueue.c:3353 kernel/workqueue.c:3440] kthread() [kernel/kthread.c:436] ret_from_fork() [arch/x86/kernel/process.c:164] Reject authenticator lengths that exceed the remaining packet payload.		
CVE-2026-31638	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: rxrpc: Only put the call ref if one was acquired rxrpc_input_packet_on_conn() can process a to-client packet after the current client call on the channel has already been torn down. In that case chan->call is NULL, rxrpc_try_get_call() returns NULL and there is no reference to drop. The client-side implicit-end error path does not account for that and unconditionally calls rxrpc_put_call(). This turns a protocol error path into a kernel crash instead of rejecting the packet. Only drop the call reference if one was actually acquired. Keep the existing protocol error handling unchanged.	2026-04-24	7.5
CVE-2026-31640	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: rxrpc: Fix use of wrong skb when comparing queued RESP challenge serial In rxrpc_post_response(), the code should be comparing the challenge serial number from the cached response before deciding to switch to a newer response, but looks at the newer packet private data instead, rendering the comparison always false. Fix this by switching to look at the older packet. Fix further[1] to substitute the new packet in place of the old one if newer and also to release whichever we don't use.	2026-04-24	7.5
CVE-2026-31662	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: tipc: fix bc_ackers underflow on duplicate GRP_ACK_MSG The GRP_ACK_MSG handler in tipc_group_proto_rcv() currently decrements bc_ackers on every inbound group ACK, even when the same member has already acknowledged the current broadcast round. Because bc_ackers is a u16, a duplicate ACK received after the last legitimate ACK wraps the counter to 65535. Once wrapped, tipc_group_bc_cong() keeps reporting congestion and later group broadcasts on the affected socket stay blocked until the group is recreated. Fix this by ignoring duplicate or stale ACKs before touching bc_acked or bc_ackers. This makes repeated GRP_ACK_MSG handling idempotent and prevents the underflow path.	2026-04-24	7.5
CVE-2026-31676	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: rxrpc: only handle RESPONSE during service challenge Only process RESPONSE packets while the service connection is still in RXRPC_CONN_SERVICE_CHALLENGING. Check that state under state_lock before running response verification and security initialization, then use a local secured flag to decide whether to queue the secured-connection work after the state transition. This keeps duplicate or late RESPONSE packets from re-running the setup path and removes the unlocked post-transition state test.	2026-04-25	7.5
CVE-2026-6751	mozilla - multiple products	Uninitialized memory in the Audio/Video: Web Codecs component. This vulnerability was fixed in Firefox 150, Firefox ESR 140.10, Thunderbird 150, and Thunderbird 140.10.	2026-04-21	7.3
CVE-2026-6752	mozilla - multiple products	Incorrect boundary conditions in the WebRTC component. This vulnerability was fixed in Firefox 150, Firefox ESR 115.35, Firefox ESR 140.10, Thunderbird 150, and Thunderbird 140.10.	2026-04-21	7.3
CVE-2026-6753	mozilla - multiple products	Incorrect boundary conditions in the WebRTC component. This vulnerability was fixed in Firefox 150, Firefox ESR 140.10, Thunderbird 150, and Thunderbird 140.10.	2026-04-21	7.3
CVE-2026-40542	apache - httpclient	Missing critical step in authentication in Apache HttpClient 5.6 allows an attacker to cause the client to accept SCRAM-SHA-256 authentication without proper mutual authentication verification. Users are recommended to upgrade to version 5.6.1, which fixes this issue.	2026-04-22	7.3
CVE-2026-41134	microsoft - kiota	Kiota is an OpenAPI based HTTP Client code generator. Versions prior to 1.31.1 are affected by a code-generation literal injection vulnerability in multiple writer sinks (for example:	2026-04-22	7.3

		serialization/deserialization keys, path/query parameter mappings, URL template metadata, enum/property metadata, and default value emission). When malicious values from an OpenAPI description are emitted into generated source without context-appropriate escaping, an attacker can break out of string literals and inject additional code into generated clients. This issue is only practically exploitable when the OpenAPI description used for generation is from an untrusted source, or a normally trusted OpenAPI description has been compromised/tampered with. Only generating from trusted, integrity-protected API descriptions significantly reduces the risk. To remediate the issue, upgrade Kiota to 1.31.1 or later and regenerate/refresh existing generated clients as a precaution. Refreshing generated clients ensures previously generated vulnerable code is replaced with hardened output.		
CVE-2026-5935	ibm - Total Storage Service Console (TSSC) / TS4500 IMC	IBM Total Storage Service Console (TSSC) / TS4500 IMC 9.2, 9.3, 9.4, 9.5, 9.6 TSSC/IMC could allow an unauthenticated user to execute arbitrary commands with normal user privileges on the system due to improper validation of user supplied input.	2026-04-23	7.3
CVE-2026-31569	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: LoongArch: KVM: Handle the case that EIOINTC's coremap is empty EIOINTC's coremap in eiointc_update_sw_coremap() can be empty, currently we get a cpuid with -1 in this case, but we actually need 0 because it's similar as the case that cpuid >= 4. This fix an out-of-bounds access to kvm_arch::phyid_map::phys_map[].	2026-04-24	7.3
CVE-2026-23774	dell - multiple products	Dell PowerProtect Data Domain with Data Domain Operating System (DD OS) of Feature Release versions 7.7.1.0 through 8.5, LTS2025 release version 8.3.1.0 through 8.3.1.10, LTS2024 release versions 7.13.1.0 through 7.13.1.40, contain an OS command injection vulnerability. A high privileged attacker with remote access could potentially exploit this vulnerability, leading to arbitrary command execution.	2026-04-20	7.2
CVE-2026-24504	dell - multiple products	Dell PowerProtect Data Domain, versions 7.7.1.0 through 8.6, LTS2025 release version 8.3.1.0 through 8.3.1.20, LTS2024 release versions 7.13.1.0 through 7.13.1.60 contain an improper input validation vulnerability. A high privileged attacker with remote access could potentially exploit this vulnerability, leading to arbitrary command execution with root privileges.	2026-04-20	7.2
CVE-2026-24505	dell - multiple products	Dell PowerProtect Data Domain, versions 8.5 through 8.6 contain an improper input validation vulnerability. A high privileged attacker with remote access could potentially exploit this vulnerability, leading to arbitrary command execution with root privileges.	2026-04-20	7.2
CVE-2026-24506	dell - multiple products	Dell PowerProtect Data Domain, versions 7.7.1.0 through 8.6, LTS2025 release version 8.3.1.0 through 8.3.1.20, LTS2024 release versions 7.13.1.0 through 7.13.1.60 contain an OS command injection vulnerability. A high privileged attacker with remote access could potentially exploit this vulnerability, leading to arbitrary command execution as root.	2026-04-20	7.2
CVE-2026-26943	dell - multiple products	Dell PowerProtect Data Domain, versions 7.7.1.0 through 8.6, LTS2025 release version 8.3.1.0 through 8.3.1.20, LTS2024 release versions 7.13.1.0 through 7.13.1.60 contain an OS command injection vulnerability. A high privileged attacker with remote access could potentially exploit this vulnerability, leading to arbitrary command execution with root privileges.	2026-04-20	7.2
CVE-2026-34292	oracle - multiple products	Vulnerability in the Oracle WebLogic Server product of Oracle Fusion Middleware (component: Core). Supported versions that are affected are 12.2.1.4.0 and 14.1.1.0.0. Easily exploitable vulnerability allows high privileged attacker with network access via HTTP to compromise Oracle WebLogic Server. Successful attacks of this vulnerability can result in takeover of Oracle WebLogic Server. CVSS 3.1 Base Score 7.2 (Confidentiality, Integrity and Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:H/I:H/A:H).	2026-04-21	7.2
CVE-2026-6855	red hat - multiple products	A flaw was found in InstructLab. A local attacker could exploit a path traversal vulnerability in the chat session handler by manipulating the `logs_dir` parameter. This allows the attacker to create new directories and write files to arbitrary locations on the system, potentially leading to unauthorized data modification or disclosure.	2026-04-22	7.1
CVE-2026-31470	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: virt: tdx-guest: Fix handling of host controlled 'quote' buffer length Validate host controlled value `quote_buf->out_len` that determines how many bytes of the quote are copied out to guest userspace. In TDX environments with remote attestation, quotes are not considered private, and can be forwarded to an attestation server. Catch scenarios where the host specifies a response length larger than the guest's allocation, or otherwise races modifying the response while the guest consumes it. This prevents contents beyond the pages allocated for `quote_buf` (up to TSM_REPORT_OUTBLOB_MAX) from being read out to guest userspace, and possibly forwarded in attestation requests. Recall that some deployments want per-container configs-tsm-report interfaces, so the leak may cross container protection boundaries, not just local root.	2026-04-22	7.1
CVE-2026-31484	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: io_uring/fdinfo: fix OOB read in SQE_MIXED wrap check __io_uring_show_fdinfo() iterates over pending SQEs and, for 128-byte SQEs on an IORING_SETUP_SQE_MIXED ring, needs to detect when the second	2026-04-22	7.1

		half of the SQE would be past the end of the sq_sqes array. The current check tests (++sq_head & sq_mask) == 0, but sq_head is only incremented when a 128-byte SQE is encountered, not on every iteration. The actual array index is sq_idx = (i + sq_head) & sq_mask, which can be sq_mask (the last slot) while the wrap check passes. Fix by checking sq_idx directly. Keep the sq_head increment so the loop still skips the second half of the 128-byte SQE on the next iteration.		
CVE-2026-31486	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: hwmon: (pmbus/core) Protect regulator operations with mutex The regulator operations pmbus_regulator_get_voltage(), pmbus_regulator_set_voltage(), and pmbus_regulator_list_voltage() access PMBus registers and shared data but were not protected by the update_lock mutex. This could lead to race conditions. However, adding mutex protection directly to these functions causes a deadlock because pmbus_regulator_notify() (which calls regulator_notifier_call_chain()) is often called with the mutex already held (e.g., from pmbus_fault_handler()). If a regulator callback then calls one of the now-protected voltage functions, it will attempt to acquire the same mutex. Rework pmbus_regulator_notify() to utilize a worker function to send notifications outside of the mutex protection. Events are stored as atomics in a per-page bitmask and processed by the worker. Initialize the worker and its associated data during regulator registration, and ensure it is cancelled on device removal using devm_add_action_or_reset(). While at it, remove the unnecessary include of linux/of.h.	2026-04-22	7.1
CVE-2026-31568	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: s390/mm: Add missing secure storage access fixups for donated memory There are special cases where secure storage access exceptions happen in a kernel context for pages that don't have the PG_arch_1 bit set. That bit is set for non-exported guest secure storage (memory) but is absent on storage donated to the Ultravisor since the kernel isn't allowed to export donated pages. Prior to this patch we would try to export the page by calling arch_make_folio_accessible() which would instantly return since the arch bit is absent signifying that the page was already exported and no further action is necessary. This leads to secure storage access exception loops which can never be resolved. With this patch we unconditionally try to export and if that fails we fixup.	2026-04-24	7.1
CVE-2026-31614	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: smb: client: fix off-by-8 bounds check in check_wsl_eas() The bounds check uses (u8 *)ea + nlen + 1 + vlen as the end of the EA name and value, but ea_data sits at offset sizeof(struct smb2_file_full_ea_info) = 8 from ea, not at offset 0. The strncmp() later reads ea->ea_data[0..nlen-1] and the value bytes follow at ea_data[nlen+1..nlen+vlen], so the actual end is ea->ea_data + nlen + 1 + vlen. Isn't pointer math fun? The earlier check (u8 *)ea > end - sizeof(*ea) only guarantees the 8-byte header is in bounds, but since the last EA is placed within 8 bytes of the end of the response, the name and value bytes are read past the end of iov. Fix this mess all up by using ea->ea_data as the base for the bounds check. An "untrusted" server can use this to leak up to 8 bytes of kernel heap into the EA name comparison and influence which WSL xattr the data is interpreted as.	2026-04-24	7.1
CVE-2026-31626	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: staging: rtl8723bs: initialize le_tmp64 in rtw_BIP_verify() Initialize le_tmp64 to zero in rtw_BIP_verify() to prevent using uninitialized data.	2026-04-24	7.1

		Smatch warns that only 6 bytes are copied to this 8-byte (u64) variable, leaving the last two bytes uninitialized: drivers/staging/rtl8723bs/core/rtw_security.c:1308 rtw_BIP_verify() warn: not copying enough bytes for '&le_tmp64' (8 vs 6 bytes) Initializing the variable at the start of the function fixes this warning and ensures predictable behavior.		
CVE-2026-31674	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: netfilter: ip6t_rt: reject oversized addrnr in rt_mt6_check() Reject rt match rules whose addrnr exceeds IP6T_RT_HOPS. rt_mt6() expects addrnr to stay within the bounds of rtinfo->addrs[]. Validate addrnr during rule installation so malformed rules are rejected before the match logic can use an out-of-range value.	2026-04-25	7.1
CVE-2026-31679	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: openvswitch: validate MPLS set/set_masked payload length validate_set() accepted OVS_KEY_ATTR_MPLS as variable-sized payload for SET/SET_MASKED actions. In action handling, OVS expects fixed-size MPLS key data (struct ovs_key_mpls). Use the already normalized key_len (masked case included) and reject non-matching MPLS action key sizes. Reject invalid MPLS action payload lengths early.	2026-04-25	7.1
CVE-2026-34314	oracle - multiple products	Vulnerability in the Oracle Financial Services Analytical Applications Infrastructure product of Oracle Financial Services Applications (component: Platform). Supported versions that are affected are 8.0.7.9, 8.0.8.7 and 8.1.2.5. Difficult to exploit vulnerability allows low privileged attacker with network access via HTTP to compromise Oracle Financial Services Analytical Applications Infrastructure. Successful attacks of this vulnerability can result in unauthorized creation, deletion or modification access to critical data or all Oracle Financial Services Analytical Applications Infrastructure accessible data as well as unauthorized access to critical data or complete access to all Oracle Financial Services Analytical Applications Infrastructure accessible data. CVSS 3.1 Base Score 6.8 (Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:H/PR:L/UI:N/S:U/C:H/I:H/A:N).	2026-04-21	6.8
CVE-2026-34325	oracle - multiple products	Vulnerability in the Oracle Financial Services Analytical Applications Infrastructure product of Oracle Financial Services Applications (component: User Interface). Supported versions that are affected are 8.0.7.9, 8.0.8.7 and 8.1.2.5. Easily exploitable vulnerability allows low privileged attacker with logon to the infrastructure where Oracle Financial Services Analytical Applications Infrastructure executes to compromise Oracle Financial Services Analytical Applications Infrastructure. Successful attacks require human interaction from a person other than the attacker. Successful attacks of this vulnerability can result in unauthorized access to critical data or complete access to all Oracle Financial Services Analytical Applications Infrastructure accessible data as well as unauthorized update, insert or delete access to some of Oracle Financial Services Analytical Applications Infrastructure accessible data and unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of Oracle Financial Services Analytical Applications Infrastructure. CVSS 3.1 Base Score 6.8 (Confidentiality, Integrity and Availability impacts). CVSS Vector: (CVSS:3.1/AV:L/AC:L/PR:L/UI:R/S:U/C:H/I:L/A:H).	2026-04-21	6.8
CVE-2026-22747	vmware - spring_security	Vulnerability in Spring Spring Security. SubjectX500PrincipalExtractor does not correctly handle certain malformed X.509 certificate CN values, which can lead to reading the wrong value for the username. In a carefully crafted certificate, this can lead to an attacker impersonating another user. This issue affects Spring Security: from 7.0.0 through 7.0.4.	2026-04-22	6.8
CVE-2026-22761	dell - multiple products	Dell PowerProtect Data Domain, versions 8.5 through 8.6 contain a command injection vulnerability. A high privileged attacker with remote access could potentially exploit this vulnerability, leading to arbitrary command execution with root privileges.	2026-04-20	6.7
CVE-2026-26942	dell - multiple products	Dell PowerProtect Data Domain, versions 8.5 through 8.6 contain(s) an Improper Neutralization of Special Elements used in an OS Command ('OS command injection vulnerability. A high privileged attacker with remote access could potentially exploit this vulnerability, leading to arbitrary command execution with root privileges.	2026-04-20	6.7
CVE-2026-26951	dell - multiple products	Dell PowerProtect Data Domain, versions 7.7.1.0 through 8.6, LTS2025 release version 8.3.1.0 through 8.3.1.20, LTS2024 release versions 7.13.1.0 through 7.13.1.60 contain a stack-based buffer overflow vulnerability. A high privileged attacker with local access could potentially exploit this vulnerability, leading to arbitrary command execution with root privileges.	2026-04-20	6.7
CVE-2026-34277	oracle - multiple products	Vulnerability in the PeopleSoft Enterprise PeopleTools product of Oracle PeopleSoft (component: Fluid Core). Supported versions that are affected are 8.61-8.62. Easily exploitable vulnerability allows high privileged attacker with network access via HTTP to compromise PeopleSoft Enterprise PeopleTools. While the vulnerability is in PeopleSoft Enterprise PeopleTools, attacks may significantly impact additional products (scope change). Successful attacks of this vulnerability can result in unauthorized update, insert or delete access to some of PeopleSoft Enterprise PeopleTools accessible data as well as unauthorized read access to a subset of PeopleSoft Enterprise PeopleTools accessible data and unauthorized ability to cause a partial denial of service (partial DOS) of PeopleSoft Enterprise PeopleTools. CVSS 3.1 Base Score 6.6 (Confidentiality, Integrity and Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:C/C:L/I:L/A:L).	2026-04-21	6.6

CVE-2026-40449	samsung - one	Integer overflow in buffer size calculation could result in out of bounds memory access when handling large tensors in Samsung Open Source ONE. Affected version is prior to commit 1.30.0.	2026-04-22	6.6
CVE-2026-40450	samsung - one	Integer overflow in output tensor copy size calculation in Samsung Open Source ONE could cause incorrect copy length and memory corruption for oversized tensors. Affected version is prior to commit 1.30.0.	2026-04-22	6.6
CVE-2026-41664	samsung - one	Integer overflow in memory copy size calculation in Samsung Open Source ONE could lead to invalid memory operations with large tensor shapes. Affected version is prior to commit 1.30.0.	2026-04-22	6.6
CVE-2026-41666	samsung - one	Integer overflow in tensor copy size calculation in Samsung Open Source ONE could lead to out of bounds access during loop state propagation. Affected version is prior to commit 1.30.0.	2026-04-22	6.6
CVE-2026-41667	samsung - one	Integer overflow in constant tensor data size calculation in Samsung Open Source ONE could cause incorrect buffer sizing for large constant nodes. Affected version is prior to commit 1.30.0.	2026-04-22	6.6
CVE-2026-6839	samsung - one	Improper validation of STRING tensor offsets could allows malformed string metadata to trigger out of bounds access during constant tensor import in Samsung Open Source ONE Affected version is prior to commit 1.30.0.	2026-04-22	6.6
CVE-2026-6755	mozilla - multiple products	Mitigation bypass in the DOM: postMessage component. This vulnerability was fixed in Firefox 150 and Thunderbird 150.	2026-04-21	6.5
CVE-2026-6763	mozilla - multiple products	Mitigation bypass in the File Handling component. This vulnerability was fixed in Firefox 150, Firefox ESR 140.10, Thunderbird 150, and Thunderbird 140.10.	2026-04-21	6.5
CVE-2026-6764	mozilla - multiple products	Incorrect boundary conditions in the DOM: Device Interfaces component. This vulnerability was fixed in Firefox 150, Firefox ESR 140.10, Thunderbird 150, and Thunderbird 140.10.	2026-04-21	6.5
CVE-2026-6770	mozilla - multiple products	Other issue in the Storage: IndexedDB component. This vulnerability was fixed in Firefox 150, Firefox ESR 140.10, Thunderbird 150, and Thunderbird 140.10.	2026-04-21	6.5
CVE-2026-22009	oracle - multiple products	Vulnerability in the MySQL Server product of Oracle MySQL (component: Server: Optimizer). Supported versions that are affected are 8.0.0-8.0.45, 8.4.0-8.4.8 and 9.0.0-9.6.0. Easily exploitable vulnerability allows low privileged attacker with network access via multiple protocols to compromise MySQL Server. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of MySQL Server. CVSS 3.1 Base Score 6.5 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:N/I:N/A:H).	2026-04-21	6.5
CVE-2026-22017	oracle - multiple products	Vulnerability in the MySQL Server product of Oracle MySQL (component: Server: Optimizer). Supported versions that are affected are 8.0.0-8.0.45, 8.4.0-8.4.8 and 9.0.0-9.6.0. Easily exploitable vulnerability allows low privileged attacker with network access via multiple protocols to compromise MySQL Server. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of MySQL Server. CVSS 3.1 Base Score 6.5 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:N/I:N/A:H).	2026-04-21	6.5
CVE-2026-34266	oracle - peoplesoft_enterprise_human_capital_management_absence_management	Vulnerability in the PeopleSoft Enterprise HCM Absence Management product of Oracle PeopleSoft (component: Absence Management). The supported version that is affected is 9.2. Easily exploitable vulnerability allows high privileged attacker with network access via HTTP to compromise PeopleSoft Enterprise HCM Absence Management. Successful attacks of this vulnerability can result in unauthorized creation, deletion or modification access to critical data or all PeopleSoft Enterprise HCM Absence Management accessible data as well as unauthorized access to critical data or complete access to all PeopleSoft Enterprise HCM Absence Management accessible data. CVSS 3.1 Base Score 6.5 (Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:H/I:H/A:N).	2026-04-21	6.5
CVE-2026-34270	oracle - multiple products	Vulnerability in the MySQL Server product of Oracle MySQL (component: Server: Group Replication Plugin). Supported versions that are affected are 8.0.0-8.0.45, 8.4.0-8.4.8 and 9.0.0-9.6.0. Easily exploitable vulnerability allows low privileged attacker with network access via multiple protocols to compromise MySQL Server. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of MySQL Server. CVSS 3.1 Base Score 6.5 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:N/I:N/A:H).	2026-04-21	6.5
CVE-2026-34271	oracle - multiple products	Vulnerability in the MySQL Server product of Oracle MySQL (component: Server: Group Replication Plugin). Supported versions that are affected are 8.0.0-8.0.45, 8.4.0-8.4.8 and 9.0.0-9.6.0. Easily exploitable vulnerability allows low privileged attacker with network access via multiple protocols to compromise MySQL Server. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of MySQL Server. CVSS 3.1 Base Score 6.5 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:N/I:N/A:H).	2026-04-21	6.5
CVE-2026-34272	oracle - mysql_server	Vulnerability in the MySQL Server product of Oracle MySQL (component: Server: Optimizer). Supported versions that are affected are 9.0.0-9.6.0. Easily exploitable vulnerability allows low privileged attacker with network access via multiple protocols to compromise MySQL Server. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of MySQL Server. CVSS 3.1 Base Score 6.5 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:N/I:N/A:H).	2026-04-21	6.5
CVE-2026-34276	oracle - multiple products	Vulnerability in the MySQL Server product of Oracle MySQL (component: Server: Group Replication Plugin). Supported versions that are affected are 8.0.0-8.0.45, 8.4.0-8.4.8 and 9.0.0-9.6.0. Easily exploitable vulnerability allows low privileged attacker with network access via multiple protocols to compromise MySQL Server. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of MySQL Server. CVSS 3.1 Base Score 6.5 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:N/I:N/A:H).	2026-04-21	6.5
CVE-2026-34280	oracle - peoplesoft_enterprise_hcm_human_resources	Vulnerability in the PeopleSoft Enterprise HCM Human Resources product of Oracle PeopleSoft (component: Job Profile Manager). The supported version that is affected is 9.2. Easily exploitable vulnerability allows high privileged attacker with network access via HTTP to compromise PeopleSoft Enterprise HCM Human Resources. Successful attacks of this vulnerability can result in unauthorized creation, deletion or modification access to critical data or all PeopleSoft Enterprise HCM Human Resources accessible data as well as unauthorized access to critical data or complete access to all PeopleSoft Enterprise HCM Human Resources accessible data. CVSS 3.1 Base Score 6.5	2026-04-21	6.5

		(Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:H/I:H/A:N).		
CVE-2026-34281	oracle - solaris	Vulnerability in the Oracle Solaris product of Oracle Systems (component: Kernel). The supported version that is affected is 11.4. Easily exploitable vulnerability allows low privileged attacker with logon to the infrastructure where Oracle Solaris executes to compromise Oracle Solaris. While the vulnerability is in Oracle Solaris, attacks may significantly impact additional products (scope change). Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of Oracle Solaris. CVSS 3.1 Base Score 6.5 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:C/C:N/I:N/A:H).	2026-04-21	6.5
CVE-2026-34295	oracle - peoplesoft_enterprise_scm_purchasing	Vulnerability in the PeopleSoft Enterprise SCM Purchasing product of Oracle PeopleSoft (component: Purchasing). The supported version that is affected is 9.2. Easily exploitable vulnerability allows low privileged attacker with network access via HTTP to compromise PeopleSoft Enterprise SCM Purchasing. Successful attacks of this vulnerability can result in unauthorized access to critical data or complete access to all PeopleSoft Enterprise SCM Purchasing accessible data. CVSS 3.1 Base Score 6.5 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N).	2026-04-21	6.5
CVE-2026-34299	oracle - peoplesoft_enterprise_fin_maintenance_management	Vulnerability in the PeopleSoft Enterprise FIN Maintenance Management product of Oracle PeopleSoft (component: Work Order Management). The supported version that is affected is 9.2. Easily exploitable vulnerability allows low privileged attacker with network access via HTTP to compromise PeopleSoft Enterprise FIN Maintenance Management. Successful attacks of this vulnerability can result in unauthorized access to critical data or complete access to all PeopleSoft Enterprise FIN Maintenance Management accessible data. CVSS 3.1 Base Score 6.5 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N).	2026-04-21	6.5
CVE-2026-34300	oracle - peoplesoft_enterprise_fin_contracts	Vulnerability in the PeopleSoft Enterprise FIN Contracts product of Oracle PeopleSoft (component: Contracts). The supported version that is affected is 9.2. Easily exploitable vulnerability allows low privileged attacker with network access via HTTP to compromise PeopleSoft Enterprise FIN Contracts. Successful attacks of this vulnerability can result in unauthorized access to critical data or complete access to all PeopleSoft Enterprise FIN Contracts accessible data. CVSS 3.1 Base Score 6.5 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N).	2026-04-21	6.5
CVE-2026-34301	oracle - peoplesoft_enterprise_fin_maintenance_management	Vulnerability in the PeopleSoft Enterprise FIN Maintenance Management product of Oracle PeopleSoft (component: Work Order Management). The supported version that is affected is 9.2. Easily exploitable vulnerability allows low privileged attacker with network access via HTTP to compromise PeopleSoft Enterprise FIN Maintenance Management. Successful attacks of this vulnerability can result in unauthorized access to critical data or complete access to all PeopleSoft Enterprise FIN Maintenance Management accessible data. CVSS 3.1 Base Score 6.5 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N).	2026-04-21	6.5
CVE-2026-34303	oracle - multiple products	Vulnerability in the MySQL Server product of Oracle MySQL (component: Server: Optimizer). Supported versions that are affected are 8.0.0-8.0.45, 8.4.0-8.4.8 and 9.0.0-9.6.0. Easily exploitable vulnerability allows low privileged attacker with network access via multiple protocols to compromise MySQL Server. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of MySQL Server. CVSS 3.1 Base Score 6.5 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:N/I:N/A:H).	2026-04-21	6.5
CVE-2026-34306	oracle - peoplesoft_enterprise_fin_project_costing	Vulnerability in the PeopleSoft Enterprise FIN Project Costing product of Oracle PeopleSoft (component: Projects). The supported version that is affected is 9.2. Easily exploitable vulnerability allows low privileged attacker with network access via HTTP to compromise PeopleSoft Enterprise FIN Project Costing. Successful attacks of this vulnerability can result in unauthorized access to critical data or complete access to all PeopleSoft Enterprise FIN Project Costing accessible data. CVSS 3.1 Base Score 6.5 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N).	2026-04-21	6.5
CVE-2026-34308	oracle - multiple products	Vulnerability in the MySQL Server product of Oracle MySQL (component: Server: JSON). Supported versions that are affected are 8.0.0-8.0.45, 8.4.0-8.4.8 and 9.0.0-9.6.0. Easily exploitable vulnerability allows low privileged attacker with network access via multiple protocols to compromise MySQL Server. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of MySQL Server. CVSS 3.1 Base Score 6.5 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:N/I:N/A:H).	2026-04-21	6.5
CVE-2026-34313	oracle - multiple products	Vulnerability in the Oracle Financial Services Analytical Applications Infrastructure product of Oracle Financial Services Applications (component: Platform). Supported versions that are affected are 8.0.7.9, 8.0.8.7 and 8.1.2.5. Easily exploitable vulnerability allows low privileged attacker with network access via HTTP to compromise Oracle Financial Services Analytical Applications Infrastructure. Successful attacks of this vulnerability can result in unauthorized access to critical data or complete access to all Oracle Financial Services Analytical Applications Infrastructure accessible data. CVSS 3.1 Base Score 6.5 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N).	2026-04-21	6.5
CVE-2026-34315	oracle - multiple products	Vulnerability in the Oracle WebLogic Server product of Oracle Fusion Middleware (component: Web Services). Supported versions that are affected are 12.2.1.4.0, 14.1.1.0.0, 14.1.2.0.0 and 15.1.1.0.0. Easily exploitable vulnerability allows unauthenticated attacker with network access via HTTP to compromise Oracle WebLogic Server. Successful attacks require human interaction from a person other than the attacker. Successful attacks of this vulnerability can result in unauthorized creation, deletion or modification access to critical data or all Oracle WebLogic Server accessible data. CVSS 3.1 Base Score 6.5 (Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:U/C:N/I:H/A:N).	2026-04-21	6.5
CVE-2026-34324	oracle - multiple products	Vulnerability in the Oracle Life Sciences InForm product of Oracle Life Science Applications (component: App Server). Supported versions that are affected are 7.0.1.0 and 7.0.1.1. Easily exploitable vulnerability allows unauthenticated attacker with network access via HTTP to compromise Oracle Life Sciences InForm. Successful attacks of this vulnerability can result in unauthorized update, insert or delete access to some of Oracle Life Sciences InForm accessible data as well as unauthorized read access to a subset of Oracle Life Sciences InForm accessible data. CVSS 3.1 Base Score 6.5 (Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:L/A:N).	2026-04-21	6.5

CVE-2026-1352	ibm - multiple products	IBM Db2 11.5.0 through 11.5.9, and 12.1.0 through 12.1.4 for Linux, UNIX and Windows (includes Db2 Connect Server) could allow an authenticated user to cause a denial of service due to improper neutralization of special elements in data query logic.	2026-04-23	6.5
CVE-2026-5926	ibm - multiple products	IBM Verify Identity Access Container 11.0 through 11.0.2 and IBM Security Verify Access Container 10.0 through 10.0.9.1 and IBM Verify Identity Access 11.0 through 11.0.2 and IBM Security Verify Access 10.0 through 10.0.9.1 uses weaker than expected cryptographic algorithms that could allow an attacker to decrypt highly sensitive information.	2026-04-23	6.5
CVE-2026-6732	red hat - multiple products	A flaw was found in libxml2. This vulnerability occurs when the library processes a specially crafted XML Schema Definition (XSD) validated document that includes an internal entity reference. An attacker could exploit this by providing a malicious document, leading to a type confusion error that causes the application to crash. This results in a denial of service (DoS), making the affected system or application unavailable.	2026-04-23	6.5
CVE-2026-41043	apache - multiple products	Improper Neutralization of Script-Related HTML Tags in a Web Page (Basic XSS) vulnerability in Apache ActiveMQ, Apache ActiveMQ Web. An authenticated attacker can show malicious content when browsing queues in the web console by overriding the content type to be HTML (instead of XML) and by injecting HTML into a JMS selector field. This issue affects Apache ActiveMQ: before 5.19.6, from 6.0.0 before 6.2.5; Apache ActiveMQ Web: before 5.19.6, from 6.0.0 before 6.2.5. Users are recommended to upgrade to version 6.2.5 or 5.19.6, which fixes the issue.	2026-04-24	6.5
CVE-2026-5265	red hat - multiple products	When generating an ICMP Destination Unreachable or Packet Too Big response, the handler copies a portion of the original packet into the ICMP error body using the IP header's self-declared total length (ip_tot_len for IPv4, ip6_plen for IPv6) without validating it against the actual packet buffer size. A VM can send a short packet with an inflated IP length field that triggers an ICMP error (e.g., by hitting a reject ACL), causing ovn-controller to read heap memory beyond the valid packet data and include it in the ICMP response sent back to the VM.	2026-04-24	6.5
CVE-2026-35252	oracle - multiple products	Vulnerability in the Oracle Security Service product of Oracle Fusion Middleware (component: C Oracle SSL API). Supported versions that are affected are 12.2.1.4.0 and 12.1.3.0.0. Difficult to exploit vulnerability allows low privileged attacker with network access via HTTPS to compromise Oracle Security Service. Successful attacks require human interaction from a person other than the attacker. Successful attacks of this vulnerability can result in unauthorized creation, deletion or modification access to critical data or all Oracle Security Service accessible data as well as unauthorized access to critical data or complete access to all Oracle Security Service accessible data. CVSS 3.1 Base Score 6.4 (Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:H/PR:L/UI:R/S:U/C:H/I:H/A:N).	2026-04-21	6.4
CVE-2026-35154	dell - multiple products	Dell PowerProtect Data Domain appliances, versions 7.7.1.0 through 8.7.0.0, LTS2025 release versions 8.3.1.0 through 8.3.1.20, LTS2024 release versions 7.13.1.0 through 7.13.1.60 contain an improper privilege management vulnerability in IDRAC. A high privileged attacker with local access could potentially exploit this vulnerability, leading to elevation of privileges to access unauthorized delete operation in IDRAC.	2026-04-20	6.3
CVE-2026-6757	mozilla - multiple products	Invalid pointer in the JavaScript: WebAssembly component. This vulnerability was fixed in Firefox 150, Firefox ESR 140.10, Thunderbird 150, and Thunderbird 140.10.	2026-04-21	6.3
CVE-2026-6762	mozilla - multiple products	Spoofing issue in the DOM: Core & HTML component. This vulnerability was fixed in Firefox 150, Firefox ESR 115.35, Firefox ESR 140.10, Thunderbird 150, and Thunderbird 140.10.	2026-04-21	6.3
CVE-2026-34323	oracle - multiple products	Vulnerability in the Oracle Life Sciences InForm product of Oracle Life Science Applications (component: IDM Authentication). Supported versions that are affected are 7.0.1.0 and 7.0.1.1. Easily exploitable vulnerability allows unauthenticated attacker with network access via HTTP to compromise Oracle Life Sciences InForm. Successful attacks require human interaction from a person other than the attacker. Successful attacks of this vulnerability can result in unauthorized update, insert or delete access to some of Oracle Life Sciences InForm accessible data as well as unauthorized read access to a subset of Oracle Life Sciences InForm accessible data and unauthorized ability to cause a partial denial of service (partial DOS) of Oracle Life Sciences InForm. CVSS 3.1 Base Score 6.3 (Confidentiality, Integrity and Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:U/C:L/I:L/A:L).	2026-04-21	6.3
CVE-2025-62233	apache - dolphinscheduler	Deserialization of Untrusted Data vulnerability in Apache DolphinScheduler RPC module. This issue affects Apache DolphinScheduler: Version >= 3.2.0 and < 3.3.1. Attackers who can access the Master or Worker nodes can compromise the system by creating a StandardRpcRequest, injecting a malicious class type into it, and sending RPC requests to the DolphinScheduler Master/Worker nodes. Users are recommended to upgrade to version [3.3.1], which fixes the issue.	2026-04-24	6.3
CVE-2026-28950	apple - multiple products	A logging issue was addressed with improved data redaction. This issue is fixed in iOS 18.7.8 and iPadOS 18.7.8, iOS 26.4.2 and iPadOS 26.4.2. Notifications marked for deletion could be unexpectedly retained on the device.	2026-04-22	6.2
CVE-2026-34269	oracle - multiple products	Vulnerability in the PeopleSoft Enterprise PeopleTools product of Oracle PeopleSoft (component: Portal). Supported versions that are affected are 8.61-8.62. Easily exploitable vulnerability allows unauthenticated attacker with network access via HTTP to compromise PeopleSoft Enterprise PeopleTools. Successful attacks require human interaction from a person other than the attacker and while the vulnerability is in PeopleSoft Enterprise PeopleTools, attacks may significantly impact additional products (scope change). Successful attacks of this vulnerability can result in unauthorized update, insert or delete access to some of PeopleSoft Enterprise PeopleTools accessible data as well as unauthorized read access to a subset of PeopleSoft Enterprise	2026-04-21	6.1

		PeopleTools accessible data. CVSS 3.1 Base Score 6.1 (Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:C/C:L/I:L/A:N).		
CVE-2026-34274	oracle - configurator	Vulnerability in the Oracle Configurator product of Oracle E-Business Suite (component: User Interface). Supported versions that are affected are 12.2.3-12.2.15. Easily exploitable vulnerability allows unauthenticated attacker with network access via HTTP to compromise Oracle Configurator. Successful attacks require human interaction from a person other than the attacker and while the vulnerability is in Oracle Configurator, attacks may significantly impact additional products (scope change). Successful attacks of this vulnerability can result in unauthorized update, insert or delete access to some of Oracle Configurator accessible data as well as unauthorized read access to a subset of Oracle Configurator accessible data. CVSS 3.1 Base Score 6.1 (Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:C/C:L/I:L/A:N).	2026-04-21	6.1
CVE-2026-34283	oracle - multiple products	Vulnerability in the Oracle Identity Manager product of Oracle Fusion Middleware (component: Identity Console). Supported versions that are affected are 12.2.1.4.0 and 14.1.2.0.0. Easily exploitable vulnerability allows unauthenticated attacker with network access via HTTP to compromise Oracle Identity Manager. Successful attacks require human interaction from a person other than the attacker and while the vulnerability is in Oracle Identity Manager, attacks may significantly impact additional products (scope change). Successful attacks of this vulnerability can result in unauthorized update, insert or delete access to some of Oracle Identity Manager accessible data as well as unauthorized read access to a subset of Oracle Identity Manager accessible data. CVSS 3.1 Base Score 6.1 (Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:C/C:L/I:L/A:N).	2026-04-21	6.1
CVE-2026-34284	oracle - multiple products	Vulnerability in the Oracle Business Process Management Suite product of Oracle Fusion Middleware (component: Human workflow 11g+). Supported versions that are affected are 12.2.1.4.0 and 14.1.2.0.0. Easily exploitable vulnerability allows unauthenticated attacker with network access via HTTP to compromise Oracle Business Process Management Suite. Successful attacks require human interaction from a person other than the attacker and while the vulnerability is in Oracle Business Process Management Suite, attacks may significantly impact additional products (scope change). Successful attacks of this vulnerability can result in unauthorized update, insert or delete access to some of Oracle Business Process Management Suite accessible data as well as unauthorized read access to a subset of Oracle Business Process Management Suite accessible data. CVSS 3.1 Base Score 6.1 (Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:C/C:L/I:L/A:N).	2026-04-21	6.1
CVE-2026-41665	samsung - one	Integer overflow in scratch buffer initialization size calculation in Samsung Open Source ONE cause incorrect memory initialization for large intermediate tensors. Affected version is prior to commit 1.30.0.	2026-04-22	6.1
CVE-2026-6861	red hat - multiple products	A flaw was found in GNU Emacs. This vulnerability, a memory corruption issue, occurs when Emacs processes specially crafted SVG (Scalable Vector Graphics) CSS (Cascading Style Sheets) data. A local user could exploit this by convincing a victim to open a malicious SVG file, which may lead to a denial of service (DoS) or potentially information disclosure.	2026-04-22	6.1
CVE-2026-22003	oracle - multiple products	Vulnerability in the Oracle Java SE, Oracle GraalVM Enterprise Edition product of Oracle Java SE (component: Hotspot). Supported versions that are affected are Oracle Java SE: 8u481 and 8u481-b50; Oracle GraalVM Enterprise Edition: 21.3.17. Difficult to exploit vulnerability allows low privileged attacker with logon to the infrastructure where Oracle Java SE, Oracle GraalVM Enterprise Edition executes to compromise Oracle Java SE, Oracle GraalVM Enterprise Edition. Successful attacks require human interaction from a person other than the attacker. Successful attacks of this vulnerability can result in unauthorized creation, deletion or modification access to critical data or all Oracle Java SE, Oracle GraalVM Enterprise Edition accessible data and unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of Oracle Java SE, Oracle GraalVM Enterprise Edition. Note: This vulnerability applies to Java deployments, typically in clients running sandboxed Java Web Start applications or sandboxed Java applets, that load and run untrusted code (e.g., code that comes from the internet) and rely on the Java sandbox for security. This vulnerability does not apply to Java deployments, typically in servers, that load and run only trusted code (e.g., code installed by an administrator). CVSS 3.1 Base Score 6.0 (Integrity and Availability impacts). CVSS Vector: (CVSS:3.1/AV:L/AC:H/PR:L/UI:R/S:U/C:N/I:H/A:H).	2026-04-21	6
CVE-2026-35247	oracle - vm_virtualbox	Vulnerability in the Oracle VM VirtualBox product of Oracle Virtualization (component: Core). The supported version that is affected is 7.2.6. Easily exploitable vulnerability allows high privileged attacker with logon to the infrastructure where Oracle VM VirtualBox executes to compromise Oracle VM VirtualBox. While the vulnerability is in Oracle VM VirtualBox, attacks may significantly impact additional products (scope change). Successful attacks of this vulnerability can result in unauthorized access to critical data or complete access to all Oracle VM VirtualBox accessible data. CVSS 3.1 Base Score 6.0 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:L/AC:L/PR:H/UI:N/S:C/C:H/I:N/A:N).	2026-04-21	6
CVE-2026-34288	oracle - identity_manager _connector	Vulnerability in the Oracle Identity Manager Connector product of Oracle Fusion Middleware (component: Core). The supported version that is affected is 12.2.1.4.0. Difficult to exploit vulnerability allows unauthenticated attacker with network access via HTTP to compromise Oracle Identity Manager Connector. Successful attacks of this vulnerability can result in unauthorized access to critical data or complete access to all Oracle Identity Manager Connector accessible data. CVSS 3.1 Base Score 5.9 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:N/A:N).	2026-04-21	5.9
CVE-2026-34289	oracle - identity_manager _connector	Vulnerability in the Oracle Identity Manager Connector product of Oracle Fusion Middleware (component: Core). The supported version that is affected is 12.2.1.4.0. Difficult to exploit vulnerability allows unauthenticated attacker with network access via HTTPS to compromise Oracle Identity Manager Connector. Successful attacks of this vulnerability can result in unauthorized access to critical data or complete access to all Oracle Identity Manager Connector accessible data. CVSS 3.1 Base Score 5.9 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:N/A:N).	2026-04-21	5.9
CVE-2026-34294	oracle - identity_manager _connector	Vulnerability in the Oracle Identity Manager Connector product of Oracle Fusion Middleware (component: Microsoft Active Directory). The supported version that is affected is 12.2.1.4.0. Difficult to exploit vulnerability allows low privileged attacker with network access via LDAP to		

		compromise Oracle Identity Manager Connector. Successful attacks of this vulnerability can result in unauthorized creation, deletion or modification access to critical data or all Oracle Identity Manager Connector accessible data as well as unauthorized read access to a subset of Oracle Identity Manager Connector accessible data. CVSS 3.1 Base Score 5.9 (Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:H/PR:L/UI:N/S:U/C:L/I:H/A:N).		
CVE-2026-35241	oracle - peoplesoft_enterprise_cs_student_records	Vulnerability in the PeopleSoft Enterprise CS Student Records product of Oracle PeopleSoft (component: Research Tracking). The supported version that is affected is 9.2. Easily exploitable vulnerability allows low privileged attacker with network access via HTTP to compromise PeopleSoft Enterprise CS Student Records. Successful attacks require human interaction from a person other than the attacker. Successful attacks of this vulnerability can result in unauthorized access to critical data or complete access to all PeopleSoft Enterprise CS Student Records accessible data. CVSS 3.1 Base Score 5.7 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:L/UI:R/S:U/C:H/I:N/A:N).	2026-04-21	5.7
CVE-2026-34302	oracle - workflow	Vulnerability in the Oracle Workflow product of Oracle E-Business Suite (component: Workflow Loader). Supported versions that are affected are 12.2.3-12.2.15. Easily exploitable vulnerability allows high privileged attacker with network access via HTTP to compromise Oracle Workflow. While the vulnerability is in Oracle Workflow, attacks may significantly impact additional products (scope change). Successful attacks of this vulnerability can result in unauthorized update, insert or delete access to some of Oracle Workflow accessible data and unauthorized ability to cause a partial denial of service (partial DOS) of Oracle Workflow. CVSS 3.1 Base Score 5.5 (Integrity and Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:C/C:N/I:L/A:L).	2026-04-21	5.5
CVE-2026-6843	red hat - multiple products	A flaw was found in nano. A local user could exploit a format string vulnerability in the `statusline()` function. By creating a directory with a name containing `printf` specifiers, the application attempts to display this name, leading to a segmentation fault (SEGV). This results in a Denial of Service (DoS) for the `nano` application.	2026-04-22	5.5
CVE-2026-6844	red hat - multiple products	A flaw was found in the `readelf` utility of the binutils package. A local attacker could exploit two Denial of Service (DoS) vulnerabilities by providing a specially crafted Executable and Linkable Format (ELF) file. One vulnerability, a resource exhaustion (CWE-400), can lead to an out-of-memory condition. The other, a null pointer dereference (CWE-476), can cause a segmentation fault. Both issues can result in the `readelf` utility becoming unresponsive or crashing, leading to a denial of service.	2026-04-22	5.5
CVE-2026-31472	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: xfrm: iptfs: validate inner IPv4 header length in IPTFS payload Add validation of the inner IPv4 packet tot_len and ihl fields parsed from decrypted IPTFS payloads in __input_process_payload(). A crafted ESP packet containing an inner IPv4 header with tot_len=0 causes an infinite loop: iplen=0 leads to capturelen=min(0, remaining)=0, so the data offset never advances and the while(data < tail) loop never terminates, spinning forever in softirq context. Reject inner IPv4 packets where tot_len < ihl*4 or ihl*4 < sizeof(struct iphdr), which catches both the tot_len=0 case and malformed ihl values. The normal IP stack performs this validation in ip_rcv_core(), but IPTFS extracts and processes inner packets before they reach that layer.	2026-04-22	5.5
CVE-2026-31480	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: tracing: Fix potential deadlock in cpu hotplug with osnoise The following sequence may leads deadlock in cpu hotplug: task1 task2 task3 ----- ----- ----- mutex_lock(&interface_lock) [CPU GOING OFFLINE] cpus_write_lock(); osnoise_cpu_die(); kthread_stop(task3); wait_for_completion(); osnoise_sleep(); mutex_lock(&interface_lock); cpus_read_lock(); [DEAD LOCK] Fix by swap the order of cpus_read_lock() and mutex_lock(&interface_lock).	2026-04-22	5.5
CVE-2026-31481	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: tracing: Drain deferred trigger frees if kthread creation fails Boot-time trigger registration can fail before the trigger-data cleanup kthread exists. Deferring those frees until late init is fine, but the post-boot fallback must still drain the deferred list if kthread creation never succeeds.	2026-04-22	5.5

		Otherwise, boot-deferred nodes can accumulate on trigger_data_free_list, later frees fall back to synchronously freeing only the current object, and the older queued entries are leaked forever. To trigger this, add the following to the kernel command line: trace_event=sched_switch trace_trigger=sched_switch.traceon,sched_switch.traceon The second traceon trigger will fail and be freed. This triggers a NULL pointer dereference and crashes the kernel. Keep the deferred boot-time behavior, but when kthread creation fails, drain the whole queued list synchronously. Do the same in the late-init drain path so queued entries are not stranded there either.		
CVE-2026-31482	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: s390/entry: Scrub r12 register on kernel entry Before commit f33f2d4c7c80 ("s390/bp: remove TIF_ISOLATE_BP"), all entry handlers loaded r12 with the current task pointer (lg %r12,__LC_CURRENT) for use by the BPENTER/BPEXIT macros. That commit removed TIF_ISOLATE_BP, dropping both the branch prediction macros and the r12 load, but did not add r12 to the register clearing sequence. Add the missing xgr %r12,%r12 to make the register scrub consistent across all entry points.	2026-04-22	5.5
CVE-2026-31483	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: s390/syscalls: Add spectre boundary for syscall dispatch table The s390 syscall number is directly controlled by userspace, but does not have an array_index_nospec() boundary to prevent access past the syscall function pointer tables.	2026-04-22	5.5
CVE-2026-31487	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: spi: use generic driver_override infrastructure When a driver is probed through __driver_attach(), the bus' match() callback is called without the device lock held, thus accessing the driver_override field without a lock, which can cause a UAF. Fix this by using the driver-core driver_override infrastructure taking care of proper locking internally. Note that calling match() from __driver_attach() without the device lock held is intentional. [1] Also note that we do not enable the driver_override feature of struct bus_type, as SPI - in contrast to most other buses - passes "" to sysfs_emit() when the driver_override pointer is NULL. Thus, printing "\n" instead of "(null)\n".	2026-04-22	5.5
CVE-2026-31491	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: RDMA/irdma: Harden depth calculation functions An issue was exposed where OS can pass in U32_MAX for SQ/RQ/SRQ size. This can cause integer overflow and truncation of SQ/RQ/SRQ depth returning a success when it should have failed. Harden the functions to do all depth calculations and boundary checking in u64 sizes.	2026-04-22	5.5
CVE-2026-31492	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: RDMA/irdma: Initialize free_qp completion before using it In irdma_create_qp, if ib_copy_to_udata fails, it will call irdma_destroy_qp to clean up which will attempt to wait on the free_qp completion, which is not initialized yet. Fix this by initializing the completion before the ib_copy_to_udata call.	2026-04-22	5.5
CVE-2026-31495	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: netfilter: ctnetlink: use netlink policy range checks Replace manual range and mask validations with netlink policy annotations in ctnetlink code paths, so that the netlink core rejects invalid values early and can generate extack errors.	2026-04-22	5.5

		- CTA_PROTOINFO_TCP_STATE: reject values > TCP_CONNTRACK_SYN_SENT2 at policy level, removing the manual >= TCP_CONNTRACK_MAX check. - CTA_PROTOINFO_TCP_WSCALE_ORIGINAL/REPLY: reject values > TCP_MAX_WSCALE (14). The normal TCP option parsing path already clamps to this value, but the ctnetlink path accepted 0-255, causing undefined behavior when used as a u32 shift count. - CTA_FILTER_ORIG_FLAGS/REPLY_FLAGS: use NLA_POLICY_MASK with CTA_FILTER_F_ALL, removing the manual mask checks. - CTA_EXPECT_FLAGS: use NLA_POLICY_MASK with NF_CT_EXPECT_MASK, adding a new mask define grouping all valid expect flags. Extracted from a broader nf-next patch by Florian Westphal, scoped to ctnetlink for the fixes tree.		
CVE-2026-31496	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: netfilter: nf_conntrack_expect: skip expectations in other netns via proc Skip expectations that do not reside in this netns. Similar to e77e6ff502ea ("netfilter: conntrack: do not dump other netns's conntrack entries via proc").	2026-04-22	5.5
CVE-2026-31497	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: Bluetooth: btusb: clamp SCO altsetting table indices btusb_work() maps the number of active SCO links to USB alternate settings through a three-entry lookup table when CVSD traffic uses transparent voice settings. The lookup currently indexes alts[] with data->sco_num - 1 without first constraining sco_num to the number of available table entries. While the table only defines alternate settings for up to three SCO links, data->sco_num comes from hci_conn_num() and is used directly. Cap the lookup to the last table entry before indexing it so the driver keeps selecting the highest supported alternate setting without reading past alts[].	2026-04-22	5.5
CVE-2026-31498	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: Bluetooth: L2CAP: Fix ERTM re-init and zero pdu_len infinite loop l2cap_config_req() processes CONFIG_REQ for channels in BT_CONNECTED state to support L2CAP reconfiguration (e.g. MTU changes). However, since both CONF_INPUT_DONE and CONF_OUTPUT_DONE are already set from the initial configuration, the reconfiguration path falls through to l2cap_ertm_init(), which re-initializes tx_q, srej_q, srej_list, and retrans_list without freeing the previous allocations and sets chan->sdu to NULL without freeing the existing skb. This leaks all previously allocated ERTM resources. Additionally, l2cap_parse_conf_req() does not validate the minimum value of remote_mps derived from the RFC max_pdu_size option. A zero value propagates to l2cap_segment_sdu() where pdu_len becomes zero, causing the while loop to never terminate since len is never decremented, exhausting all available memory. Fix the double-init by skipping l2cap_ertm_init() and l2cap_chan_ready() when the channel is already in BT_CONNECTED state, while still allowing the reconfiguration parameters to be updated through l2cap_parse_conf_req(). Also add a pdu_len zero check in l2cap_segment_sdu() as a safeguard.	2026-04-22	5.5
CVE-2026-31499	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: Bluetooth: L2CAP: Fix deadlock in l2cap_conn_del() l2cap_conn_del() calls cancel_delayed_work_sync() for both info_timer and id_addr_timer while holding conn->lock. However, the work functions l2cap_info_timeout() and l2cap_conn_update_id_addr() both acquire conn->lock, creating a potential AB-BA deadlock if the work is already executing when l2cap_conn_del() takes the lock. Move the work cancellations before acquiring conn->lock and use disable_delayed_work_sync() to additionally prevent the works from being rearmed after cancellation, consistent with the pattern used in hci_conn_del().	2026-04-22	5.5
CVE-2026-31503	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: udp: Fix wildcard bind conflict check when using hash2 When binding a udp_sock to a local address and port, UDP uses two hashes (udptable->hash and udptable->hash2) for collision	2026-04-22	5.5

		detection. The current code switches to "hash2" when hslot->count > 10. "hash2" is keyed by local address and local port. "hash" is keyed by local port only. The issue can be shown in the following bind sequence (pseudo code): bind(fd1, "[fd00::1]:8888") bind(fd2, "[fd00::2]:8888") bind(fd3, "[fd00::3]:8888") bind(fd4, "[fd00::4]:8888") bind(fd5, "[fd00::5]:8888") bind(fd6, "[fd00::6]:8888") bind(fd7, "[fd00::7]:8888") bind(fd8, "[fd00::8]:8888") bind(fd9, "[fd00::9]:8888") bind(fd10, "[fd00::10]:8888") /* Correctly return -EADDRINUSE because "hash" is used * instead of "hash2". udp_lib_lport_inuse() detects the * conflict. */ bind(fail_fd, "[::]:8888") /* After one more socket is bound to "[fd00::11]:8888", * hslot->count exceeds 10 and "hash2" is used instead. */ bind(fd11, "[fd00::11]:8888") bind(fail_fd, "[::]:8888") /* succeeds unexpectedly */ The same issue applies to the IPv4 wildcard address "0.0.0.0" and the IPv4-mapped wildcard address "::ffff:0.0.0.0". For example, if there are existing sockets bound to "192.168.1.[1-11]:8888", then binding "0.0.0.0:8888" or "[::ffff:0.0.0.0]:8888" can also miss the conflict when hslot->count > 10. TCP inet_csk_get_port() already has the correct check in inet_use_bhash2_on_bind(). Rename it to inet_use_hash2_on_bind() and move it to inet_hashtables.h so udp.c can reuse it in this fix.		
CVE-2026-31509	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: nfc: nci: fix circular locking dependency in nci_close_device nci_close_device() flushes rx_wq and tx_wq while holding req_lock. This causes a circular locking dependency because nci_rx_work() running on rx_wq can end up taking req_lock too: nci_rx_work -> nci_rx_data_packet -> nci_data_exchange_complete -> __sk_destruct -> rawsock_destruct -> nfc_deactivate_target -> nci_deactivate_target -> nci_request -> mutex_lock(&ndev->req_lock) Move the flush of rx_wq after req_lock has been released. This should safe (I think) because NCI_UP has already been cleared and the transport is closed, so the work will see it and return -ENETDOWN. NIPA has been hitting this running the nci selftest with a debug kernel on roughly 4% of the runs.	2026-04-22	5.5
CVE-2026-31510	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: Bluetooth: L2CAP: Fix null-ptr-deref on l2cap_sock_ready_cb Before using sk pointer, check if it is null. Fix the following: KASAN: null-ptr-deref in range [0x0000000000000260-0x0000000000000267] CPU: 0 UID: 0 PID: 5985 Comm: kworker/0:5 Not tainted 7.0.0-rc4-00029-ga989fde763f4 #1 PREEMPT(full) Hardware name: QEMU Standard PC (Q35 + ICH9, 2009), BIOS 1.17.0-9.fc43 06/10/2025 Workqueue: events l2cap_info_timeout RIP: 0010:kasan_byte_accessible+0x12/0x30 Code: 79 ff ff ff 0f 1f 40 00 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 0f 1f 40 d6 48 c1 ef 03 48 b8 00 00 00 00 00 fc ff df <0f> b6 04 07 3c 08 0f 92 c0 c3 cc cce veth0_macvtap: entered promiscuous mode RSP: 0018:ffffc90006e0f808 EFLAGS: 00010202 RAX: dfffc0000000000 RBX: ffffffff89746018 RCX: 0000000080000001	2026-04-22	5.5

		RDX: 0000000000000000 RSI: ffffffff89746018 RDI: 000000000000004c RBP: 0000000000000000 R08: 0000000000000001 R09: 0000000000000000 R10: dfffc0000000000 R11: ffffffff8aae3e70 R12: 0000000000000000 R13: 0000000000000260 R14: 0000000000000260 R15: 0000000000000001 FS: 0000000000000000(0000) GS:ffff8880983c2000(0000) knlGS:0000000000000000 CS: 0010 DS: 0000 ES: 0000 CR0: 0000000080050033 CR2: 00005582615a5008 CR3: 000000007007e000 CR4: 0000000000752ef0 PKRU: 55555554 Call Trace: <TASK> __kasan_check_byte+0x12/0x40 lock_acquire+0x79/0x2e0 lock_sock_nested+0x48/0x100 ? l2cap_sock_ready_cb+0x46/0x160 l2cap_sock_ready_cb+0x46/0x160 l2cap_conn_start+0x779/0xff0 ? __pfx_l2cap_conn_start+0x10/0x10 ? l2cap_info_timeout+0x60/0xa0 ? __pfx___mutex_lock+0x10/0x10 l2cap_info_timeout+0x68/0xa0 ? process_scheduled_works+0xa8d/0x18c0 process_scheduled_works+0xb6e/0x18c0 ? __pfx_process_scheduled_works+0x10/0x10 ? assign_work+0x3d5/0x5e0 worker_thread+0xa53/0xfc0 kthread+0x388/0x470 ? __pfx_worker_thread+0x10/0x10 ? __pfx_kthread+0x10/0x10 ret_from_fork+0x51e/0xb90 ? __pfx_ret_from_fork+0x10/0x10 veth1_macvtap: entered promiscuous mode ? __switch_to+0xc7d/0x1450 ? __pfx_kthread+0x10/0x10 ret_from_fork_asm+0x1a/0x30 </TASK> Modules linked in: ---[end trace 0000000000000000]--- batman_adv: batadv0: Interface activated: batadv_slave_0 batman_adv: batadv0: Interface activated: batadv_slave_1 netdevsim netdevsim7 netdevsim0: set [1, 0] type 2 family 0 port 6081 - 0 netdevsim netdevsim7 netdevsim1: set [1, 0] type 2 family 0 port 6081 - 0 netdevsim netdevsim7 netdevsim2: set [1, 0] type 2 family 0 port 6081 - 0 netdevsim netdevsim7 netdevsim3: set [1, 0] type 2 family 0 port 6081 - 0 RIP: 0010:kasan_byte_accessible+0x12/0x30 Code: 79 ff ff ff 0f 1f 40 00 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 0f 1f 40 d6 48 c1 ef 03 48 b8 00 00 00 00 00 fc ff df <0f> b6 04 07 3c 08 0f 92 c0 c3 cc cce ieee80211 phy39: Selected rate control algorithm 'minstrel_ht' RSP: 0018:ffffc90006e0f808 EFLAGS: 00010202 RAX: dfffc0000000000 RBX: ffffffff89746018 RCX: 0000000080000001 RDX: 0000000000000000 RSI: ffffffff89746018 RDI: 000000000000004c RBP: 0000000000000000 R08: 0000000000000001 R09: 0000000000000000 R10: dfffc0000000000 R11: ffffffff8aae3e70 R12: 0000000000000000 R13: 0000000000000260 R14: 0000000000000260 R15: 0000000000000001 FS: 0000000000000000(0000) GS:ffff8880983c2000(0000) knlGS:0000000000000000 CS: 0010 DS: 0000 ES: 0000 CR0: 0000000080050033 CR2: 00007f7e16139e9c CR3: 000000000e74e000 CR4: 0000000000752ef0 PKRU: 55555554 Kernel panic - not syncing: Fatal exception		
CVE-2026-31512	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: Bluetooth: L2CAP: Validate PDU length before reading SDU length in l2cap_ecred_data_rcv() l2cap_ecred_data_rcv() reads the SDU length field from skb->data using get_unaligned_le16() without first verifying that skb contains at least L2CAP_SDULEN_SIZE (2) bytes. When skb->len is less than 2, this reads past the valid data in the skb. The ERTM reassembly path correctly calls pskb_may_pull() before reading the SDU length (l2cap_reassemble_sdu, L2CAP_SAR_START case). Apply the same validation to the Enhanced Credit Based Flow Control data path.	2026-04-22	5.5
CVE-2026-31514	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: erofs: set fileio bio failed in short read case For file-backed mount, IO requests are handled by vfs_iocb_iter_read(). However, it can be interrupted by SIGKILL, returning the number of bytes actually copied. Unused folios in bio are unexpectedly marked as uptodate. vfs_read	2026-04-22	5.5

		filemap_read filemap_get_pages filemap_readahead erofs_fileio_readahead erofs_fileio_rq_submit vfs_iocb_iter_read filemap_read filemap_get_pages <= detect signal erofs_fileio_ki_complete <= set all folios uptodate This patch addresses this by setting short read bio with an error directly.		
CVE-2026-31515	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: af_key: validate families in pfkey_send_migrate() syzbot was able to trigger a crash in skb_put() [1] Issue is that pfkey_send_migrate() does not check old/new families, and that set_ipsecrequest() @family argument was truncated, thus possibly overfilling the skb. Validate families early, do not wait set_ipsecrequest(). [1] skbuff: skb_over_panic: text:ffffffff8a752120 len:392 put:16 head:ffff88802a4ad040 data:ffff88802a4ad040 tail:0x188 end:0x180 dev:<NULL> kernel BUG at net/core/skbuff.c:214 ! Call Trace: <TASK> skb_over_panic net/core/skbuff.c:219 [inline] skb_put+0x159/0x210 net/core/skbuff.c:2655 skb_put_zero include/linux/skbuff.h:2788 [inline] set_ipsecrequest net/key/af_key.c:3532 [inline] pfkey_send_migrate+0x1270/0x2e50 net/key/af_key.c:3636 km_migrate+0x155/0x260 net/xfrm/xfrm_state.c:2848 xfrm_migrate+0x2140/0x2450 net/xfrm/xfrm_policy.c:4705 xfrm_do_migrate+0x8ff/0xaa0 net/xfrm/xfrm_user.c:3150	2026-04-22	5.5
CVE-2026-31517	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: xfrm: iptfs: fix skb_put() panic on non-linear skb during reassembly In iptfs_reassem_cont(), IP-TFS attempts to append data to the new inner packet 'newskb' that is being reassembled. First a zero-copy approach is tried if it succeeds then newskb becomes non-linear. When a subsequent fragment in the same datagram does not meet the fast-path conditions, a memory copy is performed. It calls skb_put() to append the data and as newskb is non-linear it triggers SKB_LINEAR_ASSERT check. Oops: invalid opcode: 0000 [#1] SMP NOPTI [...] RIP: 0010:skb_put+0x3c/0x40 [...] Call Trace: <IRQ> iptfs_reassem_cont+0x1ab/0x5e0 [xfrm_iptfs] iptfs_input_ordered+0x2af/0x380 [xfrm_iptfs] iptfs_input+0x122/0x3e0 [xfrm_iptfs] xfrm_input+0x91e/0x1a50 xfrm4_esp_rcv+0x3a/0x110 ip_protocol_deliver_rcu+0x1d7/0x1f0 ip_local_deliver_finish+0xbe/0x1e0 __netif_receive_skb_core.constprop.0+0xb56/0x1120 __netif_receive_skb_list_core+0x133/0x2b0 netif_receive_skb_list_internal+0x1ff/0x3f0 napi_complete_done+0x81/0x220 virtnet_poll+0x9d6/0x116e [virtio_net] __napi_poll.constprop.0+0x2b/0x270 net_rx_action+0x162/0x360 handle_softirqs+0xdc/0x510 __irq_exit_rcu+0xe7/0x110 irq_exit_rcu+0xe/0x20 common_interrupt+0x85/0xa0 </IRQ> <TASK> Fix this by checking if the skb is non-linear. If it is, linearize it by	2026-04-22	5.5

		calling skb_linearize(). As the initial allocation of newskb originally reserved enough tailroom for the entire reassembled packet we do not need to check if we have enough tailroom or extend it.		
CVE-2026-31518	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: esp: fix skb leak with espintcp and async crypto When the TX queue for espintcp is full, esp_output_tail_tcp will return an error and not free the skb, because with synchronous crypto, the common xfrm output code will drop the packet for us. With async crypto (esp_output_done), we need to drop the skb when esp_output_tail_tcp returns an error.	2026-04-22	5.5
CVE-2026-31519	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: btrfs: set BTRFS_ROOT_ORPHAN_CLEANUP during subvol create We have recently observed a number of subvolumes with broken dentries. ls-ing the parent dir looks like: drwxrwxrwt 1 root root 16 Jan 23 16:49 . drwxr-xr-x 1 root root 24 Jan 23 16:48 .. d????????? ? ? ? ? ? broken_subvol and similarly stat-ing the file fails. In this state, deleting the subvol fails with ENOENT, but attempting to create a new file or subvol over it errors out with EEXIST and even aborts the fs. Which leaves us a bit stuck. dmesg contains a single notable error message reading: "could not do orphan cleanup -2" 2 is ENOENT and the error comes from the failure handling path of btrfs_orphan_cleanup(), with the stack leading back up to btrfs_lookup(). btrfs_lookup btrfs_lookup_dentry btrfs_orphan_cleanup // prints that message and returns -ENOENT After some detailed inspection of the internal state, it became clear that: - there are no orphan items for the subvol - the subvol is otherwise healthy looking, it is not half-deleted or anything, there is no drop progress, etc. - the subvol was created a while ago and does the meaningful first btrfs_orphan_cleanup() call that sets BTRFS_ROOT_ORPHAN_CLEANUP much later. - after btrfs_orphan_cleanup() fails, btrfs_lookup_dentry() returns -ENOENT, which results in a negative dentry for the subvolume via d_splice_alias(NULL, dentry), leading to the observed behavior. The bug can be mitigated by dropping the dentry cache, at which point we can successfully delete the subvolume if we want. i.e., btrfs_lookup() btrfs_lookup_dentry() if (!sb_rdonly(inode->vfs_inode)->vfs_inode) btrfs_orphan_cleanup(sub_root) test_and_set_bit(BTRFS_ROOT_ORPHAN_CLEANUP) btrfs_search_slot() // finds orphan item for inode N ... prints "could not do orphan cleanup -2" if (inode == ERR_PTR(-ENOENT)) inode = NULL; return d_splice_alias(NULL, dentry) // NEGATIVE DENTRY for valid subvolume btrfs_orphan_cleanup() does test_and_set_bit(BTRFS_ROOT_ORPHAN_CLEANUP) on the root when it runs, so it cannot run more than once on a given root, so something else must run concurrently. However, the obvious routes to deleting an orphan when nlinks goes to 0 should not be able to run without first doing a lookup into the subvolume, which should run btrfs_orphan_cleanup() and set the bit. The final important observation is that create_subvol() calls d_instantiate_new() but does not set BTRFS_ROOT_ORPHAN_CLEANUP, so if the dentry cache gets dropped, the next lookup into the subvolume will make a real call into btrfs_orphan_cleanup() for the first time. This opens up the possibility of concurrently deleting the inode/orphan items	2026-04-22	5.5

		but most typical evict() paths will be holding a reference on the parent dentry (child dentry holds parent->d_lockref.count via dget in d_alloc(), released in __dentry_kill()) and prevent the parent from being removed from the dentry cache. The one exception is delayed iputs. Ordered extent creation calls igrab() on the inode. If the file is unlinked and closed while those refs are held, iput() in __dentry_kill() decrements i_count but does not trigger eviction (i_count > 0). The child dentry is freed and the subvol dentry's d_lockref.count drops to 0, making it evictable while the inode is still alive. Since there are two races (the race between writeback and unlink and the race between lookup and delayed iputs), and there are too many moving parts, the following three diagrams show the complete picture. (Only the second and third are races) Phase 1: Create Subvol in dentry cache without BTRFS_ROOT_ORPHAN_CLEANUP set btrfs_mksubvol() lookup_one_len() __lookup_slow() d_alloc_parallel() __d_alloc() // d_lockref.count = 1 create_subvol(dentry) // doesn't touch the bit.. d_instantiate_new(dentry, inode) // dentry in cache with d_lockref.c ---truncated---		
CVE-2026-31520	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: HID: apple: avoid memory leak in apple_report_fixup() The apple_report_fixup() function was returning a newly kmemdup()-allocated buffer, but never freeing it. The caller of report_fixup() does not take ownership of the returned pointer, but it *is* permitted to return a sub-portion of the input rdesc, whose lifetime is managed by the caller.	2026-04-22	5.5
CVE-2026-31521	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: module: Fix kernel panic when a symbol st_shndx is out of bounds The module loader doesn't check for bounds of the ELF section index in simplify_symbols(): for (i = 1; i < symsec->sh_size / sizeof(Elf_Sym); i++) { const char *name = info->strtab + sym[i].st_name; switch (sym[i].st_shndx) { case SHN_COMMON: [...] default: /* Divert to percpu allocation if a percpu var. */ if (sym[i].st_shndx == info->index.pcpu) secbase = (unsigned long)mod_percpu(mod); else /** HERE --> **/ secbase = info->sechdrs[sym[i].st_shndx].sh_addr; sym[i].st_value += secbase; break; } } A symbol with an out-of-bounds st_shndx value, for example 0xffff (known as SHN_XINDEX or SHN_HIRESERVE), may cause a kernel panic: BUG: unable to handle page fault for address: ... RIP: 0010:simplify_symbols+0x2b2/0x480 ... Kernel panic - not syncing: Fatal exception This can happen when module ELF is legitimately using SHN_XINDEX or when it is corrupted. Add a bounds check in simplify_symbols() to validate that st_shndx is within the valid range before using it. This issue was discovered due to a bug in llvm-objcopy, see relevant	2026-04-22	5.5

		discussion for details [1]. [1] https://lore.kernel.org/linux-modules/20251224005752.201911-1-ihor.solodrai@linux.dev/		
CVE-2026-31522	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: HID: magicmouse: avoid memory leak in magicmouse_report_fixup() The magicmouse_report_fixup() function was returning a newly kmemdup()-allocated buffer, but never freeing it. The caller of report_fixup() does not take ownership of the returned pointer, but it *is* permitted to return a sub-portion of the input rdesc, whose lifetime is managed by the caller.	2026-04-22	5.5
CVE-2026-31524	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: HID: asus: avoid memory leak in asus_report_fixup() The asus_report_fixup() function was returning a newly allocated kmemdup()-allocated buffer, but never freeing it. Switch to devm_kzalloc() to ensure the memory is managed and freed automatically when the device is removed. The caller of report_fixup() does not take ownership of the returned pointer, but it is permitted to return a pointer whose lifetime is at least that of the input buffer. Also fix a harmless out-of-bounds read by copying only the original descriptor size.	2026-04-22	5.5
CVE-2026-31526	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: bpf: Fix exception exit lock checking for subprogs process_bpf_exit_full() passes check_lock = !curframe to check_resource_leak(), which is false in cases when bpf_throw() is called from a static subprog. This makes check_resource_leak() to skip validation of active_rcu_locks, active_preempt_locks, and active_irq_id on exception exits from subprogs. At runtime bpf_throw() unwinds the stack via ORC without releasing any user-acquired locks, which may cause various issues as the result. Fix by setting check_lock = true for exception exits regardless of curframe, since exceptions bypass all intermediate frame cleanup. Update the error message prefix to "bpf_throw" for exception exits to distinguish them from normal BPF_EXIT. Fix reject_subprog_with_rcu_read_lock test which was previously passing for the wrong reason. Test program returned directly from the subprog call without closing the RCU section, so the error was triggered by the unclosed RCU lock on normal exit, not by bpf_throw. Update __msg annotations for affected tests to match the new "bpf_throw" error prefix. The spin_lock case is not affected because they are already checked [1] at the call site in do_check_insn() before bpf_throw can run. [1] https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/tree/kernel/bpf/verifier.c?h=v7.0-rc4#n21098	2026-04-22	5.5
CVE-2026-31529	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: cxl/region: Fix leakage in __construct_region() Failing the first sysfs_update_group() needs to explicitly kfree the resource as it is too early for cxl_region_iomem_release() to do so.	2026-04-22	5.5
CVE-2026-6862	red hat - multiple products	A flaw was found in libefiboot, a component of efivar. The device path node parser in libefiboot fails to validate that each node's Length field is at least 4 bytes, which is the minimum size for an EFI (Extensible Firmware Interface) device path node header. A local user could exploit this vulnerability by providing a specially crafted device path node. This can lead to infinite recursion, causing stack exhaustion and a process crash, resulting in a denial of service (DoS).	2026-04-22	5.5
CVE-2025-36074	ibm - Security Verify Directory (Container)	IBM Security Verify Directory (Container) 10.0.0 through 10.0.0.3 IBM Security Verify Directory could be vulnerable to malicious file upload by not validating file type. A privileged user could upload malicious files into the system that can be sent to victims for performing further attacks against the system.	2026-04-23	5.5
CVE-2026-4918	ibm - guardium_data_protection	IBM Guardium Data Protection 12.1 is vulnerable to stored cross-site scripting. This vulnerability allows an administrative user to embed arbitrary JavaScript code in the Web UI thus altering the intended functionality potentially leading to credentials disclosure within a trusted session.	2026-04-23	5.5
CVE-2026-31531	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved:	2026-04-23	5.5

		ipv4: nexthop: allocate skb dynamically in rtm_get_nexthop() When querying a nexthop object via RTM_GETNEXTHOP, the kernel currently allocates a fixed-size skb using NLMSG_GOODSIZE. While sufficient for single nexthops and small Equal-Cost Multi-Path groups, this fixed allocation fails for large nexthop groups like 512 nexthops. This results in the following warning splat: WARNING: net/ipv4/nexthop.c:3395 at rtm_get_nexthop+0x176/0x1c0, CPU#20: rep/4608 [...] RIP: 0010:rtm_get_nexthop (net/ipv4/nexthop.c:3395) [...] Call Trace: <TASK> rtnetlink_rcv_msg (net/core/rtnetlink.c:6989) netlink_rcv_skb (net/netlink/af_netlink.c:2550) netlink_unicast (net/netlink/af_netlink.c:1319 net/netlink/af_netlink.c:1344) netlink_sendmsg (net/netlink/af_netlink.c:1894) ____sys_sendmsg (net/socket.c:721 net/socket.c:736 net/socket.c:2585) ___sys_sendmsg (net/socket.c:2641) __sys_sendmsg (net/socket.c:2671) do_syscall_64 (arch/x86/entry/syscall_64.c:63 arch/x86/entry/syscall_64.c:94) entry_SYSCALL_64_after_hwframe (arch/x86/entry/entry_64.S:130) </TASK> Fix this by allocating the size dynamically using nh_nlmsg_size() and using nlmsg_new(), this is consistent with nexthop_notify() behavior. In addition, adjust nh_nlmsg_size_grp() so it calculates the size needed based on flags passed. While at it, also add the size of NHA_FDB for nexthop group size calculation as it was missing too. This cannot be reproduced via iproute2 as the group size is currently limited and the command fails as follows: addattr_l ERROR: message exceeded bound of 1048		
CVE-2026-31537	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: smb: server: make use of smbdirect_socket.send_io.bcredits It turns out that our code will corrupt the stream of reassabled data transfer messages when we trigger an immendiate (empty) send. In order to fix this we'll have a single 'batch' credit per connection. And code getting that credit is free to use as much messages until remaining_length reaches 0, then the batch credit it given back and the next logical send can happen.	2026-04-24	5.5
CVE-2026-31540	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: drm/i915/gt: Check set_default_submission() before deferencing When the i915 driver firmware binaries are not present, the set_default_submission pointer is not set. This pointer is dereferenced during suspend anyways. Add a check to make sure it is set before dereferencing. [23.289926] PM: suspend entry (deep) [23.293558] Filesystems sync: 0.000 seconds [23.298010] Freezing user space processes [23.302771] Freezing user space processes completed (elapsed 0.000 seconds) [23.309766] OOM killer disabled. [23.313027] Freezing remaining freezable tasks [23.318540] Freezing remaining freezable tasks completed (elapsed 0.001 seconds) [23.342038] serial 00:05: disabled [23.345719] serial 00:02: disabled [23.349342] serial 00:01: disabled [23.353782] sd 0:0:0:0: [sda] Synchronizing SCSI cache [23.358993] sd 1:0:0:0: [sdb] Synchronizing SCSI cache [23.361635] ata1.00: Entering standby power mode [23.368863] ata2.00: Entering standby power mode [23.445187] BUG: kernel NULL pointer dereference, address: 0000000000000000 [23.452194] #PF: supervisor instruction fetch in kernel mode [23.457896] #PF: error_code(0x0010) - not-present page [23.463065] PGD 0 P4D 0 [23.465640] Oops: Oops: 0010 [#1] SMP NOPTI [23.469869] CPU: 8 UID: 0 PID: 211 Comm: kworker/u48:18 Tainted: G S W 6.19.0-rc4-00020-gf0b9d8eb98df #10 PREEMPT(voluntary)	2026-04-24	5.5

		[23.482512] Tainted: [S]=CPU_OUT_OF_SPEC, [W]=WARN [23.496511] Workqueue: async async_run_entry_fn [23.501087] RIP: 0010:0x0 [23.503755] Code: Unable to access opcode bytes at 0xffffffffffffffd6. [23.510324] RSP: 0018:ffffb4a60065fca8 EFLAGS: 00010246 [23.515592] RAX: 0000000000000000 RBX: ffff9f428290e000 RCX: 000000000000000f [23.522765] RDX: 0000000000000000 RSI: 0000000000000282 RDI: ffff9f428290e000 [23.529937] RBP: ffff9f4282907070 R08: ffff9f4281130428 R09: 00000000ffffffff [23.537111] R10: 0000000000000000 R11: 0000000000000001 R12: ffff9f42829070f8 [23.544284] R13: ffff9f4282906028 R14: ffff9f4282900000 R15: ffff9f4282906b68 [23.551457] FS: 0000000000000000(0000) GS:ffff9f466b2cf000(0000) knlGS:0000000000000000 [23.559588] CS: 0010 DS: 0000 ES: 0000 CR0: 0000000080050033 [23.565365] CR2: ffffffffffffffd6 CR3: 000000031c230001 CR4: 0000000000f70ef0 [23.572539] PKRU: 55555554 [23.575281] Call Trace: [23.577770] <TASK> [23.579905] intel_engines_reset_default_submission+0x42/0x60 [23.585695] __intel_gt_unset_wedged+0x191/0x200 [23.590360] intel_gt_unset_wedged+0x20/0x40 [23.594675] gt_sanitize+0x15e/0x170 [23.598290] i915_gem_suspend_late+0x6b/0x180 [23.602692] i915_drm_suspend_late+0x35/0xf0 [23.607008] ? __pfx_pci_pm_suspend_late+0x10/0x10 [23.611843] dpm_run_callback+0x78/0x1c0 [23.615817] device_suspend_late+0xde/0x2e0 [23.620037] async_suspend_late+0x18/0x30 [23.624082] async_run_entry_fn+0x25/0xa0 [23.628129] process_one_work+0x15b/0x380 [23.632182] worker_thread+0x2a5/0x3c0 [23.635973] ? __pfx_worker_thread+0x10/0x10 [23.640279] kthread+0xf6/0x1f0 [23.643464] ? __pfx_kthread+0x10/0x10 [23.647263] ? __pfx_kthread+0x10/0x10 [23.651045] ret_from_fork+0x131/0x190 [23.654837] ? __pfx_kthread+0x10/0x10 [23.658634] ret_from_fork_asm+0x1a/0x30 [23.662597] </TASK> [23.664826] Modules linked in: [23.667914] CR2: 0000000000000000 [23.671271] ----------[cut here]---------- (cherry picked from commit daa199abc3d3d1740c9e3a2c3e9216ae5b447cad)		
CVE-2026-31542	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: x86/platform/uv: Handle deconfigured sockets When a socket is deconfigured, it's mapped to SOCK_EMPTY (0xffff). This causes a panic while allocating UV hub info structures. Fix this by using NUMA_NO_NODE, allowing UV hub info structures to be allocated on valid nodes.	2026-04-24	5.5
CVE-2026-31543	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: crash_dump: don't log dm-crypt key bytes in read_key_from_user_keying When debug logging is enabled, read_key_from_user_keying() logs the first 8 bytes of the key payload and partially exposes the dm-crypt key. Stop logging any key bytes.	2026-04-24	5.5
CVE-2026-31544	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: firmware: arm_scmi: Fix NULL dereference on notify error path Since commit b5daf93b809d1 ("firmware: arm_scmi: Avoid notifier registration for unsupported events") the call chains leading to the helper __scmi_event_handler_get_ops expect an ERR_PTR to be returned on failure to get an handler for the requested event key, while the current helper can still return a NULL when no handler could be found or created. Fix by forcing an ERR_PTR return value when the handler reference is NULL.	2026-04-24	5.5
CVE-2026-31545	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: NFC: nxp-nci: allow GPIOs to sleep Allow the firmware and enable GPIOs to sleep. This fixes a `WARN_ON' and allows the driver to operate GPIOs which are connected to I2C GPIO expanders. -- >8 --	2026-04-24	5.5

		kernel: WARNING: CPU: 3 PID: 2636 at drivers/gpio/gpiolib.c:3880 gpiod_set_value+0x88/0x98 -- >8 --		
CVE-2026-31546	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net: bonding: fix NULL deref in bond_debug_rlb_hash_show rlb_clear_slave intentionally keeps RLB hash-table entries on the rx_hashtbl_used_head list with slave set to NULL when no replacement slave is available. However, bond_debug_rlb_hash_show visites client_info->slave without checking if it's NULL. Other used-list iterators in bond_alb.c already handle this NULL-slave state safely: - rlb_update_client returns early on !client_info->slave - rlb_req_update_slave_clients, rlb_clear_slave, and rlb_rebalance compare slave values before visiting - lb_req_update_subnet_clients continues if slave is NULL The following NULL deref crash can be trigger in bond_debug_rlb_hash_show: [1.289791] BUG: kernel NULL pointer dereference, address: 0000000000000000 [1.292058] RIP: 0010:bond_debug_rlb_hash_show (drivers/net/bonding/bond_debugfs.c:41) [1.293101] RSP: 0018:ffffc900004a7d00 EFLAGS: 00010286 [1.293333] RAX: 0000000000000000 RBX: ffff888102b48200 RCX: ffff888102b48204 [1.293631] RDX: ffff888102b48200 RSI: ffffffff839daad5 RDI: ffff888102815078 [1.293924] RBP: ffff888102815078 R08: ffff888102b4820e R09: 0000000000000000 [1.294267] R10: 0000000000000000 R11: 0000000000000000 R12: ffff888100f929c0 [1.294564] R13: ffff888100f92a00 R14: 0000000000000001 R15: ffffc900004a7ed8 [1.294864] FS: 0000000001395380(0000) GS:ffff888196e75000(0000) knlGS:0000000000000000 [1.295239] CS: 0010 DS: 0000 ES: 0000 CR0: 0000000080050033 [1.295480] CR2: 0000000000000000 CR3: 0000000102adc004 CR4: 0000000000772ef0 [1.295897] Call Trace: [1.296134] seq_read_iter (fs/seq_file.c:231) [1.296341] seq_read (fs/seq_file.c:164) [1.296493] full_proxy_read (fs/debugfs/file.c:378 (discriminator 1)) [1.296658] vfs_read (fs/read_write.c:572) [1.296981] ksys_read (fs/read_write.c:717) [1.297132] do_syscall_64 (arch/x86/entry/syscall_64.c:63 (discriminator 1) arch/x86/entry/syscall_64.c:94 (discriminator 1)) [1.297325] entry_SYSCALL_64_after_hwframe (arch/x86/entry/entry_64.S:130) Add a NULL check and print "(none)" for entries with no assigned slave.	2026-04-24	5.5
CVE-2026-31547	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: drm/xe: Fix missing runtime PM reference in ccs_mode_store ccs_mode_store() calls xe_gt_reset() which internally invokes xe_pm_runtime_get_noresume(). That function requires the caller to already hold an outer runtime PM reference and warns if none is held: [46.891177] xe 0000:03:00.0: [drm] Missing outer runtime PM protection [46.891178] WARNING: drivers/gpu/drm/xe/xe_pm.c:885 at xe_pm_runtime_get_noresume+0x8b/0xc0 Fix this by protecting xe_gt_reset() with the scope-based guard(xe_pm_runtime)(xe), which is the preferred form when the reference lifetime matches a single scope. v2: - Use scope-based guard(xe_pm_runtime)(xe) (Shuicheng) - Update commit message accordingly (cherry picked from commit 7937ea733f79b3f25e802a0c8360bf7423856f36)	2026-04-24	5.5
CVE-2026-31549	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: i2c: cp2615: fix serial string NULL-deref at probe The cp2615 driver uses the USB device serial string as the i2c adapter name but does not make sure that the string exists. Verify that the device has a serial number before accessing it to avoid triggering a NULL-pointer dereference (e.g. with malicious devices).	2026-04-24	5.5
CVE-2026-31550	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: pmdomain: bcm: bcm2835-power: Increase ASB control timeout The bcm2835_asb_control() function uses a tight polling loop to wait	2026-04-24	5.5

		for the ASB bridge to acknowledge a request. During intensive workloads, this handshake intermittently fails for V3D's master ASB on BCM2711, resulting in "Failed to disable ASB master for v3d" errors during runtime PM suspend. As a consequence, the failed power-off leaves V3D in a broken state, leading to bus faults or system hangs on later accesses. As the timeout is insufficient in some scenarios, increase the polling timeout from 1us to 5us, which is still negligible in the context of a power domain transition. Also, replace the open-coded ktime_get_ns()/ cpu_relax() polling loop with readl_poll_timeout_atomic().		
CVE-2026-31551	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: wifi: mac80211: Fix static_branch_dec() underflow for aql_disable. syzbot reported static_branch_dec() underflow in aql_enable_write(). [0] The problem is that aql_enable_write() does not serialise concurrent write()s to the debugfs. aql_enable_write() checks static_key_false(&aql_disable.key) and later calls static_branch_inc() or static_branch_dec(), but the state may change between the two calls. aql_disable does not need to track inc/dec. Let's use static_branch_enable() and static_branch_disable(). [0]: val == 0 WARNING: kernel/jump_label.c:311 at __static_key_slow_dec_cpuslocked.part.0+0x107/0x120 kernel/jump_label.c:311, CPU#0: syz.1.3155/20288 Modules linked in: CPU: 0 UID: 0 PID: 20288 Comm: syz.1.3155 Tainted: G U L syzkaller #0 PREEMPT(full) Tainted: [U]=USER, [L]=SOFTLOCKUP Hardware name: Google Google Compute Engine/Google Compute Engine, BIOS Google 01/24/2026 RIP: 0010:__static_key_slow_dec_cpuslocked.part.0+0x107/0x120 kernel/jump_label.c:311 Code: f2 c9 ff 5b 5d c3 cc cc cc cc e8 54 f2 c9 ff 48 89 df e8 ac f9 ff ff eb ad e8 45 f2 c9 ff 90 0f 0b 90 eb a2 e8 3a f2 c9 ff 90 <0f> 0b 90 eb 97 48 89 df e8 5c 4b 33 00 e9 36 ff ff ff 0f 1f 80 00 RSP: 0018:ffffc9000b9f7c10 EFLAGS: 00010293 RAX: 0000000000000000 RBX: ffffffff9b3e5d40 RCX: ffffffff823c57b4 RDX: ffff8880285a0000 RSI: ffffffff823c5846 RDI: ffff8880285a0000 RBP: 0000000000000000 R08: 0000000000000005 R09: 0000000000000000 R10: 0000000000000000 R11: 0000000000000000 R12: 000000000000000a R13: 1ffff9200173ef88 R14: 0000000000000001 R15: ffffc9000b9f7e98 FS: 00007f530dd726c0(0000) GS:ffff8881245e3000(0000) knlGS:0000000000000000 CS: 0010 DS: 0000 ES: 0000 CR0: 0000000080050033 CR2: 0000200000001140 CR3: 000000007cc4a000 CR4: 00000000003526f0 Call Trace: <TASK> __static_key_slow_dec_cpuslocked kernel/jump_label.c:297 [inline] __static_key_slow_dec kernel/jump_label.c:321 [inline] static_key_slow_dec+0x7c/0xc0 kernel/jump_label.c:336 aql_enable_write+0x2b2/0x310 net/mac80211/debugfs.c:343 short_proxy_write+0x133/0x1a0 fs/debugfs/file.c:383 vfs_write+0x2aa/0x1070 fs/read_write.c:684 ksys_pwrite64 fs/read_write.c:793 [inline] __do_sys_pwrite64 fs/read_write.c:801 [inline] __se_sys_pwrite64 fs/read_write.c:798 [inline] __x64_sys_pwrite64+0x1eb/0x250 fs/read_write.c:798 do_syscall_x64 arch/x86/entry/syscall_64.c:63 [inline] do_syscall_64+0xc9/0xf80 arch/x86/entry/syscall_64.c:94 entry_SYSCALL_64_after_hwframe+0x77/0x7f RIP: 0033:0x7f530cf9aeb9 Code: ff c3 66 2e 0f 1f 84 00 00 00 00 00 0f 1f 44 00 00 48 89 f8 48 89 f7 48 89 d6 48 89 ca 4d 89 c2 4d 89 c8 4c 8b 4c 24 08 0f 05 <48> 3d 01 f0 ff ff 73 01 c3 48 c7 c1 e8 ff ff ff f7 d8 64 89 01 48 RSP: 002b:00007f530dd72028 EFLAGS: 00000246 ORIG_RAX: 0000000000000012 RAX: ffffffffffffffda RBX: 00007f530d215fa0 RCX: 00007f530cf9aeb9 RDX: 0000000000000003 RSI: 0000000000000000 RDI: 0000000000000010 RBP: 00007f530d008c1f R08: 0000000000000000 R09: 0000000000000000 R10: 4200000000000005 R11: 0000000000000246 R12: 0000000000000000 R13: 00007f530d216038 R14: 00007f530d215fa0 R15: 00007ffde89fb978 </TASK>	2026-04-24	5.5
CVE-2026-31555	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: futex: Clear stale exiting pointer in futex_lock_pi() retry path Fuzzying/stressing futexes triggered: WARNING: kernel/futex/core.c:825 at wait_for_owner_exiting+0x7a/0x80, CPU#11: futex_lock_pi_s/524	2026-04-24	5.5

		When futex_lock_pi_atomic() sees the owner is exiting, it returns -EBUSY and stores a refcounted task pointer in 'exiting'. After wait_for_owner_exiting() consumes that reference, the local pointer is never reset to nil. Upon a retry, if futex_lock_pi_atomic() returns a different error, the bogus pointer is passed to wait_for_owner_exiting(). CPU0 CPU1 CPU2 futex_lock_pi(uaddr) // acquires the PI futex exit() futex_cleanup_begin() futex_state = EXITING; futex_lock_pi(uaddr) futex_lock_pi_atomic() attach_to_pi_owner() // observes EXITING *exiting = owner; // takes ref return -EBUSY wait_for_owner_exiting(-EBUSY, owner) put_task_struct(); // drops ref // exiting still points to owner goto retry; futex_lock_pi_atomic() lock_pi_update_atomic() cmpxchg(uaddr) *uaddr ^= WAITERS // whatever // value changed return -EAGAIN; wait_for_owner_exiting(-EAGAIN, exiting) // stale WARN_ON_ONCE(exiting) Fix this by resetting upon retry, essentially aligning it with requeue_pi.		
CVE-2026-31556	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: xfs: scrub: unlock dquot before early return in quota scrub xchk_quota_item can return early after calling xchk_fblock_process_error. When that helper returns false, the function returned immediately without dropping dq->q_qlock, which can leave the dquot lock held and risk lock leaks or deadlocks in later quota operations. Fix this by unlocking dq->q_qlock before the early return.	2026-04-24	5.5
CVE-2026-31559	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: LoongArch: Fix missing NULL checks for kstrdup() 1. Replace "of_find_node_by_path("/")" with "of_root" to avoid multiple calls to "of_node_put()". 2. Fix a potential kernel oops during early boot when memory allocation fails while parsing CPU model from device tree.	2026-04-24	5.5
CVE-2026-31560	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: spi: spi-dw-dma: fix print error log when wait finish transaction If an error occurs, the device may not have a current message. In this case, the system will crash. In this case, it's better to use dev from the struct ctlr (struct spi_controller*).	2026-04-24	5.5
CVE-2026-31561	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: x86/cpu: Remove X86_CR4_FRED from the CR4 pinned bits mask Commit in Fixes added the FRED CR4 bit to the CR4 pinned bits mask so that whenever something else modifies CR4, that bit remains set. Which in itself is a perfectly fine idea. However, there's an issue when during boot FRED is initialized: first on the BSP and later on the APs. Thus, there's a window in time when exceptions cannot be handled. This becomes particularly nasty when running as SEV-{ES,SNP} or TDX guests which, when they manage to trigger exceptions during that short window described above, triple fault due to FRED MSRs not being set up yet. See Link tag below for a much more detailed explanation of the situation.	2026-04-24	5.5

		So, as a result, the commit in that Link URL tried to address this shortcoming by temporarily disabling CR4 pinning when an AP is not online yet. However, that is a problem in itself because in this case, an attack on the kernel needs to only modify the online bit - a single bit in RW memory - and then disable CR4 pinning and then disable SM*P, leading to more and worse things to happen to the system. So, instead, remove the FRED bit from the CR4 pinning mask, thus obviating the need to temporarily disable CR4 pinning. If someone manages to disable FRED when poking at CR4, then idt_invalidate() would make sure the system would crash'n'burn on the first exception triggered, which is a much better outcome security-wise.		
CVE-2026-31562	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: drm/mediatek: dsi: Store driver data before invoking mipi_dsi_host_register The call to mipi_dsi_host_register triggers a callback to mtk_dsi_bind, which uses dev_get_drvdata to retrieve the mtk_dsi struct, so this structure needs to be stored inside the driver data before invoking it. As drvdata is currently uninitialized it leads to a crash when registering the DSI DRM encoder right after acquiring the mode_config.idr_mutex, blocking all subsequent DRM operations. Fixes the following crash during mediatek-drm probe (tested on Xiaomi Smart Clock x04g): Unable to handle kernel NULL pointer dereference at virtual address 0000000000000040 [...] Modules linked in: mediatek_drm(+) drm_display_helper cec drm_client_lib drm_dma_helper drm_kms_helper panel_simple [...] Call trace: drm_mode_object_add+0x58/0x98 (P) __drm_encoder_init+0x48/0x140 drm_encoder_init+0x6c/0xa0 drm_simple_encoder_init+0x20/0x34 [drm_kms_helper] mtk_dsi_bind+0x34/0x13c [mediatek_drm] component_bind_all+0x120/0x280 mtk_drm_bind+0x284/0x67c [mediatek_drm] try_to_bring_up_aggregate_device+0x23c/0x320 __component_add+0xa4/0x198 component_add+0x14/0x20 mtk_dsi_host_attach+0x78/0x100 [mediatek_drm] mipi_dsi_attach+0x2c/0x50 panel_simple_dsi_probe+0x4c/0x9c [panel_simple] mipi_dsi_drv_probe+0x1c/0x28 really_probe+0xc0/0x3dc __driver_probe_device+0x80/0x160 driver_probe_device+0x40/0x120 __device_attach_driver+0xbc/0x17c bus_for_each_drv+0x88/0xf0 __device_attach+0x9c/0x1cc device_initial_probe+0x54/0x60 bus_probe_device+0x34/0xa0 device_add+0x5b0/0x800 mipi_dsi_device_register_full+0xdc/0x16c mipi_dsi_host_register+0xc4/0x17c mtk_dsi_probe+0x10c/0x260 [mediatek_drm] platform_probe+0x5c/0xa4 really_probe+0xc0/0x3dc __driver_probe_device+0x80/0x160 driver_probe_device+0x40/0x120 __driver_attach+0xc8/0x1f8 bus_for_each_dev+0x7c/0xe0 driver_attach+0x24/0x30 bus_add_driver+0x11c/0x240 driver_register+0x68/0x130 __platform_register_drivers+0x64/0x160 mtk_drm_init+0x24/0x1000 [mediatek_drm] do_one_initcall+0x60/0x1d0 do_init_module+0x54/0x240 load_module+0x1838/0x1dc0 init_module_from_file+0xd8/0xf0 __arm64_sys_finit_module+0x1b4/0x428	2026-04-24	5.5

		invoke_syscall.constprop.0+0x48/0xc8 do_el0_svc+0x3c/0xb8 el0_svc+0x34/0xe8 el0t_64_sync_handler+0xa0/0xe4 el0t_64_sync+0x198/0x19c Code: 52800022 941004ab 2a0003f3 37f80040 (29005a80)		
CVE-2026-31564	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: LoongArch: KVM: Fix base address calculation in kvm_eiointc_regs_access() In function kvm_eiointc_regs_access(), the register base address is caculated from array base address plus offset, the offset is absolute value from the base address. The data type of array base address is u64, it should be converted into the "void *" type and then plus the offset.	2026-04-24	5.5
CVE-2026-31565	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: RDMA/irdma: Fix deadlock during netdev reset with active connections Resolve deadlock that occurs when user executes netdev reset while RDMA applications (e.g., rping) are active. The netdev reset causes ice driver to remove irdma auxiliary driver, triggering device_delete and subsequent client removal. During client removal, uverbs_client waits for QP reference count to reach zero while cma_client holds the final reference, creating circular dependency and indefinite wait in iWARP mode. Skip QP reference count wait during device reset to prevent deadlock.	2026-04-24	5.5
CVE-2026-31567	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: PM: sleep: Drop spurious WARN_ON() from pm_restore_gfp_mask() Commit 35e4a69b2003f ("PM: sleep: Allow pm_restrict_gfp_mask() stacking") introduced refcount-based GFP mask management that warns when pm_restore_gfp_mask() is called with saved_gfp_count == 0. Some hibernation paths call pm_restore_gfp_mask() defensively where the GFP mask may or may not be restricted depending on the execution path. For example, the uswsusp interface invokes it in SNAPSHOT_CREATE_IMAGE, SNAPSHOT_UNFREEZE, and snapshot_release(). Before the stacking change this was a silent no-op; it now triggers a spurious WARNING. Remove the WARN_ON() wrapper from the !saved_gfp_count check while retaining the check itself, so that defensive calls remain harmless without producing false warnings. [rjw: Subject tweak]	2026-04-24	5.5
CVE-2026-31571	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: drm/i915: Unlink NV12 planes earlier unlink_nv12_plane() will clobber parts of the plane state potentially already set up by plane_atomic_check(), so we must make sure not to call the two in the wrong order. The problem happens when a plane previously selected as a Y plane is now configured as a normal plane by user space. plane_atomic_check() will first compute the proper plane state based on the userspace request, and unlink_nv12_plane() later clears some of the state. This used to work on account of unlink_nv12_plane() skipping the state clearing based on the plane visibility. But I removed that check, thinking it was an impossible situation. Now when that situation happens unlink_nv12_plane() will just WARN and proceed to clobber the state. Rather than reverting to the old way of doing things, I think it's more clear if we unlink the NV12 planes before we even compute the new plane state. (cherry picked from commit 017ecd04985573eeeb0745fa2c23896fb22ee0cc)	2026-04-24	5.5
CVE-2026-31573	linux - linux_kernel	In the Linux kernel, the following vulnerability has been resolved: media: verisilicon: Fix kernel panic due to __initconst misuse Fix a kernel panic when probing the driver as a module: Unable to handle kernel paging request at virtual address ffffd9c18eb05000 of_find_matching_node_and_match+0x5c/0x1a0	2026-04-24	5.5

		hantro_probe+0x2f4/0x7d0 [hantro_vpu] The imx8mq_vpu_shared_resources array is referenced by variant structures through their shared_devices field. When built as a module, __initconst causes this data to be freed after module init, but it's later accessed during probe, causing a page fault. The imx8mq_vpu_shared_resources is referenced from non-init code, so keeping __initconst or __initconst_or_module here is wrong. Drop the __initconst annotation and let it live in the normal .rodata section. A bug of __initconst called from regular non-init probe code leading to bugs during probe deferrals or during unbind-bind cycles.		
CVE-2026-31574	linux - linux_kernel	In the Linux kernel, the following vulnerability has been resolved: clockevents: Add missing resets of the next_event_forced flag The prevention mechanism against timer interrupt starvation missed to reset the next_event_forced flag in a couple of places: - When the clock event state changes. That can cause the flag to be stale over a shutdown/startup sequence - When a non-forced event is armed, which then prevents rearming before that event. If that event is far out in the future this will cause missed timer interrupts. - In the suspend wakeup handler. That led to stalls which have been reported by several people. Add the missing resets, which fixes the problems for the reporters.	2026-04-24	5.5
CVE-2026-31575	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: mm/userfaultfd: fix hugetlb fault mutex hash calculation In mfill_atomic_hugetlb(), linear_page_index() is used to calculate the page index for hugetlb_fault_mutex_hash(). However, linear_page_index() returns the index in PAGE_SIZE units, while hugetlb_fault_mutex_hash() expects the index in huge page units. This mismatch means that different addresses within the same huge page can produce different hash values, leading to the use of different mutexes for the same huge page. This can cause races between faulting threads, which can corrupt the reservation map and trigger the BUG_ON in resv_map_release(). Fix this by introducing hugetlb_linear_page_index(), which returns the page index in huge page granularity, and using it in place of linear_page_index().	2026-04-24	5.5
CVE-2026-31577	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: nilfs2: fix NULL i_assoc_inode dereference in nilfs_mdt_save_to_shadow_map The DAT inode's btree node cache (i_assoc_inode) is initialized lazily during btree operations. However, nilfs_mdt_save_to_shadow_map() assumes i_assoc_inode is already initialized when copying dirty pages to the shadow map during GC. If NILFS_IOCTL_CLEAN_SEGMENTS is called immediately after mount before any btree operation has occurred on the DAT inode, i_assoc_inode is NULL leading to a general protection fault. Fix this by calling nilfs_attach_btree_node_cache() on the DAT inode in nilfs_dat_read() at mount time, ensuring i_assoc_inode is always initialized before any GC operation can use it.	2026-04-24	5.5
CVE-2026-31579	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: wireguard: device: use exit_rtnl callback instead of manual rtnl_lock in pre_exit wg_netns_pre_exit() manually acquires rtnl_lock() inside the pernet .pre_exit callback. This causes a hung task when another thread holds rtnl_mutex - the cleanup_net workqueue (or the setup_net failure rollback path) blocks indefinitely in wg_netns_pre_exit() waiting to acquire the lock. Convert to .exit_rtnl, introduced in commit 7a60d91c690b ("net: Add ->exit_rtnl() hook to struct pernet_operations."), where the framework already holds RTNL and batches all callbacks under a single rtnl_lock()/rtnl_unlock() pair, eliminating the contention	2026-04-24	5.5

		window. The rcu_assign_pointer(wg->creating_net, NULL) is safe to move from .pre_exit to .exit_rtnl (which runs after synchronize_rcu()) because all RCU readers of creating_net either use maybe_get_net() - which returns NULL for a dying namespace with zero refcount - or access net->user_ns which remains valid throughout the entire ops_undo_list sequence. [Jason: added __net_exit and __read_mostly annotations that were missing.]		
CVE-2026-31585	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: media: vidtv: fix nfeeds state corruption on start_streaming failure syzbot reported a memory leak in vidtv_psi_service_desc_init [1]. When vidtv_start_streaming() fails inside vidtv_start_feed(), the nfeeds counter is left incremented even though no feed was actually started. This corrupts the driver state: subsequent start_feed calls see nfeeds > 1 and skip starting the mux, while stop_feed calls eventually try to stop a non-existent stream. This state corruption can also lead to memory leaks, since the mux and channel resources may be partially allocated during a failed start_streaming but never cleaned up, as the stop path finds dvb->streaming == false and returns early. Fix by decrementing nfeeds back when start_streaming fails, keeping the counter in sync with the actual number of active feeds. [1] BUG: memory leak unreferenced object 0xffff888145b50820 (size 32): comm "syz.0.17", pid 6068, jiffies 4294944486 backtrace (crc 90a0c7d4): vidtv_psi_service_desc_init+0x74/0x1b0 drivers/media/test-drivers/vidtv/vidtv_psi.c:288 vidtv_channel_s302m_init+0xb1/0x2a0 drivers/media/test-drivers/vidtv/vidtv_channel.c:83 vidtv_channels_init+0x1b/0x40 drivers/media/test-drivers/vidtv/vidtv_channel.c:524 vidtv_mux_init+0x516/0xbe0 drivers/media/test-drivers/vidtv/vidtv_mux.c:518 vidtv_start_streaming drivers/media/test-drivers/vidtv/vidtv_bridge.c:194 [inline] vidtv_start_feed+0x33e/0x4d0 drivers/media/test-drivers/vidtv/vidtv_bridge.c:239	2026-04-24	5.5
CVE-2026-31590	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: KVM: SEV: Drop WARN on large size for KVM_MEMORY_ENCRYPT_REG_REGION Drop the WARN in sev_pin_memory() on npages overflowing an int, as the WARN is comically trivially to trigger from userspace, e.g. by doing: struct kvm_enc_region range = { .addr = 0, .size = -1ul, }; __vm_ioctl(vm, KVM_MEMORY_ENCRYPT_REG_REGION, &range); Note, the checks in sev_mem_enc_register_region() that presumably exist to verify the incoming address+size are completely worthless, as both "addr" and "size" are u64s and SEV is 64-bit only, i.e. they _can't_ be greater than ULONG_MAX. That wart will be cleaned up in the near future. if (range->addr > ULONG_MAX || range->size > ULONG_MAX) return -EINVAL; Opportunistically add a comment to explain why the code calculates the number of pages the "hard" way, e.g. instead of just shifting @ulen.	2026-04-24	5.5
CVE-2026-31591	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: KVM: SEV: Lock all vCPUs when synchronzing VMSAs for SNP launch finish Lock all vCPUs when synchronizing and encrypting VMSAs for SNP guests, as allowing userspace to manipulate and/or run a vCPU while its state is being synchronized would at best corrupt vCPU state, and at worst crash the host kernel. Opportunistically assert that vcpu->mutex is held when synchronizing its VMSA (the SEV-ES path already locks vCPUs).	2026-04-24	5.5
CVE-2026-31592	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: KVM: SEV: Protect *all* of sev_mem_enc_register_region() with kvm->lock	2026-04-24	5.5

		Take and hold kvm->lock for before checking sev_guest() in sev_mem_enc_register_region(), as sev_guest() isn't stable unless kvm->lock is held (or KVM can guarantee KVM_SEV_INIT{2} has completed and can't rollack state). If KVM_SEV_INIT{2} fails, KVM can end up trying to add to a not-yet-initialized sev->regions_list, e.g. triggering a #GP Oops: general protection fault, probably for non-canonical address 0xdfffffc0000000000: 0000 [#1] SMP KASAN NOPTI KASAN: null-ptr-deref in range [0x0000000000000000-0x0000000000000007] CPU: 110 UID: 0 PID: 72717 Comm: syz.15.11462 Tainted: G U W O 6.16.0-smp-DEV #1 NONE Tainted: [U]=USER, [W]=WARN, [O]=OOT_MODULE Hardware name: Google, Inc. Arcadia_IT_80/Arcadia_IT_80, BIOS 12.52.0-0 10/28/2024 RIP: 0010:sev_mem_enc_register_region+0x3f0/0x4f0 ../include/linux/list.h:83 Code: <41> 80 3c 04 00 74 08 4c 89 ff e8 f1 c7 a2 00 49 39 ed 0f 84 c6 00 RSP: 0018:ffff88838647fbb8 EFLAGS: 00010256 RAX: dfffffc0000000000 RBX: 1ffff92015cf1e0b RCX: dfffffc0000000000 RDX: 0000000000000000 RSI: 0000000000001000 RDI: ffff888367870000 RBP: ffffc900ae78f050 R08: ffffea000d9e0007 R09: 1ffffd4001b3c000 R10: dfffffc0000000000 R11: fffff94001b3c001 R12: 0000000000000000 R13: ffff8982ab0bde00 R14: ffffc900ae78f058 R15: 0000000000000000 FS: 00007f34e9dc66c0(0000) GS:ffff89ee64d33000(0000) knlGS:0000000000000000 CS: 0010 DS: 0000 ES: 0000 CR0: 0000000080050033 CR2: 00007fe180adef98 CR3: 000000047210e000 CR4: 0000000000350ef0 Call Trace: <TASK> kvm_arch_vm_ioctl+0xa72/0x1240 ../arch/x86/kvm/x86.c:7371 kvm_vm_ioctl+0x649/0x990 ../virt/kvm/kvm_main.c:5363 __se_sys_ioctl+0x101/0x170 ../fs/ioctl.c:51 do_syscall_x64 ../arch/x86/entry/syscall_64.c:63 [inline] do_syscall_64+0x6f/0x1f0 ../arch/x86/entry/syscall_64.c:94 entry_SYSCALL_64_after_hwframe+0x76/0x7e RIP: 0033:0x7f34e9f7e9a9 Code: <48> 3d 01 f0 ff ff 73 01 c3 48 c7 c1 a8 ff ff ff f7 d8 64 89 01 48 RSP: 002b:00007f34e9dc6038 EFLAGS: 00000246 ORIG_RAX: 0000000000000010 RAX: ffffffffffffffda RBX: 00007f34ea1a6080 RCX: 00007f34e9f7e9a9 RDX: 0000200000000280 RSI: 000000008010aebb RDI: 0000000000000007 RBP: 00007f34ea000d69 R08: 0000000000000000 R09: 0000000000000000 R10: 0000000000000000 R11: 0000000000000246 R12: 0000000000000000 R13: 0000000000000000 R14: 00007f34ea1a6080 R15: 00007ffce77197a8 </TASK> with a syzlang reproducer that looks like: syz_kvm_add_vcpu$x86(0x0, &(0x7f0000000040)={0x0, &(0x7f0000000180)=ANY=[], 0x70}) (async) syz_kvm_add_vcpu$x86(0x0, &(0x7f0000000080)={0x0, &(0x7f0000000180)=ANY=[@ANYBLOB="..."], 0x4f}) (async) r0 = openat$kvm(0xffffffffffffff9c, &(0x7f0000000200), 0x0, 0x0) r1 = ioctl$KVM_CREATE_VM(r0, 0xae01, 0x0) r2 = openat$kvm(0xffffffffffffff9c, &(0x7f0000000240), 0x0, 0x0) r3 = ioctl$KVM_CREATE_VM(r2, 0xae01, 0x0) ioctl$KVM_SET_CLOCK(r3, 0xc008aeba, &(0x7f0000000040)={0x1, 0x8, 0x0, 0x5625e9b0}) (async) ioctl$KVM_SET_PIT2(r3, 0x8010aebb, &(0x7f0000000280)={[...], 0x5}) (async) ioctl$KVM_SET_PIT2(r1, 0x4070aea0, 0x0) (async) r4 = ioctl$KVM_CREATE_VM(0xffffffffffffffff, 0xae01, 0x0) openat$kvm(0xffffffffffffff9c, 0x0, 0x0, 0x0) (async) ioctl$KVM_SET_USER_MEMORY_REGION(r4, 0x4020ae46, &(0x7f0000000400)={0x0, 0x0, 0x0, 0x2000, &(0x7f0000001000/0x2000)=nil}) (async) r5 = ioctl$KVM_CREATE_VCPU(r4, 0xae41, 0x2) close(r0) (async) openat$kvm(0xffffffffffffff9c, &(0x7f0000000000), 0x8000, 0x0) (async) ioctl$KVM_SET_GUEST_DEBUG(r5, 0x4048ae9b, &(0x7f0000000300)={0x4376ea830d46549b, 0x0, [0x46, 0x0, 0x0, 0x0, 0x0, 0x1000]}) (async) ioctl$KVM_RUN(r5, 0xae80, 0x0) Opportunistically use guard() to avoid having to define a new error label and goto usage.		
CVE-2026-31593	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: KVM: SEV: Reject attempts to sync VMSA of an already-launched/encrypted vCPU Reject synchronizing vCPU state to its associated VMSA if the vCPU has already been launched, i.e. if the VMSA has already been encrypted. On a host with SNP enabled, accessing guest-private memory generates an RMP #PF and panics the host. BUG: unable to handle page fault for address: ff1276cbfdf36000 #PF: supervisor write access in kernel mode #PF: error_code(0x80000003) - RMP violation	2026-04-24	5.5

		PGD 5a31801067 P4D 5a31802067 PUD 40ccfb5063 PMD 40e5954063 PTE 80000040fdf36163 SEV-SNP: PFN 0x40fdf36, RMP entry: [0x6010ffffffffff001 - 0x000000000000001f] Oops: Oops: 0003 [#1] SMP NOPTI CPU: 33 UID: 0 PID: 996180 Comm: qemu-system-x86 Tainted: G OE Tainted: [O]=OOT_MODULE, [E]=UNSIGNED_MODULE Hardware name: Dell Inc. PowerEdge R7625/0H1TJT, BIOS 1.5.8 07/21/2023 RIP: 0010:sev_es_sync_vmsa+0x54/0x4c0 [kvm_amd] Call Trace: <TASK> snp_launch_update_vmsa+0x19d/0x290 [kvm_amd] snp_launch_finish+0xb6/0x380 [kvm_amd] sev_mem_enc_ioctl+0x14e/0x720 [kvm_amd] kvm_arch_vm_ioctl+0x837/0xcf0 [kvm] kvm_vm_ioctl+0x3fd/0xcc0 [kvm] __x64_sys_ioctl+0xa3/0x100 x64_sys_call+0xfe0/0x2350 do_syscall_64+0x81/0x10f0 entry_SYSCALL_64_after_hwframe+0x76/0x7e RIP: 0033:0x7ffff673287d </TASK> Note, the KVM flaw has been present since commit ad73109ae7ec ("KVM: SVM: Provide support to launch and run an SEV-ES guest"), but has only been actively dangerous for the host since SNP support was added. With SEV-ES, KVM would "just" clobber guest state, which is totally fine from a host kernel perspective since userspace can clobber guest state any time before sev_launch_update_vmsa().		
CVE-2026-31594	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: PCI: endpoint: pci-epf-vntb: Remove duplicate resource teardown epf_ntb_epc_destroy() duplicates the teardown that the caller is supposed to perform later. This leads to an oops when .allow_link fails or when .drop_link is performed. The following is an example oops of the former case: Unable to handle kernel paging request at virtual address dead000000000108 [...] [dead000000000108] address between user and kernel address ranges Internal error: Oops: 0000000096000044 [#1] SMP [...] Call trace: pci_epc_remove_epf+0x78/0xe0 (P) pci_primary_epc_epf_link+0x88/0xa8 configfs_symlink+0x1f4/0x5a0 vfs_symlink+0x134/0x1d8 do_symlinkat+0x88/0x138 __arm64_sys_symlinkat+0x74/0xe0 [...] Remove the helper, and drop pci_epc_put(). EPC device refcounting is tied to the configfs EPC group lifetime, and pci_epc_put() in the .drop_link path is sufficient.	2026-04-24	5.5
CVE-2026-31595	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: PCI: endpoint: pci-epf-vntb: Stop cmd_handler work in epf_ntb_epc_cleanup Disable the delayed work before clearing BAR mappings and doorbells to avoid running the handler after resources have been torn down. Unable to handle kernel paging request at virtual address ffff800083f46004 [...] Internal error: Oops: 0000000096000007 [#1] SMP [...] Call trace: epf_ntb_cmd_handler+0x54/0x200 [pci_epf_vntb] (P) process_one_work+0x154/0x3b0 worker_thread+0x2c8/0x400 kthread+0x148/0x210 ret_from_fork+0x10/0x20	2026-04-24	5.5
CVE-2026-31596	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ocfs2: handle invalid dinode in ocfs2_group_extend [BUG] kernel BUG at fs/ocfs2/resize.c:308! Oops: invalid opcode: 0000 [#1] SMP KASAN NOPTI RIP: 0010:ocfs2_group_extend+0x10aa/0x1ae0 fs/ocfs2/resize.c:308 Code: 8b8520ff ffff83f8 860f8580 030000e8 5cc3c1fe Call Trace:	2026-04-24	5.5

		... ocfs2_ioctl+0x175/0x6e0 fs/ocfs2/ioctl.c:869 vfs_ioctl fs/ioctl.c:51 [inline] __do_sys_ioctl fs/ioctl.c:597 [inline] __se_sys_ioctl fs/ioctl.c:583 [inline] __x64_sys_ioctl+0x197/0x1e0 fs/ioctl.c:583 x64_sys_call+0x1144/0x26a0 arch/x86/include/generated/asm/syscalls_64.h:17 do_syscall_x64 arch/x86/entry/syscall_64.c:63 [inline] do_syscall_64+0x93/0xf80 arch/x86/entry/syscall_64.c:94 entry_SYSCALL_64_after_hwframe+0x76/0x7e ... [CAUSE] ocfs2_group_extend() assumes that the global bitmap inode block returned from ocfs2_inode_lock() has already been validated and BUG_ONs when the signature is not a dinode. That assumption is too strong for crafted filesystems because the JBD2-managed buffer path can bypass structural validation and return an invalid dinode to the resize ioctl. [FIX] Validate the dinode explicitly in ocfs2_group_extend(). If the global bitmap buffer does not contain a valid dinode, report filesystem corruption with ocfs2_error() and fail the resize operation instead of crashing the kernel.		
CVE-2026-31599	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: media: vidtv: fix NULL pointer dereference in vidtv_channel_pmt_match_sections syzbot reported a general protection fault in vidtv_psi_desc_assign [1]. vidtv_psi_pmt_stream_init() can return NULL on memory allocation failure, but vidtv_channel_pmt_match_sections() does not check for this. When tail is NULL, the subsequent call to vidtv_psi_desc_assign(&tail->descriptor, desc) dereferences a NULL pointer offset, causing a general protection fault. Add a NULL check after vidtv_psi_pmt_stream_init(). On failure, clean up the already-allocated stream chain and return. [1] Oops: general protection fault, probably for non-canonical address 0xdfffffc0000000000: 0000 [#1] SMP KASAN PTI KASAN: null-ptr-deref in range [0x0000000000000000-0x0000000000000007] RIP: 0010:vidtv_psi_desc_assign+0x24/0x90 drivers/media/test-drivers/vidtv/vidtv_psi.c:629 Call Trace: <TASK> vidtv_channel_pmt_match_sections drivers/media/test-drivers/vidtv/vidtv_channel.c:349 [inline] vidtv_channel_si_init+0x1445/0x1a50 drivers/media/test-drivers/vidtv/vidtv_channel.c:479 vidtv_mux_init+0x526/0xbe0 drivers/media/test-drivers/vidtv/vidtv_mux.c:519 vidtv_start_streaming drivers/media/test-drivers/vidtv/vidtv_bridge.c:194 [inline] vidtv_start_feed+0x33e/0x4d0 drivers/media/test-drivers/vidtv/vidtv_bridge.c:239	2026-04-24	5.5
CVE-2026-31601	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: vfio/xe: Reorganize the init to decouple migration from reset Attempting to issue reset on VF devices that don't support migration leads to the following: BUG: unable to handle page fault for address: 00000000000011f8 #PF: supervisor read access in kernel mode #PF: error_code(0x0000) - not-present page PGD 0 P4D 0 Oops: Oops: 0000 [#1] SMP NOPTI CPU: 2 UID: 0 PID: 7443 Comm: xe_sriov_flr Tainted: G S U 7.0.0-rc1-lgci-xe-xe-4588-cec43d5c2696af219-nodebug+ #1 PREEMPT(lazy) Tainted: [S]=CPU_OUT_OF_SPEC, [U]=USER Hardware name: Intel Corporation Alder Lake Client Platform/AlderLake-P DDR4 RVP, BIOS RPLPFWI1.R00.4035.A00.2301200723 01/20/2023 RIP: 0010:xe_sriov_vfio_wait_flr_done+0xc/0x80 [xe] Code: ff c3 cc cc cc cc 0f 1f 84 00 00 00 00 00 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 0f 1f 44 00 00 55 48 89 e5 41 54 53 <83> bf f8 11 00 00 02 75 61 41 89 f4 85 f6 74 52 48 8b 47 08 48 89 RSP: 0018:ffffc9000f7c39b8 EFLAGS: 00010202 RAX: ffffffffa04d8660 RBX: ffff88813e3e4000 RCX: 0000000000000000 RDX: 0000000000000000 RSI: 0000000000000000 RDI: 0000000000000000 RBP: ffffc9000f7c39c8 R08: 0000000000000000 R09: 0000000000000000 R10: 0000000000000000 R11: 0000000000000000 R12: ffff888101a48800 R13: ffff88813e3e4150 R14: ffff888130d0d008 R15: ffff88813e3e40d0 FS: 00007877d3d0d940(0000) GS:ffff88890b6d3000(0000) knlGS:0000000000000000 CS: 0010 DS: 0000 ES: 0000 CR0: 0000000080050033	2026-04-24	5.5

		CR2: 00000000000011f8 CR3: 000000015a762000 CR4: 0000000000f52ef0 PKRU: 55555554 Call Trace: <TASK> xe_vfio_pci_reset_done+0x49/0x120 [xe_vfio_pci] pci_dev_restore+0x3b/0x80 pci_reset_function+0x109/0x140 reset_store+0x5c/0xb0 dev_attr_store+0x17/0x40 sysfs_kf_write+0x72/0x90 kernfs_fop_write_iter+0x161/0x1f0 vfs_write+0x261/0x440 ksys_write+0x69/0xf0 __x64_sys_write+0x19/0x30 x64_sys_call+0x259/0x26e0 do_syscall_64+0xcb/0x1500 ? __fput+0x1a2/0x2d0 ? fput_close_sync+0x3d/0xa0 ? __x64_sys_close+0x3e/0x90 ? x64_sys_call+0x1b7c/0x26e0 ? do_syscall_64+0x109/0x1500 ? __task_pid_nr_ns+0x68/0x100 ? __do_sys_getpid+0x1d/0x30 ? x64_sys_call+0x10b5/0x26e0 ? do_syscall_64+0x109/0x1500 ? putname+0x41/0x90 ? do_faccessat+0x1e8/0x300 ? __x64_sys_access+0x1c/0x30 ? x64_sys_call+0x1822/0x26e0 ? do_syscall_64+0x109/0x1500 ? tick_program_event+0x43/0xa0 ? hrtimer_interrupt+0x126/0x260 ? irqentry_exit+0xb2/0x710 entry_SYSCALL_64_after_hwframe+0x76/0x7e RIP: 0033:0x7877d5f1c5a4 Code: c7 00 16 00 00 00 b8 ff ff ff ff c3 66 2e 0f 1f 84 00 00 00 00 00 f3 0f 1e fa 80 3d a5 ea 0e 00 00 74 13 b8 01 00 00 00 0f 05 <48> 3d 00 f0 ff ff 77 54 c3 0f 1f 00 55 48 89 e5 48 83 ec 20 48 89 RSP: 002b:00007fff48e5f908 EFLAGS: 00000202 ORIG_RAX: 0000000000000001 RAX: ffffffffffffffda RBX: 0000000000000000 RCX: 00007877d5f1c5a4 RDX: 0000000000000001 RSI: 00007877d621b0c9 RDI: 0000000000000009 RBP: 0000000000000001 R08: 00005fb49113b010 R09: 0000000000000007 R10: 0000000000000000 R11: 0000000000000202 R12: 00007877d621b0c9 R13: 0000000000000009 R14: 00007fff48e5fac0 R15: 00007fff48e5fac0 </TASK> This is caused by the fact that some of the xe_vfio_pci_core_device members needed for handling reset are only initialized as part of migration init. Fix the problem by reorganizing the code to decouple VF init from migration init.		
CVE-2026-31603	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: staging: sm750fb: fix division by zero in ps_to_hz() ps_to_hz() is called from hw_sm750_crtc_set_mode() without validating that pixclock is non-zero. A zero pixclock passed via FBIOPUT_VSCREENINFO causes a division by zero. Fix by rejecting zero pixclock in lynxfb_ops_check_var(), consistent with other framebuffer drivers.	2026-04-24	5.5
CVE-2026-31604	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: wifi: rtw88: fix device leak on probe failure Driver core holds a reference to the USB interface and its parent USB device while the interface is bound to a driver and there is no need to take additional references unless the structures are needed after disconnect. This driver takes a reference to the USB device during probe but does not to release it on all probe errors (e.g. when descriptor parsing fails). Drop the redundant device reference to fix the leak, reduce cargo culting, make it easier to spot drivers where an extra reference is needed, and reduce the risk of further memory leaks.	2026-04-24	5.5
CVE-2026-31605	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: fbdev: udlfb: avoid divide-by-zero on FBIOPUT_VSCREENINFO	2026-04-24	5.5

		Much like commit 19f953e74356 ("fbdev: fb_pm2fb: Avoid potential divide by zero error"), we also need to prevent that same crash from happening in the udlfb driver as it uses pixclock directly when dividing, which will crash.		
CVE-2026-31606	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: usb: gadget: f_hid: don't call cdev_init while cdev in use When calling unbind, then bind again, cdev_init reinitialized the cdev, even though there may still be references to it. That's the case when the /dev/hidg* device is still opened. This obviously unsafe behavior like oopes. This fixes this by using cdev_alloc to put the cdev on the heap. That way, we can simply allocate a new one in hidg_bind.	2026-04-24	5.5
CVE-2026-31610	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ksmbd: fix mechToken leak when SPNEGO decode fails after token alloc The kernel ASN.1 BER decoder calls action callbacks incrementally as it walks the input. When ksmbd_decode_negTokenInit() reaches the mechToken [2] OCTET STRING element, ksmbd_neg_token_alloc() allocates conn->mechToken immediately via kmemdup_nul(). If a later element in the same blob is malformed, then the decoder will return nonzero after the allocation is already live. This could happen if mechListMIC [3] overrunse the enclosing SEQUENCE. decode_negotiation_token() then sets conn->use_spnego = false because both the negTokenInit and negTokenTarg grammars failed. The cleanup at the bottom of smb2_sess_setup() is gated on use_spnego: if (conn->use_spnego && conn->mechToken) { kfree(conn->mechToken); conn->mechToken = NULL; } so the kfree is skipped, causing the mechToken to never be freed. This codepath is reachable pre-authentication, so untrusted clients can cause slow memory leaks on a server without even being properly authenticated. Fix this up by not checking check for use_spnego, as it's not required, so the memory will always be properly freed. At the same time, always free the memory in ksmbd_conn_free() incase some other failure path forgot to free it.	2026-04-24	5.5
CVE-2026-31615	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: usb: gadget: renesas_usb3: validate endpoint index in standard request handlers The GET_STATUS and SET/CLEAR_FEATURE handlers extract the endpoint number from the host-supplied wIndex without any sort of validation. Fix this up by validating the number of endpoints actually match up with the number the device has before attempting to dereference a pointer based on this math. This is just like what was done in commit ee0d382feb44 ("usb: gadget: aspeed_udc: validate endpoint index for ast udc") for the aspeed driver.	2026-04-24	5.5
CVE-2026-31616	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: usb: gadget: f_phonet: fix skb frags[] overflow in pn_rx_complete() A broken/bored/mean USB host can overflow the skb_shared_info->frags[] array on a Linux gadget exposing a Phonet function by sending an unbounded sequence of full-page OUT transfers. pn_rx_complete() finalizes the skb only when req->actual < req->length, where req->length is set to PAGE_SIZE by the gadget. If the host always sends exactly PAGE_SIZE bytes per transfer, fp->rx.skb will never be reset and each completion will add another fragment via skb_add_rx_frag(). Once nr_frags exceeds MAX_SKB_FRAGS (default 17), subsequent frag stores overwrite memory adjacent to the shinfo on the heap. Drop the skb and account a length error when the frag limit is reached, matching the fix applied in t7xx by commit f0813bcd2d9d ("net: wwan: t7xx: fix potential skb->frags overflow in RX path").	2026-04-24	5.5
CVE-2026-31617	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved:	2026-04-24	5.5

		usb: gadget: f_ncm: validate minimum block_len in ncm_unwrap_ntb() The block_len read from the host-supplied NTB header is checked against ntb_max but has no lower bound. When block_len is smaller than opts->ndp_size, the bounds check of: ndp_index > (block_len - opts->ndp_size) will underflow producing a huge unsigned value that ndp_index can never exceed, defeating the check entirely. The same underflow occurs in the datagram index checks against block_len - opts->dpe_size. With those checks neutered, a malicious USB host can choose ndp_index and datagram offsets that point past the actual transfer, and the skb_put_data() copies adjacent kernel memory into the network skb. Fix this by rejecting block lengths that cannot hold at least the NTB header plus one NDP. This will make block_len - opts->ndp_size and block_len - opts->dpe_size both well-defined. Commit 8d2b1a1ec9f5 ("CDC-NCM: avoid overflow in sanity checking") fixed a related class of issues on the host side of NCM.		
CVE-2026-31618	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: fbdev: tdfxfb: avoid divide-by-zero on FBIOPUT_VSCREENINFO Much like commit 19f953e74356 ("fbdev: fb_pm2fb: Avoid potential divide by zero error"), we also need to prevent that same crash from happening in the udlfb driver as it uses pixclock directly when dividing, which will crash.	2026-04-24	5.5
CVE-2026-31619	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ALSA: fireworks: bound device-supplied status before string array lookup The status field in an EFW response is a 32-bit value supplied by the firewire device. efr_status_names[] has 17 entries so a status value outside that range goes off into the weeds when looking at the %s value. Even worse, the status could return EFR_STATUS_INCOMPLETE which is 0x80000000, and is obviously not in that array of potential strings. Fix this up by properly bounding the index against the array size and printing "unknown" if it's not recognized.	2026-04-24	5.5
CVE-2026-31621	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: bnge: return after auxiliary_device_uninit() in error path When auxiliary_device_add() fails, the error block calls auxiliary_device_uninit() but does not return. The uninit drops the last reference and synchronously runs bnge_aux_dev_release(), which sets bd->auxr_dev = NULL and frees the underlying object. The subsequent bd->auxr_dev->net = bd->netdev then dereferences NULL, which is not a good thing to have happen when trying to clean up from an error. Add the missing return, as the auxiliary bus documentation states is a requirement (seems that LLM tools read documentation better than humans do...)	2026-04-24	5.5
CVE-2026-31623	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net: usb: cdc-phonet: fix skb frags[] overflow in rx_complete() A malicious USB device claiming to be a CDC Phonet modem can overflow the skb_shared_info->frags[] array by sending an unbounded sequence of full-page bulk transfers. Drop the skb and increment the length error when the frag limit is reached. This matches the same fix that commit f0813bcd2d9d ("net: wwan: t7xx: fix potential skb->frags overflow in RX path") did for the t7xx driver.	2026-04-24	5.5
CVE-2026-31624	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: HID: core: clamp report_size in s32ton() to avoid undefined shift s32ton() shifts by n-1 where n is the field's report_size, a value that comes directly from a HID device. The HID parser bounds report_size only to <= 256, so a broken HID device can supply a report descriptor with a wide field that triggers shift exponents up to 256 on a 32-bit type when an output report is built via hid_output_field() or hid_set_field(). Commit ec61b41918587 ("HID: core: fix shift-out-of-bounds in	2026-04-24	5.5

		hid_report_raw_event") added the same n > 32 clamp to the function snto32(), but s32ton() was never given the same fix as I guess syzbot hadn't figured out how to fuzz a device the same way. Fix this up by just clamping the max value of n, just like snto32() does.		
CVE-2026-31625	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: HID: alps: fix NULL pointer dereference in alps_raw_event() Commit ecfa6f34492c ("HID: Add HID_CLAIMED_INPUT guards in raw_event callbacks missing them") attempted to fix up the HID drivers that had missed the previous fix that was done in 2ff5baa9b527 ("HID: appleir: Fix potential NULL dereference at raw event handle"), but the alps driver was missed. Fix this up by properly checking in the hid-alps driver that it had been claimed correctly before attempting to process the raw event.	2026-04-24	5.5
CVE-2026-31628	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: x86/CPU: Fix FPDSS on Zen1 Zen1's hardware divider can leave, under certain circumstances, partial results from previous operations. Those results can be leaked by another, attacker thread. Fix that with a chicken bit.	2026-04-24	5.5
CVE-2026-31632	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: rxrpc: Fix leak of rxgk context in rxgk_verify_response() Fix rxgk_verify_response() to clean up the rxgk context it creates.	2026-04-24	5.5
CVE-2026-31634	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: rxrpc: fix reference count leak in rxrpc_server_keyring() This patch fixes a reference count leak in rxrpc_server_keyring() by checking if rx->securities is already set.	2026-04-24	5.5
CVE-2026-31639	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: rxrpc: Fix key reference count leak from call->key When creating a client call in rxrpc_alloc_client_call(), the code obtains a reference to the key. This is never cleaned up and gets leaked when the call is destroyed. Fix this by freeing call->key in rxrpc_destroy_call(). Before the patch, it shows the key reference counter elevated: $ cat /proc/keys | grep afs@54321 1bffe9cd I--Q--i 8053480 4169w 3b010000 1000 1000 rxrpc afs@54321: ka $ After the patch, the invalidated key is removed when the code exits: $ cat /proc/keys | grep afs@54321 $	2026-04-24	5.5
CVE-2026-31642	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: rxrpc: Fix call removal to use RCU safe deletion Fix rxrpc call removal from the rxnet->calls list to use list_del_rcu() rather than list_del_init() to prevent stuffing up reading /proc/net/rxrpc/calls from potentially getting into an infinite loop. This, however, means that list_empty() no longer works on an entry that's been deleted from the list, making it harder to detect prior deletion. Fix this by: Firstly, make rxrpc_destroy_all_calls() only dump the first ten calls that are unexpectedly still on the list. Limiting the number of steps means there's no need to call cond_resched() or to remove calls from the list here, thereby eliminating the need for rxrpc_put_call() to check for that. rxrpc_put_call() can then be fixed to unconditionally delete the call from the list as it is the only place that the deletion occurs.	2026-04-24	5.5
CVE-2026-31643	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: rxrpc: Fix key parsing memleak	2026-04-24	5.5

		In rxrpc_preparse_xdr_yfs_rxgk(), the memory attached to token->rxgk can be leaked in a few error paths after it's allocated. Fix this by freeing it in the "reject_token:" case.		
CVE-2026-31645	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net: lan966x: fix page pool leak in error paths lan966x_fdma_rx_alloc() creates a page pool but does not destroy it if the subsequent fdma_alloc_coherent() call fails, leaking the pool. Similarly, lan966x_fdma_init() frees the coherent DMA memory when lan966x_fdma_tx_alloc() fails but does not destroy the page pool that was successfully created by lan966x_fdma_rx_alloc(), leaking it. Add the missing page_pool_destroy() calls in both error paths.	2026-04-24	5.5
CVE-2026-31646	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net: lan966x: fix page_pool error handling in lan966x_fdma_rx_alloc_page_pool() page_pool_create() can return an ERR_PTR on failure. The return value is used unconditionally in the loop that follows, passing the error pointer through xdp_rxq_info_reg_mem_model() into page_pool_use_xdp_mem(), which dereferences it, causing a kernel oops. Add an IS_ERR check after page_pool_create() to return early on failure.	2026-04-24	5.5
CVE-2026-31647	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: idpf: fix PREEMPT_RT raw/bh spinlock nesting for async VC handling Switch from using the completion's raw spinlock to a local lock in the idpf_vc_xn struct. The conversion is safe because complete/_all() are called outside the lock and there is no reason to share the completion lock in the current logic. This avoids invalid wait context reported by the kernel due to the async handler taking BH spinlock: [805.726977] ============================ [805.726991] [BUG: Invalid wait context] [805.727006] 7.0.0-rc2-net-devq-031026+ #28 Tainted: G S OE [805.727026] ---------- [805.727038] kworker/u261:0/572 is trying to lock: [805.727051] ff190da6a8dbb6a0 (&vport_config->mac_filter_list_lock){+...}-{3:3}, at: idpf_mac_filter_async_handler+0xe9/0x260 [idpf] [805.727099] other info that might help us debug this: [805.727111] context-{5:5} [805.727119] 3 locks held by kworker/u261:0/572: [805.727132] #0: ff190da6db3e6148 ((wq_completion)idpf-0000:83:00.0-mbx){+.+.}-{0:0}, at: process_one_work+0x4b5/0x730 [805.727163] #1: ff3c6f0a6131fe50 ((work_completion)(&(&adapter->mbx_task)->work)){+.+.}-{0:0}, at: process_one_work+0x1e5/0x730 [805.727191] #2: ff190da765190020 (&x->wait#34){+.+.}-{2:2}, at: idpf_recv_mb_msg+0xc8/0x710 [idpf] [805.727218] stack backtrace: ... [805.727238] Workqueue: idpf-0000:83:00.0-mbx idpf_mbx_task [idpf] [805.727247] Call Trace: [805.727249] <TASK> [805.727251] dump_stack_lvl+0x77/0xb0 [805.727259] __lock_acquire+0xb3b/0x2290 [805.727268] ? __irq_work_queue_local+0x59/0x130 [805.727275] lock_acquire+0xc6/0x2f0 [805.727277] ? idpf_mac_filter_async_handler+0xe9/0x260 [idpf] [805.727284] ? _printk+0x5b/0x80 [805.727290] _raw_spin_lock_bh+0x38/0x50 [805.727298] ? idpf_mac_filter_async_handler+0xe9/0x260 [idpf] [805.727303] idpf_mac_filter_async_handler+0xe9/0x260 [idpf] [805.727310] idpf_recv_mb_msg+0x1c8/0x710 [idpf] [805.727317] process_one_work+0x226/0x730 [805.727322] worker_thread+0x19e/0x340 [805.727325] ? __pfx_worker_thread+0x10/0x10 [805.727328] kthread+0xf4/0x130 [805.727333] ? __pfx_kthread+0x10/0x10 [805.727336] ret_from_fork+0x32c/0x410 [805.727345] ? __pfx_kthread+0x10/0x10 [805.727347] ret_from_fork_asm+0x1a/0x30 [805.727354] </TASK>	2026-04-24	5.5
CVE-2026-31651	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: mmc: vub300: fix NULL-deref on disconnect	2026-04-24	5.5

		Make sure to deregister the controller before dropping the reference to the driver data on disconnect to avoid NULL-pointer dereferences or use-after-free.		
CVE-2026-31653	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: mm/damon/sysfs: dealloc repeat_call_control if damon_call() fails damon_call() for repeat_call_control of DAMON_SYSFS could fail if somehow the kdamond is stopped before the damon_call(). It could happen, for example, when te damon context was made for monitroing of a virtual address processes, and the process is terminated immediately, before the damon_call() invocation. In the case, the dyanmically allocated repeat_call_control is not deallocated and leaked. Fix the leak by deallocating the repeat_call_control under the damon_call() failure. This issue is discovered by sashiko [1].	2026-04-24	5.5
CVE-2026-31654	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: mm/vma: fix memory leak in __mmap_region() commit 605f6586ecf7 ("mm/vma: do not leak memory when .mmap_prepare swaps the file") handled the success path by skipping get_file() via file_doesnt_need_get, but missed the error path. When /dev/zero is mmap'd with MAP_SHARED, mmap_zero_prepare() calls shmem_zero_setup_desc() which allocates a new shmem file to back the mapping. If __mmap_new_vma() subsequently fails, this replacement file is never fput()'d - the original is released by ksys_mmap_pgoff(), but nobody releases the new one. Add fput() for the swapped file in the error path. Reproducible with fault injection. FAULT_INJECTION: forcing a failure. name failslab, interval 1, probability 0, space 0, times 1 CPU: 2 UID: 0 PID: 366 Comm: syz.7.14 Not tainted 7.0.0-rc6 #2 PREEMPT(full) Hardware name: QEMU Ubuntu 24.04 PC v2 (i440FX + PIIX, arch_caps fix, 1996), BIOS 1.16.3-debian-1.16.3-2 04/01/2014 Call Trace: <TASK> dump_stack_lvl+0x164/0x1f0 should_fail_ex+0x525/0x650 should_failslab+0xdf/0x140 kmem_cache_alloc_noprof+0x78/0x630 vm_area_alloc+0x24/0x160 __mmap_region+0xf6b/0x2660 mmap_region+0x2eb/0x3a0 do_mmap+0xc79/0x1240 vm_mmap_pgoff+0x252/0x4c0 ksys_mmap_pgoff+0xf8/0x120 __x64_sys_mmap+0x12a/0x190 do_syscall_64+0xa9/0x580 entry_SYSCALL_64_after_hwframe+0x76/0x7e </TASK> kmemleak: 1 new suspected memory leaks (see /sys/kernel/debug/kmemleak) BUG: memory leak unreferenced object 0xffff8881118aca80 (size 360): comm "syz.7.14", pid 366, jiffies 4294913255 hex dump (first 32 bytes): 00 00 00 00 ad 4e ad de ff ff ff ff 00 00 00 00 .....N..... ff ff ff ff ff ff ff ff c0 28 4d ae ff ff ff ff .....(M..... backtrace (crc db0f53bc): kmem_cache_alloc_noprof+0x3ab/0x630 alloc_empty_file+0x5a/0x1e0 alloc_file_pseudo+0x135/0x220 __shmem_file_setup+0x274/0x420 shmem_zero_setup_desc+0x9c/0x170 mmap_zero_prepare+0x123/0x140 __mmap_region+0xdda/0x2660 mmap_region+0x2eb/0x3a0 do_mmap+0xc79/0x1240 vm_mmap_pgoff+0x252/0x4c0 ksys_mmap_pgoff+0xf8/0x120 __x64_sys_mmap+0x12a/0x190 do_syscall_64+0xa9/0x580	2026-04-24	5.5

		entry_SYSCALL_64_after_hwframe+0x76/0x7e Found by syzkaller.		
CVE-2026-31655	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: pmdomain: imx8mp-blk-ctrl: Keep the NOC_HDCP clock enabled Keep the NOC_HDCP clock always enabled to fix the potential hang caused by the NoC ADB400 port power down handshake.	2026-04-24	5.5
CVE-2026-31658	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net: altera-tse: fix skb leak on DMA mapping error in tse_start_xmit() When dma_map_single() fails in tse_start_xmit(), the function returns NETDEV_TX_OK without freeing the skb. Since NETDEV_TX_OK tells the stack the packet was consumed, the skb is never freed, leaking memory on every DMA mapping failure. Add dev_kfree_skb_any() before returning to properly free the skb.	2026-04-24	5.5
CVE-2026-31660	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: nfc: pn533: allocate rx skb before consuming bytes pn532_receive_buf() reports the number of accepted bytes to the serdev core. The current code consumes bytes into recv_skb and may already hand a complete frame to pn533_recv_frame() before allocating a fresh receive buffer. If that alloc_skb() fails, the callback returns 0 even though it has already consumed bytes, and it leaves recv_skb as NULL for the next receive callback. That breaks the receive_buf() accounting contract and can also lead to a NULL dereference on the next skb_put_u8(). Allocate the receive skb lazily before consuming the next byte instead. If allocation fails, return the number of bytes already accepted.	2026-04-24	5.5
CVE-2026-31661	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: wifi: brcmsmac: Fix dma_free_coherent() size dma_alloc_consistent() may change the size to align it. The new size is saved in alloced. Change the free size to match the allocation size.	2026-04-24	5.5
CVE-2026-31664	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: xfrm: clear trailing padding in build_polexpire() build_expire() clears the trailing padding bytes of struct xfrm_user_expire after setting the hard field via memset_after(), but the analogous function build_polexpire() does not do this for struct xfrm_user_polexpire. The padding bytes after the __u8 hard field are left uninitialized from the heap allocation, and are then sent to userspace via netlink multicast to XFRMNLGRP_EXPIRE listeners, leaking kernel heap memory contents. Add the missing memset_after() call, matching build_expire().	2026-04-24	5.5
CVE-2026-31670	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net: rfkill: prevent unlimited numbers of rfkill events from being created Userspace can create an unlimited number of rfkill events if the system is so configured, while not consuming them from the rfkill file descriptor, causing a potential out of memory situation. Prevent this from bounding the number of pending rfkill events at a "large" number (i.e. 1000) to prevent abuses like this.	2026-04-24	5.5
CVE-2026-31671	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: xfrm_user: fix info leak in build_report() struct xfrm_user_report is a __u8 proto field followed by a struct xfrm_selector which means there is three "empty" bytes of padding, but the padding is never zeroed before copying to userspace. Fix that up by zeroing the structure before setting individual member variables.	2026-04-24	5.5
CVE-2026-31672	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: wifi: rt2x00usb: fix devres lifetime USB drivers bind to USB interfaces and any device managed resources	2026-04-24	5.5

		should have their lifetime tied to the interface rather than parent USB device. This avoids issues like memory leaks when drivers are unbound without their devices being physically disconnected (e.g. on probe deferral or configuration changes). Fix the USB anchor lifetime so that it is released on driver unbind.		
CVE-2026-6774	mozilla - multiple products	Mitigation bypass in the DOM: Security component. This vulnerability was fixed in Firefox 150 and Thunderbird 150.	2026-04-21	5.4
CVE-2026-22006	oracle - peoplesoft_enterprise_hcm_human_resources	Vulnerability in the PeopleSoft Enterprise HCM Human Resources product of Oracle PeopleSoft (component: Employee Snapshot). The supported version that is affected is 9.2. Easily exploitable vulnerability allows low privileged attacker with network access via HTTP to compromise PeopleSoft Enterprise HCM Human Resources. Successful attacks require human interaction from a person other than the attacker and while the vulnerability is in PeopleSoft Enterprise HCM Human Resources, attacks may significantly impact additional products (scope change). Successful attacks of this vulnerability can result in unauthorized update, insert or delete access to some of PeopleSoft Enterprise HCM Human Resources accessible data as well as unauthorized read access to a subset of PeopleSoft Enterprise HCM Human Resources accessible data. CVSS 3.1 Base Score 5.4 (Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:L/UI:R/S:C/C:L/I:L/A:N).	2026-04-21	5.4
CVE-2026-22019	oracle - peoplesoft_enterprise_hcm_shared_components	Vulnerability in the PeopleSoft Enterprise HCM Shared Components product of Oracle PeopleSoft (component: Person Search). The supported version that is affected is 9.2. Easily exploitable vulnerability allows low privileged attacker with network access via HTTP to compromise PeopleSoft Enterprise HCM Shared Components. Successful attacks require human interaction from a person other than the attacker and while the vulnerability is in PeopleSoft Enterprise HCM Shared Components, attacks may significantly impact additional products (scope change). Successful attacks of this vulnerability can result in unauthorized update, insert or delete access to some of PeopleSoft Enterprise HCM Shared Components accessible data as well as unauthorized read access to a subset of PeopleSoft Enterprise HCM Shared Components accessible data. CVSS 3.1 Base Score 5.4 (Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:L/UI:R/S:C/C:L/I:L/A:N).	2026-04-21	5.4
CVE-2026-34307	oracle - multiple products	Vulnerability in the PeopleSoft Enterprise PeopleTools product of Oracle PeopleSoft (component: Workflow). Supported versions that are affected are 8.61-8.62. Easily exploitable vulnerability allows low privileged attacker with network access via HTTP to compromise PeopleSoft Enterprise PeopleTools. Successful attacks require human interaction from a person other than the attacker and while the vulnerability is in PeopleSoft Enterprise PeopleTools, attacks may significantly impact additional products (scope change). Successful attacks of this vulnerability can result in unauthorized update, insert or delete access to some of PeopleSoft Enterprise PeopleTools accessible data as well as unauthorized read access to a subset of PeopleSoft Enterprise PeopleTools accessible data. CVSS 3.1 Base Score 5.4 (Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:L/UI:R/S:C/C:L/I:L/A:N).	2026-04-21	5.4
CVE-2026-35232	oracle - multiple products	Vulnerability in Oracle Fusion Middleware (component: Dynamic Monitoring Service). Supported versions that are affected are 12.2.1.4.0 and 14.1.2.0.0. Easily exploitable vulnerability allows low privileged attacker with network access via HTTP to compromise Oracle Fusion Middleware. Successful attacks require human interaction from a person other than the attacker and while the vulnerability is in Oracle Fusion Middleware, attacks may significantly impact additional products (scope change). Successful attacks of this vulnerability can result in unauthorized update, insert or delete access to some of Oracle Fusion Middleware accessible data as well as unauthorized read access to a subset of Oracle Fusion Middleware accessible data. CVSS 3.1 Base Score 5.4 (Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:L/UI:R/S:C/C:L/I:L/A:N).	2026-04-21	5.4
CVE-2026-6848	red hat - multiple products	A flaw was found in Red Hat Quay. When Red Hat Quay requests password re-verification for sensitive operations, such as token generation or robot account creation, the re-authentication prompt can be bypassed. This allows a user with a timed-out session, or an attacker with access to an idle authenticated browser session, to perform privileged actions without providing valid credentials. The vulnerability enables unauthorized execution of sensitive operations despite the user interface displaying an error for invalid credentials.	2026-04-22	5.4
CVE-2025-66335	apache - doris_mcp_server	Apache Doris MCP Server versions earlier than 0.6.1 are affected by an improper neutralization flaw in query context handling that may allow execution of unintended SQL statements and bypass of intended query validation and access restrictions through the MCP query execution interface. Version 0.6.1 and later are not affected.	2026-04-20	5.3
CVE-2026-33558	apache - multiple products	Information exposure vulnerability has been identified in Apache Kafka. The NetworkClient component will output entire requests and responses information in the DEBUG log level in the logs. By default, the log level is set to INFO level. If the DEBUG level is enabled, the sensitive information will be exposed via the requests and responses output log. The entire lists of impacted requests and responses are: * AlterConfigsRequest * AlterUserScramCredentialsRequest * ExpireDelegationTokenRequest * IncrementalAlterConfigsRequest * RenewDelegationTokenRequest * SaslAuthenticateRequest	2026-04-20	5.3

		* createDelegationTokenResponse * describeDelegationTokenResponse * SaslAuthenticateResponse This issue affects Apache Kafka: from any version supported the listed API above through v3.9.1, v4.0.0. We advise the Kafka users to upgrade to v3.9.2, v4.0.1, or later to avoid this vulnerability.		
CVE-2026-6765	mozilla - multiple products	Information disclosure in the Form Autofill component. This vulnerability was fixed in Firefox 150, Firefox ESR 140.10, Thunderbird 150, and Thunderbird 140.10.	2026-04-21	5.3
CVE-2026-6767	mozilla - multiple products	Other issue in the Libraries component in NSS. This vulnerability was fixed in Firefox 150, Firefox ESR 115.35, Firefox ESR 140.10, Thunderbird 150, and Thunderbird 140.10.	2026-04-21	5.3
CVE-2026-6775	mozilla - multiple products	Incorrect boundary conditions in the WebRTC component. This vulnerability was fixed in Firefox 150 and Thunderbird 150.	2026-04-21	5.3
CVE-2026-6777	mozilla - multiple products	Other issue in the Networking: DNS component. This vulnerability was fixed in Firefox 150 and Thunderbird 150.	2026-04-21	5.3
CVE-2026-6778	mozilla - multiple products	Invalid pointer in the Audio/Video: Playback component. This vulnerability was fixed in Firefox 150 and Thunderbird 150.	2026-04-21	5.3
CVE-2026-6779	mozilla - multiple products	Other issue in the JavaScript Engine component. This vulnerability was fixed in Firefox 150 and Thunderbird 150.	2026-04-21	5.3
CVE-2026-6783	mozilla - multiple products	Incorrect boundary conditions, integer overflow in the Audio/Video: Playback component. This vulnerability was fixed in Firefox 150 and Thunderbird 150.	2026-04-21	5.3
CVE-2026-22013	oracle - multiple products	Vulnerability in the Oracle Java SE, Oracle GraalVM for JDK, Oracle GraalVM Enterprise Edition product of Oracle Java SE (component: JGSS). Supported versions that are affected are Oracle Java SE: 8u481, 8u481-b50, 8u481-perf, 11.0.30, 17.0.18, 21.0.10, 25.0.2, 26; Oracle GraalVM for JDK: 17.0.18 and 21.0.10; Oracle GraalVM Enterprise Edition: 21.3.17. Difficult to exploit vulnerability allows unauthenticated attacker with network access via multiple protocols to compromise Oracle Java SE, Oracle GraalVM for JDK, Oracle GraalVM Enterprise Edition. Successful attacks require human interaction from a person other than the attacker. Successful attacks of this vulnerability can result in unauthorized access to critical data or complete access to all Oracle Java SE, Oracle GraalVM for JDK, Oracle GraalVM Enterprise Edition accessible data. Note: This vulnerability applies to Java deployments, typically in clients running sandboxed Java Web Start applications or sandboxed Java applets, that load and run untrusted code (e.g., code that comes from the internet) and rely on the Java sandbox for security. This vulnerability does not apply to Java deployments, typically in servers, that load and run only trusted code (e.g., code installed by an administrator). CVSS 3.1 Base Score 5.3 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:H/PR:N/UI:R/S:U/C:H/I:N/A:N).	2026-04-21	5.3
CVE-2026-22021	oracle - multiple products	Vulnerability in the Oracle Java SE, Oracle GraalVM for JDK, Oracle GraalVM Enterprise Edition product of Oracle Java SE (component: JSSE). Supported versions that are affected are Oracle Java SE: 8u481, 8u481-b50, 8u481-perf, 11.0.30, 17.0.18, 21.0.10, 25.0.2, 26; Oracle GraalVM for JDK: 17.0.18 and 21.0.10; Oracle GraalVM Enterprise Edition: 21.3.17. Easily exploitable vulnerability allows unauthenticated attacker with network access via HTTPS to compromise Oracle Java SE, Oracle GraalVM for JDK, Oracle GraalVM Enterprise Edition. Successful attacks of this vulnerability can result in unauthorized ability to cause a partial denial of service (partial DOS) of Oracle Java SE, Oracle GraalVM for JDK, Oracle GraalVM Enterprise Edition. Note: This vulnerability can be exploited by using APIs in the specified Component, e.g., through a web service which supplies data to the APIs. This vulnerability also applies to Java deployments, typically in clients running sandboxed Java Web Start applications or sandboxed Java applets, that load and run untrusted code (e.g., code that comes from the internet) and rely on the Java sandbox for security. CVSS 3.1 Base Score 5.3 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:L).	2026-04-21	5.3
CVE-2026-34273	oracle - goldengate	Vulnerability in Oracle GoldenGate (component: Libraries). Supported versions that are affected are 23.4-23.10. Easily exploitable vulnerability allows unauthenticated attacker with network access via HTTP to compromise Oracle GoldenGate. Successful attacks of this vulnerability can result in unauthorized read access to a subset of Oracle GoldenGate accessible data. CVSS 3.1 Base Score 5.3 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N).	2026-04-21	5.3
CVE-2026-22748	vmware - multiple products	Vulnerability in Spring Spring Security. When an application configures JWT decoding with NimbusJwtDecoder or NimbusReactiveJwtDecoder, it must configure an OAuth2TokenValidator<Jwt> separately, for example by calling setJwtValidator.This issue affects Spring Security: from 6.3.0 through 6.3.14, from 6.4.0 through 6.4.14, from 6.5.0 through 6.5.9, from 7.0.0 through 7.0.4.	2026-04-22	5.3
CVE-2026-40448	samsung - one	Potential Integer overflow in tensor allocation size calculation could lead to insufficient memory allocation for large tensors in Samsung Open Source ONE. Affected version is prior to commit 1.30.0.	2026-04-22	5.3
CVE-2026-6654	mozilla - thin-vec	Double-Free / Use-After-Free (UAF) in the `IntoIter::drop` and `ThinVec::clear` functions in the thin_vec crate. A panic in `ptr::drop_in_place` skips setting the length to zero.	2026-04-20	5.1
CVE-2026-35248	oracle - vm_virtualbox	Vulnerability in the Oracle VM VirtualBox product of Oracle Virtualization (component: Core). The supported version that is affected is 7.2.6. Difficult to exploit vulnerability allows high privileged attacker with logon to the infrastructure where Oracle VM VirtualBox executes to compromise Oracle VM VirtualBox. While the vulnerability is in Oracle VM VirtualBox, attacks may significantly impact additional products (scope change). Successful attacks of this vulnerability can result in unauthorized update, insert or delete access to some of Oracle VM VirtualBox accessible data as well as unauthorized read access to a subset of Oracle VM VirtualBox accessible data and unauthorized ability to cause a partial denial of service (partial DOS) of Oracle VM VirtualBox. CVSS 3.1 Base Score 5.0 (Confidentiality, Integrity and Availability impacts). CVSS Vector: (CVSS:3.1/AV:L/AC:H/PR:H/UI:N/S:C/C:L/I:L/A:L).	2026-04-21	5
CVE-2026-6845	red hat - multiple products	A flaw was found in binutils, specifically within the `readelf` utility. This vulnerability allows a local attacker to cause a Denial of Service (DoS) by tricking a user into processing a specially crafted	2026-04-22	5

		Executable and Linkable Format (ELF) file. The exploitation of this flaw can lead to the system becoming unresponsive due to excessive resource consumption or a program crash.		
CVE-2026-21998	oracle - multiple products	Vulnerability in the MySQL Server product of Oracle MySQL (component: Server: Optimizer). Supported versions that are affected are 8.0.0-8.0.45, 8.4.0-8.4.8 and 9.0.0-9.6.0. Easily exploitable vulnerability allows high privileged attacker with network access via multiple protocols to compromise MySQL Server. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of MySQL Server. CVSS 3.1 Base Score 4.9 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:N/A:H).	2026-04-21	4.9
CVE-2026-22002	oracle - multiple products	Vulnerability in the MySQL Server product of Oracle MySQL (component: Server: Optimizer). Supported versions that are affected are 8.0.0-8.0.45, 8.4.0-8.4.8 and 9.0.0-9.6.0. Easily exploitable vulnerability allows high privileged attacker with network access via multiple protocols to compromise MySQL Server. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of MySQL Server. CVSS 3.1 Base Score 4.9 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:N/A:H).	2026-04-21	4.9
CVE-2026-22004	oracle - multiple products	Vulnerability in the MySQL Server product of Oracle MySQL (component: InnoDB). Supported versions that are affected are 8.0.0-8.0.45, 8.4.0-8.4.8 and 9.0.0-9.6.0. Easily exploitable vulnerability allows high privileged attacker with network access via multiple protocols to compromise MySQL Server. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of MySQL Server. CVSS 3.1 Base Score 4.9 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:N/A:H).	2026-04-21	4.9
CVE-2026-22005	oracle - multiple products	Vulnerability in the MySQL Server product of Oracle MySQL (component: Server: Optimizer). Supported versions that are affected are 8.0.0-8.0.45, 8.4.0-8.4.8 and 9.0.0-9.6.0. Easily exploitable vulnerability allows high privileged attacker with network access via multiple protocols to compromise MySQL Server. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of MySQL Server. CVSS 3.1 Base Score 4.9 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:N/A:H).	2026-04-21	4.9
CVE-2026-34267	oracle - mysql_server	Vulnerability in the MySQL Server product of Oracle MySQL (component: Server: Optimizer). Supported versions that are affected are 8.0.0-8.0.45. Easily exploitable vulnerability allows high privileged attacker with network access via multiple protocols to compromise MySQL Server. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of MySQL Server. CVSS 3.1 Base Score 4.9 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:N/A:H).	2026-04-21	4.9
CVE-2026-34278	oracle - mysql_server	Vulnerability in the MySQL Server product of Oracle MySQL (component: Server: Optimizer). Supported versions that are affected are 8.0.0-8.0.45. Easily exploitable vulnerability allows high privileged attacker with network access via multiple protocols to compromise MySQL Server. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of MySQL Server. CVSS 3.1 Base Score 4.9 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:N/A:H).	2026-04-21	4.9
CVE-2026-34293	oracle - mysql_server	Vulnerability in the MySQL Server product of Oracle MySQL (component: Server: DML). Supported versions that are affected are 8.0.0-8.0.45. Easily exploitable vulnerability allows high privileged attacker with network access via multiple protocols to compromise MySQL Server. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of MySQL Server. CVSS 3.1 Base Score 4.9 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:N/A:H).	2026-04-21	4.9
CVE-2026-34304	oracle - multiple products	Vulnerability in the MySQL Server product of Oracle MySQL (component: InnoDB). Supported versions that are affected are 8.0.0-8.0.45, 8.4.0-8.4.8 and 9.0.0-9.6.0. Easily exploitable vulnerability allows high privileged attacker with network access via multiple protocols to compromise MySQL Server. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of MySQL Server. CVSS 3.1 Base Score 4.9 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:N/A:H).	2026-04-21	4.9
CVE-2026-35234	oracle - mysql_server	Vulnerability in the MySQL Server product of Oracle MySQL (component: Server: Partition). Supported versions that are affected are 9.0.0-9.6.0. Easily exploitable vulnerability allows high privileged attacker with network access via multiple protocols to compromise MySQL Server. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of MySQL Server. CVSS 3.1 Base Score 4.9 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:N/A:H).	2026-04-21	4.9
CVE-2026-35235	oracle - mysql_server	Vulnerability in the MySQL Server product of Oracle MySQL (component: Server: GIS). Supported versions that are affected are 9.0.0-9.6.0. Easily exploitable vulnerability allows high privileged attacker with network access via multiple protocols to compromise MySQL Server. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of MySQL Server. CVSS 3.1 Base Score 4.9 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:N/A:H).	2026-04-21	4.9
CVE-2026-35236	oracle - multiple products	Vulnerability in the MySQL Server product of Oracle MySQL (component: InnoDB). Supported versions that are affected are 8.0.0-8.0.45, 8.4.0-8.4.8 and 9.0.0-9.6.0. Easily exploitable vulnerability allows high privileged attacker with network access via multiple protocols to compromise MySQL Server. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of MySQL Server. CVSS 3.1 Base Score 4.9 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:N/A:H).	2026-04-21	4.9
CVE-2026-35237	oracle - multiple products	Vulnerability in the MySQL Server product of Oracle MySQL (component: InnoDB). Supported versions that are affected are 8.0.0-8.0.45, 8.4.0-8.4.8 and 9.0.0-9.6.0. Easily exploitable vulnerability allows high privileged attacker with network access via multiple protocols to compromise MySQL Server. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of MySQL Server. CVSS 3.1 Base Score 4.9 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:N/A:H).	2026-04-21	4.9
CVE-2026-35238	oracle - multiple products	Vulnerability in the MySQL Server product of Oracle MySQL (component: InnoDB). Supported versions that are affected are 8.0.0-8.0.45, 8.4.0-8.4.8 and 9.0.0-9.6.0. Easily exploitable vulnerability allows high privileged attacker with network access via multiple protocols to compromise MySQL Server. Successful attacks of this vulnerability can result in unauthorized ability	2026-04-21	4.9

		to cause a hang or frequently repeatable crash (complete DOS) of MySQL Server. CVSS 3.1 Base Score 4.9 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:N/A:H).		
CVE-2026-35239	oracle - multiple products	Vulnerability in the MySQL Server product of Oracle MySQL (component: Server: DML). Supported versions that are affected are 8.0.0-8.0.45, 8.4.0-8.4.8 and 9.0.0-9.6.0. Easily exploitable vulnerability allows high privileged attacker with network access via multiple protocols to compromise MySQL Server. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of MySQL Server. CVSS 3.1 Base Score 4.9 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:N/A:H).	2026-04-21	4.9
CVE-2026-35240	oracle - multiple products	Vulnerability in the MySQL Server product of Oracle MySQL (component: Server: Optimizer). Supported versions that are affected are 8.0.0-8.0.45, 8.4.0-8.4.8 and 9.0.0-9.6.0. Easily exploitable vulnerability allows high privileged attacker with network access via multiple protocols to compromise MySQL Server. Successful attacks of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete DOS) of MySQL Server. CVSS 3.1 Base Score 4.9 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:N/A:H).	2026-04-21	4.9
CVE-2026-1274	ibm - multiple products	IBM Guardium Data Protection 12.0, 12.1, and 12.2 is vulnerable to a Bypass Business Logic vulnerability in the access management control panel.	2026-04-23	4.9
CVE-2026-4917	ibm - guardium_data_protection	IBM Guardium Data Protection 12.1 could allow an administrative user to traverse directories on the system. An attacker could send a specially crafted URL request containing "dot dot" sequences (/../) to write arbitrary files on the system.	2026-04-23	4.9
CVE-2026-22751	vmware - multiple products	Vulnerability in Spring Spring Security. Applications that explicitly configure One-Time Token login with JdbcOneTimeTokenService are vulnerable to a Time-of-check Time-of-use (TOCTOU) race condition. This issue affects Spring Security: from 6.4.0 through 6.4.15, from 6.5.0 through 6.5.9, from 7.0.0 through 7.0.4.	2026-04-21	4.8
CVE-2026-34321	oracle - multiple products	Vulnerability in the Oracle Financial Services Analytical Applications Infrastructure product of Oracle Financial Services Applications (component: User Interface). Supported versions that are affected are 8.0.7.9, 8.0.8.7 and 8.1.2.5. Difficult to exploit vulnerability allows low privileged attacker with network access via HTTP to compromise Oracle Financial Services Analytical Applications Infrastructure. Successful attacks require human interaction from a person other than the attacker. Successful attacks of this vulnerability can result in unauthorized access to critical data or complete access to all Oracle Financial Services Analytical Applications Infrastructure accessible data. CVSS 3.1 Base Score 4.8 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:H/PR:L/UI:R/S:U/C:H/I:N/A:N).	2026-04-21	4.8
CVE-2026-1726	ibm - multiple products	IBM Guardium Key Lifecycle Manager 4.1, 4.1.1, 4.2, 4.2.1, 5.0, and 5.1	2026-04-23	4.8
CVE-2026-4919	ibm - guardium_data_protection	IBM Guardium Data Protection 12.1 is vulnerable to cross-site scripting. This vulnerability allows an administrative user to embed arbitrary JavaScript code in the Web UI thus altering the intended functionality potentially leading to credentials disclosure within a trusted session.	2026-04-23	4.8
CVE-2026-34298	oracle - applications_framework	Vulnerability in the Oracle Applications Framework product of Oracle E-Business Suite (component: Personalization). Supported versions that are affected are 12.2.9-12.2.15. Easily exploitable vulnerability allows high privileged attacker with network access via HTTP to compromise Oracle Applications Framework. Successful attacks of this vulnerability can result in unauthorized update, insert or delete access to some of Oracle Applications Framework accessible data as well as unauthorized read access to a subset of Oracle Applications Framework accessible data and unauthorized ability to cause a partial denial of service (partial DOS) of Oracle Applications Framework. CVSS 3.1 Base Score 4.7 (Confidentiality, Integrity and Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:L/I:L/A:L).	2026-04-21	4.7
CVE-2026-31523	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: nvme-pci: ensure we're polling a polled queue A user can change the polled queue count at run time. There's a brief window during a reset where a hipri task may try to poll that queue before the block layer has updated the queue maps, which would race with the now interrupt driven queue and may cause double completions.	2026-04-22	4.7
CVE-2025-66286	red hat - multiple products	An API design flaw in WebKitGTK and WPE WebKit allows untrusted web content to unexpectedly perform IP connections, DNS lookups, and HTTP requests. Applications expect to use the WebPage::send-request signal handler to approve or reject all network requests. However, certain types of HTTP requests bypass this signal handler.	2026-04-23	4.7
CVE-2026-31535	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: smb: client: make use of smbdirect_socket.recv_io.credits.available The logic off managing recv credits by counting posted recv_io and granted credits is racy. That's because the peer might already consumed a credit, but between receiving the incoming recv at the hardware and processing the completion in the 'recv_done' functions we likely have a window where we grant credits, which don't really exist. So we better have a decicated counter for the available credits, which will be incremented when we posted new recv buffers and drained when we grant the credits to the peer.	2026-04-24	4.7
CVE-2026-31572	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: i2c: designware: amdisp: Fix resume-probe race condition issue	2026-04-24	4.7

		Identified resume-probe race condition in kernel v7.0 with the commit 38fa29b01a6a ("i2c: designware: Combine the init functions"),but this issue existed from the beginning though not detected. The amdisp i2c device requires ISP to be in power-on state for probe to succeed. To meet this requirement, this device is added to genpd to control ISP power using runtime PM. The pm_runtime_get_sync() called before i2c_dw_probe() triggers PM resume, which powers on ISP and also invokes the amdisp i2c runtime resume before the probe completes resulting in this race condition and a NULL dereferencing issue in v7.0 Fix this race condition by using the genpd APIs directly during probe: - Call dev_pm_genpd_resume() to Power ON ISP before probe - Call dev_pm_genpd_suspend() to Power OFF ISP after probe - Set the device to suspended state with pm_runtime_set_suspended() - Enable runtime PM only after the device is fully initialized		
CVE-2026-31620	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ALSA: usx2y: us144mkii: fix NULL deref on missing interface 0 A malicious USB device with the TASCAM US-144MKII device id can have a configuration containing bInterfaceNumber=1 but no interface 0. USB configuration descriptors are not required to assign interface numbers sequentially, so usb_ifnum_to_if(dev, 0) returns will NULL, which will then be dereferenced directly. Fix this up by checking the return value properly.	2026-04-24	4.6
CVE-2026-6058	zyxel - WRE6505 v2 firmware	** UNSUPPORTED WHEN ASSIGNED ** An improper encoding or escaping vulnerability in the CGI program of Zyxel WRE6505 v2 firmware version V1.00(ABDV.3)C0 could allow an adjacent attacker on the WLAN to cause a denial-of-service (DoS) condition in the web management interface by convincing an authenticated administrator to visit the “AP Select” page while a malformed SSID is present.	2026-04-21	4.5
CVE-2026-41285	openbsd - openbsd	In OpenBSD through 7.8, the slaacd and rad daemons have an infinite loop when they receive a crafted ICMPv6 Neighbor Discovery (ND) option (over a local network) with length zero, because of an "nd_opt_len * 8 - 2" expression with no preceding check for whether nd_opt_len is zero.	2026-04-21	4.3
CVE-2026-22015	oracle - multiple products	Vulnerability in the MySQL Server product of Oracle MySQL (component: Server: Information Schema). Supported versions that are affected are 8.0.0-8.0.45, 8.4.0-8.4.8 and 9.0.0-9.6.0. Easily exploitable vulnerability allows low privileged attacker with network access via multiple protocols to compromise MySQL Server. Successful attacks of this vulnerability can result in unauthorized read access to a subset of MySQL Server accessible data. CVSS 3.1 Base Score 4.3 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:L/I:N/A:N).	2026-04-21	4.3
CVE-2026-34296	oracle - agile_product_lifecycle_management_for_process	Vulnerability in the Oracle Agile Product Lifecycle Management for Process product of Oracle Supply Chain (component: Product Quality Management). The supported version that is affected is 6.2.4. Easily exploitable vulnerability allows low privileged attacker with network access via HTTP to compromise Oracle Agile Product Lifecycle Management for Process. Successful attacks of this vulnerability can result in unauthorized read access to a subset of Oracle Agile Product Lifecycle Management for Process accessible data. CVSS 3.1 Base Score 4.3 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:L/I:N/A:N).	2026-04-21	4.3
CVE-2026-38743	apache - airflow	The authenticated /ui/dags endpoint did not enforce per-DAG access control on embedded Human-in-the-Loop (HITL) and TaskInstance records: a logged-in Airflow user with read access to at least one DAG could retrieve HITL prompts (including their request parameters) and full TaskInstance details for DAGs outside their authorized scope. Because HITL prompts and TaskInstance fields routinely carry operator parameters and free-form context attached to a task, the leak widens visibility of DAG-run data beyond the intended per-DAG RBAC boundary for every authenticated user. Users are recommended to upgrade to version 3.2.1 , which fixes this issue.	2026-04-24	4.3
CVE-2026-40690	apache - airflow	The asset dependency graph did not restrict nodes by the viewer's DAG read permissions: a user with read access to at least one DAG could browse the asset graph for any other asset in the deployment and learn the existence and names of DAGs and assets outside their authorized scope. Users are recommended to upgrade to version 3.2.1, which fixes this issue.	2026-04-24	4.3
CVE-2026-22014	oracle - user_management	Vulnerability in the Oracle User Management product of Oracle E-Business Suite (component: Workflow and Business Events). Supported versions that are affected are 12.2.7-12.2.15. Easily exploitable vulnerability allows high privileged attacker with network access via HTTP to compromise Oracle User Management. Successful attacks of this vulnerability can result in unauthorized update, insert or delete access to some of Oracle User Management accessible data as well as unauthorized read access to a subset of Oracle User Management accessible data. CVSS 3.1 Base Score 3.8 (Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:L/I:L/A:N).	2026-04-21	3.8
CVE-2026-22008	oracle - multiple products	Vulnerability in Oracle Java SE (component: Libraries). The supported version that is affected is Oracle Java SE: 25.0.1. Difficult to exploit vulnerability allows unauthenticated attacker with network access via multiple protocols to compromise Oracle Java SE. Successful attacks of this vulnerability can result in unauthorized update, insert or delete access to some of Oracle Java SE accessible data. Note: This vulnerability applies to Java deployments, typically in clients running sandboxed Java Web Start applications or sandboxed Java applets, that load and run untrusted code (e.g., code that comes from the internet) and rely on the Java sandbox for security. This vulnerability does not apply to Java deployments, typically in servers, that load and run only trusted code (e.g., code installed by an administrator). CVSS 3.1 Base Score 3.7 (Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:U/C:N/I:L/A:N).	2026-04-21	3.7

CVE-2026-22018	oracle - multiple products	Vulnerability in the Oracle Java SE, Oracle GraalVM for JDK, Oracle GraalVM Enterprise Edition product of Oracle Java SE (component: Libraries). Supported versions that are affected are Oracle Java SE: 8u481, 8u481-b50, 8u481-perf, 11.0.30, 17.0.18, 21.0.10, 25.0.2, 26; Oracle GraalVM for JDK: 17.0.18 and 21.0.10; Oracle GraalVM Enterprise Edition: 21.3.17. Difficult to exploit vulnerability allows unauthenticated attacker with network access via multiple protocols to compromise Oracle Java SE, Oracle GraalVM for JDK, Oracle GraalVM Enterprise Edition. Successful attacks of this vulnerability can result in unauthorized ability to cause a partial denial of service (partial DOS) of Oracle Java SE, Oracle GraalVM for JDK, Oracle GraalVM Enterprise Edition. Note: This vulnerability can be exploited by using APIs in the specified Component, e.g., through a web service which supplies data to the APIs. This vulnerability also applies to Java deployments, typically in clients running sandboxed Java Web Start applications or sandboxed Java applets, that load and run untrusted code (e.g., code that comes from the internet) and rely on the Java sandbox for security. CVSS 3.1 Base Score 3.7 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:U/C:N/I:N/A:L).	2026-04-21	3.7
CVE-2026-22746	vmware - multiple products	Vulnerability in Spring Spring Security. If an application is using the UserDetails#isEnabled, #isAccountNonExpired, or #isAccountNonLocked user attributes, to enable, expire, or lock users, then DaoAuthenticationProvider's timing attack defense can be bypassed for users who are disabled, expired, or locked.This issue affects Spring Security: from 5.7.0 through 5.7.22, from 5.8.0 through 5.8.24, from 6.3.0 through 6.3.15, from 6.5.0 through 6.5.9, from 7.0.0 through 7.0.4.	2026-04-22	3.7
CVE-2026-2708	red hat - multiple products	A request smuggling vulnerability exists in libsoup's HTTP/1 header parsing logic. The soup_message_headers_append_common() function in libsoup/soup-message-headers.c unconditionally appends each header value without validating for duplicate or conflicting Content-Length fields. This allows an attacker to send HTTP requests containing multiple Content-Length headers with differing values.	2026-04-23	3.7
CVE-2026-35249	oracle - vm_virtualbox	Vulnerability in the Oracle VM VirtualBox product of Oracle Virtualization (component: Core). The supported version that is affected is 7.2.6. Easily exploitable vulnerability allows high privileged attacker with logon to the infrastructure where Oracle VM VirtualBox executes to compromise Oracle VM VirtualBox. While the vulnerability is in Oracle VM VirtualBox, attacks may significantly impact additional products (scope change). Successful attacks of this vulnerability can result in unauthorized update, insert or delete access to some of Oracle VM VirtualBox accessible data. CVSS 3.1 Base Score 3.2 (Integrity impacts). CVSS Vector: (CVSS:3.1/AV:L/AC:L/PR:H/UI:N/S:C/C:N/I:L/A:N).	2026-04-21	3.2
CVE-2026-22007	oracle - multiple products	Vulnerability in the Oracle Java SE, Oracle GraalVM for JDK, Oracle GraalVM Enterprise Edition product of Oracle Java SE (component: Security). Supported versions that are affected are Oracle Java SE: 8u481, 8u481-b50, 8u481-perf, 11.0.30, 17.0.18, 21.0.10, 25.0.2, 26; Oracle GraalVM for JDK: 17.0.18 and 21.0.10; Oracle GraalVM Enterprise Edition: 21.3.17. Difficult to exploit vulnerability allows unauthenticated attacker with logon to the infrastructure where Oracle Java SE, Oracle GraalVM for JDK, Oracle GraalVM Enterprise Edition executes to compromise Oracle Java SE, Oracle GraalVM for JDK, Oracle GraalVM Enterprise Edition. Successful attacks of this vulnerability can result in unauthorized read access to a subset of Oracle Java SE, Oracle GraalVM for JDK, Oracle GraalVM Enterprise Edition accessible data. Note: This vulnerability can be exploited by using APIs in the specified Component, e.g., through a web service which supplies data to the APIs. This vulnerability also applies to Java deployments, typically in clients running sandboxed Java Web Start applications or sandboxed Java applets, that load and run untrusted code (e.g., code that comes from the internet) and rely on the Java sandbox for security. CVSS 3.1 Base Score 2.9 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:L/AC:H/PR:N/UI:N/S:U/C:L/I:N/A:N).	2026-04-21	2.9
CVE-2026-34268	oracle - multiple products	Vulnerability in the Oracle Java SE, Oracle GraalVM for JDK, Oracle GraalVM Enterprise Edition product of Oracle Java SE (component: Security). Supported versions that are affected are Oracle Java SE: 8u481, 8u481-b50, 8u481-perf, 11.0.30, 17.0.18, 21.0.10, 25.0.2, 26; Oracle GraalVM for JDK: 17.0.18 and 21.0.10; Oracle GraalVM Enterprise Edition: 21.3.17. Difficult to exploit vulnerability allows unauthenticated attacker with logon to the infrastructure where Oracle Java SE, Oracle GraalVM for JDK, Oracle GraalVM Enterprise Edition executes to compromise Oracle Java SE, Oracle GraalVM for JDK, Oracle GraalVM Enterprise Edition. Successful attacks of this vulnerability can result in unauthorized read access to a subset of Oracle Java SE, Oracle GraalVM for JDK, Oracle GraalVM Enterprise Edition accessible data. Note: This vulnerability can be exploited by using APIs in the specified Component, e.g., through a web service which supplies data to the APIs. This vulnerability also applies to Java deployments, typically in clients running sandboxed Java Web Start applications or sandboxed Java applets, that load and run untrusted code (e.g., code that comes from the internet) and rely on the Java sandbox for security. CVSS 3.1 Base Score 2.9 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:L/AC:H/PR:N/UI:N/S:U/C:L/I:N/A:N).	2026-04-21	2.9
CVE-2026-22001	oracle - multiple products	Vulnerability in the MySQL Server product of Oracle MySQL (component: Server: Information Schema). Supported versions that are affected are 8.0.0-8.0.45, 8.4.0-8.4.8 and 9.0.0-9.6.0. Easily exploitable vulnerability allows high privileged attacker with network access via multiple protocols to compromise MySQL Server. Successful attacks of this vulnerability can result in unauthorized read access to a subset of MySQL Server accessible data. CVSS 3.1 Base Score 2.7 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:L/I:N/A:N).	2026-04-21	2.7
CVE-2026-1272	ibm - multiple products	IBM Guardium Data Protection 12.0, 12.1, and 12.2 is vulnerable to Security Misconfiguration vulnerability in the user access control panel.	2026-04-23	2.7
CVE-2026-6842	red hat - multiple products	A flaw was found in nano. In environments with permissive umask settings, a local attacker can exploit incorrect directory permissions (0777 instead of 0700) for the `~/.local` directory. This allows the attacker to inject a malicious `.desktop` launcher, which could lead to unintended actions or information disclosure if the launcher is subsequently processed.	2026-04-22	2.5
CVE-2026-34312	oracle - database_server	Vulnerability in the RDBMS component of Oracle Database Server. Supported versions that are affected are 19.3-19.30. Easily exploitable vulnerability allows high privileged attacker having Row Access Method privilege with network access via multiple protocols to compromise RDBMS. Successful attacks require human interaction from a person other than the attacker. Successful attacks of this vulnerability can result in unauthorized read access to a subset of RDBMS accessible data. CVSS 3.1 Base Score 2.4 (Confidentiality impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:H/UI:R/S:U/C:L/I:N/A:N).	2026-04-21	2.4

CVE-2026-22051	netapp - StorageGRID (formerly StorageGRID Webscale)	StorageGRID (formerly StorageGRID Webscale) versions prior to 11.9.0.13 and 12.0.0.6 are susceptible to a Information Disclosure vulnerability. Successful exploit could allow an authenticated attacker with low privileges to run arbitrary metrics queries, revealing metric results that they do not have access to.	2026-04-20	2.3
CVE-2026-35250	oracle - vm_virtualbox	Vulnerability in the Oracle VM VirtualBox product of Oracle Virtualization (component: Core). The supported version that is affected is 7.2.6. Easily exploitable vulnerability allows high privileged attacker with logon to the infrastructure where Oracle VM VirtualBox executes to compromise Oracle VM VirtualBox. Successful attacks of this vulnerability can result in unauthorized ability to cause a partial denial of service (partial DOS) of Oracle VM VirtualBox. CVSS 3.1 Base Score 2.3 (Availability impacts). CVSS Vector: (CVSS:3.1/AV:L/AC:L/PR:H/UI:N/S:U/C:N/I:N/A:L).	2026-04-21	2.3
CVE-2026-5958	gnu - Sed	When sed is invoked with both -i (in-place edit) and --follow-symlinks, the function open_next_file() performs two separate, non-atomic filesystem operations on the same path: 1. resolves symlink to its target and stores the resolved path for determining when output is written, 2. opens the original symlink path (not the resolved one) to read the file. Between these two calls there is a race window. If an attacker atomically replaces the symlink with a different target during that window, sed will: read content from the new (attacker-chosen) symlink target and write the processed result to the path recorded in step 1. This can lead to arbitrary file overwrite with attacker-controlled content in the context of the sed process. This issue was fixed in version 4.10.	2026-04-20	2.1