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 27 أبريل إلى 3 مايو. 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 ٢٧th of April to ٣rd of May. 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-2025-30390	microsoft - Azure Machine Learning	Improper authorization in Azure allows an authorized attacker to elevate privileges over a network.	2025-04-30	9.9
CVE-2025-31651	apache - multiple products	Improper Neutralization of Escape, Meta, or Control Sequences vulnerability in Apache Tomcat. For a subset of unlikely rewrite rule configurations, it was possible for a specially crafted request to bypass some rewrite rules. If those rewrite rules effectively enforced security constraints, those constraints could be bypassed. This issue affects Apache Tomcat: from 11.0.0-M1 through 11.0.5, from 10.1.0-M1 through 10.1.39, from 9.0.0.M1 through 9.0.102. Users are recommended to upgrade to version [FIXED_VERSION], which fixes the issue.	2025-04-28	9.8
CVE-2025-24252	apple - multiple products	A use-after-free issue was addressed with improved memory management. This issue is fixed in macOS Sequoia 15.4, tvOS 18.4, macOS Ventura 13.7.5, iPadOS 17.7.6, macOS Sonoma 14.7.5, iOS 18.4 and iPadOS 18.4, visionOS 2.4. An attacker on the local network may be able to corrupt process memory.	2025-04-29	9.8
CVE-2025-30392	microsoft - Azure AI Bot Service	Improper authorization in Azure Bot Framework SDK allows an unauthorized attacker to elevate privileges over a network.	2025-04-30	9.8
CVE-2025-4083	mozilla - multiple products	A process isolation vulnerability in Thunderbird stemmed from improper handling of javascript: URIs, which could allow content to execute in the top-level document's process instead of the intended frame, potentially enabling a sandbox escape. This vulnerability affects Firefox < 138, Firefox ESR < 128.10, Firefox ESR < 115.23, Thunderbird < 138, and Thunderbird < 128.10.	2025-04-29	9.1
CVE-2025-2817	mozilla - multiple products	Thunderbird's update mechanism allowed a medium-integrity user process to interfere with the SYSTEM-level updater by manipulating the file-locking behavior. By injecting code into the user-privileged process, an attacker could bypass intended access controls, allowing SYSTEM-level file operations on paths controlled by a non-privileged user and enabling privilege escalation. This vulnerability affects Firefox < 138, Firefox ESR < 128.10, Firefox ESR < 115.23, Thunderbird < 138, and Thunderbird < 128.10.	2025-04-29	8.8
CVE-2025-4114	netgear - JWNR2000v2	A vulnerability classified as critical has been found in Netgear JWNR2000v2 1.0.0.11. Affected is the function check_language_file. The manipulation of the argument host leads to buffer overflow. It is possible to launch the attack remotely. The vendor was contacted early about this disclosure but did not respond in any way.	2025-04-30	8.7
CVE-2025-4115	netgear - JWNR2000v2	A vulnerability classified as critical was found in Netgear JWNR2000v2 1.0.0.11. Affected by this vulnerability is the function default_version_is_new. The manipulation of the argument host leads to buffer overflow. The attack can be launched remotely. The vendor was contacted early about this disclosure but did not respond in any way.	2025-04-30	8.7
CVE-2025-4116	netgear - JWNR2000v2	A vulnerability, which was classified as critical, has been found in Netgear JWNR2000v2 1.0.0.11. Affected by this issue is the function get_cur_lang_ver. The manipulation of the argument host leads to buffer overflow. The attack may be launched remotely. The vendor was contacted early about this disclosure but did not respond in any way.	2025-04-30	8.7
CVE-2025-4120	netgear - JWNR2000v2	A vulnerability was found in Netgear JWNR2000v2 1.0.0.11. It has been classified as critical. Affected is the function sub_4238E8. The manipulation of the argument host leads to buffer overflow. It is possible to launch the attack remotely. The vendor was contacted early about this disclosure but did not respond in any way.	2025-04-30	8.7

CVE-2025-30389	microsoft - Azure AI Bot Service	Improper authorization in Azure Bot Framework SDK allows an unauthorized attacker to elevate privileges over a network.	2025-04-30	8.7
CVE-2025-4139	netgear - EX6120	A vulnerability classified as critical was found in Netgear EX6120 1.0.0.68. Affected by this vulnerability is the function fwAcosCgiInbound. The manipulation of the argument host leads to buffer overflow. The attack can be launched remotely. The vendor was contacted early about this disclosure but did not respond in any way.	2025-04-30	8.7
CVE-2025-4140	netgear - EX6120	A vulnerability, which was classified as critical, has been found in Netgear EX6120 1.0.3.94. Affected by this issue is the function sub_30394. The manipulation of the argument host leads to buffer overflow. The attack may be launched remotely. The vendor was contacted early about this disclosure but did not respond in any way.	2025-04-30	8.7
CVE-2025-4141	netgear - EX6200	A vulnerability, which was classified as critical, was found in Netgear EX6200 1.0.3.94. This affects the function sub_3C03C. The manipulation of the argument host leads to buffer overflow. It is possible to initiate the attack remotely. The vendor was contacted early about this disclosure but did not respond in any way.	2025-04-30	8.7
CVE-2025-4142	netgear - EX6200	A vulnerability has been found in Netgear EX6200 1.0.3.94 and classified as critical. This vulnerability affects the function sub_3C8EC. The manipulation of the argument host leads to buffer overflow. The attack can be initiated remotely. The vendor was contacted early about this disclosure but did not respond in any way.	2025-04-30	8.7
CVE-2025-4145	netgear - EX6200	A vulnerability, which was classified as critical, has been found in Netgear EX6200 1.0.3.94. This issue affects the function sub_3D0BC. The manipulation of the argument host leads to buffer overflow. The attack may be initiated remotely. The vendor was contacted early about this disclosure but did not respond in any way.	2025-05-01	8.7
CVE-2025-4146	netgear - EX6200	A vulnerability, which was classified as critical, was found in Netgear EX6200 1.0.3.94. Affected is the function sub_41940. The manipulation of the argument host leads to buffer overflow. It is possible to launch the attack remotely. The vendor was contacted early about this disclosure but did not respond in any way.	2025-05-01	8.7
CVE-2025-4147	netgear - EX6200	A vulnerability has been found in Netgear EX6200 1.0.3.94 and classified as critical. Affected by this vulnerability is the function sub_47F7C. The manipulation of the argument host leads to buffer overflow. The attack can be launched remotely. The vendor was contacted early about this disclosure but did not respond in any way.	2025-05-01	8.7
CVE-2025-4148	netgear - EX6200	A vulnerability was found in Netgear EX6200 1.0.3.94 and classified as critical. Affected by this issue is the function sub_503FC. The manipulation of the argument host leads to buffer overflow. The attack may be launched remotely. The vendor was contacted early about this disclosure but did not respond in any way.	2025-05-01	8.7
CVE-2025-4149	netgear - EX6200	A vulnerability was found in Netgear EX6200 1.0.3.94. It has been classified as critical. This affects the function sub_54014. The manipulation of the argument host leads to buffer overflow. It is possible to initiate the attack remotely. The vendor was contacted early about this disclosure but did not respond in any way.	2025-05-01	8.7
CVE-2025-4150	netgear - EX6200	A vulnerability was found in Netgear EX6200 1.0.3.94. It has been declared as critical. This vulnerability affects the function sub_54340. The manipulation of the argument host leads to buffer overflow. The attack can be initiated remotely. The vendor was contacted early about this disclosure but did not respond in any way.	2025-05-01	8.7
CVE-2025-21416	microsoft - Azure Virtual Desktop	Missing authorization in Azure Virtual Desktop allows an authorized attacker to elevate privileges over a network.	2025-04-30	8.5
CVE-2025-3501	red hat - multiple products	A flaw was found in Keycloak. By setting a verification policy to 'ALL', the trust store certificate verification is skipped, which is unintended.	2025-04-29	8.2
CVE-2025-30391	microsoft - Dynamics 365 Customer Service	Improper input validation in Microsoft Dynamics allows an unauthorized attacker to disclose information over a network.	2025-04-30	8.1
CVE-2025-23375	dell - PowerProtect Data Manager	Dell PowerProtect Data Manager Reporting, version(s) 19.17, contain(s) an Incorrect Use of Privileged APIs vulnerability. A low privileged attacker with local access could potentially exploit this vulnerability, leading to Elevation of privileges.	2025-04-28	7.8
CVE-2022-49840	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: bpf, test_run: Fix alignment problem in bpf_prog_test_run_skb() We got a syzkaller problem because of aarch64 alignment fault if KFENCE enabled. When the size from user bpf program is an odd number, like 399, 407, etc, it will cause the struct skb_shared_info's unaligned access. As seen below: BUG: KFENCE: use-after-free read in __skb_clone+0x23c/0x2a0 net/core/skbuff.c:1032 Use-after-free read at 0xffff6254fffac077 (in kfence-#213): __lse_atomic_add arch/arm64/include/asm/atomic_lse.h:26 [inline] arch_atomic_add arch/arm64/include/asm/atomic.h:28 [inline] arch_atomic_inc include/linux/atomic-arch-fallback.h:270 [inline] atomic_inc include/asm-generic/atomic-instrumented.h:241 [inline] __skb_clone+0x23c/0x2a0 net/core/skbuff.c:1032 skb_clone+0xf4/0x214 net/core/skbuff.c:1481 ____bpf_clone_redirect net/core/filter.c:2433 [inline] bpf_clone_redirect+0x78/0x1c0 net/core/filter.c:2420 bpf_prog_d3839dd9068ceb51+0x80/0x330 bpf_dispatcher_nop_func include/linux/bpf.h:728 [inline] bpf_test_run+0x3c0/0x6c0 net/bpf/test_run.c:53 bpf_prog_test_run_skb+0x638/0xa7c net/bpf/test_run.c:594 bpf_prog_test_run kernel/bpf/syscall.c:3148 [inline] __do_sys_bpf kernel/bpf/syscall.c:4441 [inline] __se_sys_bpf+0xad0/0x1634 kernel/bpf/syscall.c:4381 kfence-#213: 0xffff6254fffac000-0xffff6254fffac196, size=407, cache=kmalloc-512	2025-05-01	7.8

		allocated by task 15074 on cpu 0 at 1342.585390s: kmalloc include/linux/slab.h:568 [inline] kzalloc include/linux/slab.h:675 [inline] bpf_test_init.isra.0+0xac/0x290 net/bpf/test_run.c:191 bpf_prog_test_run_skb+0x11c/0xa7c net/bpf/test_run.c:512 bpf_prog_test_run kernel/bpf/syscall.c:3148 [inline] __do_sys_bpf kernel/bpf/syscall.c:4441 [inline] __se_sys_bpf+0xad0/0x1634 kernel/bpf/syscall.c:4381 __arm64_sys_bpf+0x50/0x60 kernel/bpf/syscall.c:4381 To fix the problem, we adjust @size so that (@size + @hearoom) is a multiple of SMP_CACHE_BYTES. So we make sure the struct skb_shared_info is aligned to a cache line.		

CVE-2022-49842	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved:
		
		ASoC: core: Fix use-after-free in snd_soc_exit()
		
		KASAN reports a use-after-free:
		
		BUG: KASAN: use-after-free in device_del+0xb5b/0xc60
		Read of size 8 at addr ffff888008655050 by task rmmod/387
		CPU: 2 PID: 387 Comm: rmmod
		Hardware name: QEMU Standard PC (i440FX + PIIX, 1996)
		Call Trace:
		<TASK>
		dump_stack_lvl+0x79/0x9a
		print_report+0x17f/0x47b
		kasan_report+0xbb/0xf0
		device_del+0xb5b/0xc60
		platform_device_del.part.0+0x24/0x200
		platform_device_unregister+0x2e/0x40
		snd_soc_exit+0xa/0x22 [snd_soc_core]
		__do_sys_delete_module.constprop.0+0x34f/0x5b0
		do_syscall_64+0x3a/0x90
		entry_SYSCALL_64_after_hwframe+0x63/0xcd
		...
		</TASK>
		
		It's bacause in snd_soc_init(), snd_soc_util_init() is possble to fail,
		but its ret is ignored, which makes soc_dummy_dev unregistered twice.
		
		snd_soc_init()
		snd_soc_util_init()
		platform_device_register_simple(soc_dummy_dev)
		platform_driver_register() # fail
		platform_device_unregister(soc_dummy_dev)
		platform_driver_register() # success
		...
		snd_soc_exit()
		snd_soc_util_exit()
		# soc_dummy_dev will be unregistered for second time
		
		To fix it, handle error and stop snd_soc_init() when util_init() fail.
		Also clean debugfs when util_init() or driver_register() fail.
			2025-05-01	7.8

CVE-2022-49846	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: udf: Fix a slab-out-of-bounds write bug in udf_find_entry() Syzbot reported a slab-out-of-bounds Write bug: loop0: detected capacity change from 0 to 2048 ========== BUG: KASAN: slab-out-of-bounds in udf_find_entry+0x8a5/0x14f0 fs/udf/namei.c:253 Write of size 105 at addr ffff8880123ff896 by task syz-executor323/3610 CPU: 0 PID: 3610 Comm: syz-executor323 Not tainted 6.1.0-rc2-syzkaller-00105-gb229b6ca5abb #0 Hardware name: Google Compute Engine/Google Compute Engine, BIOS Google 10/11/2022 Call Trace: <TASK> __dump_stack lib/dump_stack.c:88 [inline] dump_stack_lvl+0x1b1/0x28e lib/dump_stack.c:106 print_address_description+0x74/0x340 mm/kasan/report.c:284 print_report+0x107/0x1f0 mm/kasan/report.c:395 kasan_report+0xcd/0x100 mm/kasan/report.c:495 kasan_check_range+0x2a7/0x2e0 mm/kasan/generic.c:189 memcpy+0x3c/0x60 mm/kasan/shadow.c:66 udf_find_entry+0x8a5/0x14f0 fs/udf/namei.c:253	2025-05-01	7.8

		udf_lookup+0xef/0x340 fs/udf/namei.c:309 lookup_open fs/namei.c:3391 [inline] open_last_lookups fs/namei.c:3481 [inline] path_openat+0x10e6/0x2df0 fs/namei.c:3710 do_filp_open+0x264/0x4f0 fs/namei.c:3740 do_sys_openat2+0x124/0x4e0 fs/open.c:1310 do_sys_open fs/open.c:1326 [inline] __do_sys_creat fs/open.c:1402 [inline] __se_sys_creat fs/open.c:1396 [inline] __x64_sys_creat+0x11f/0x160 fs/open.c:1396 do_syscall_x64 arch/x86/entry/common.c:50 [inline] do_syscall_64+0x3d/0xb0 arch/x86/entry/common.c:80 entry_SYSCALL_64_after_hwframe+0x63/0xcd RIP: 0033:0x7ffab0d164d9 Code: ff ff c3 66 2e 0f 1f 84 00 00 00 00 00 0f 1f 40 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 c0 ff ff ff f7 d8 64 89 01 48 RSP: 002b:00007ffe1a7e6bb8 EFLAGS: 00000246 ORIG_RAX: 0000000000000055 RAX: ffffffffffffffda RBX: 0000000000000000 RCX: 00007ffab0d164d9 RDX: 00007ffab0d164d9 RSI: 0000000000000000 RDI: 0000000020000180 RBP: 00007ffab0cd5a10 R08: 0000000000000000 R09: 0000000000000000 R10: 00005555573552c0 R11: 0000000000000246 R12: 00007ffab0cd5aa0 R13: 0000000000000000 R14: 0000000000000000 R15: 0000000000000000 </TASK> Allocated by task 3610: kasan_save_stack mm/kasan/common.c:45 [inline] kasan_set_track+0x3d/0x60 mm/kasan/common.c:52 ____kasan_kmalloc mm/kasan/common.c:371 [inline] __kasan_kmalloc+0x97/0xb0 mm/kasan/common.c:380 kmalloc include/linux/slab.h:576 [inline] udf_find_entry+0x7b6/0x14f0 fs/udf/namei.c:243 udf_lookup+0xef/0x340 fs/udf/namei.c:309 lookup_open fs/namei.c:3391 [inline] open_last_lookups fs/namei.c:3481 [inline] path_openat+0x10e6/0x2df0 fs/namei.c:3710 do_filp_open+0x264/0x4f0 fs/namei.c:3740 do_sys_openat2+0x124/0x4e0 fs/open.c:1310 do_sys_open fs/open.c:1326 [inline] __do_sys_creat fs/open.c:1402 [inline] __se_sys_creat fs/open.c:1396 [inline] __x64_sys_creat+0x11f/0x160 fs/open.c:1396 do_syscall_x64 arch/x86/entry/common.c:50 [inline] do_syscall_64+0x3d/0xb0 arch/x86/entry/common.c:80 entry_SYSCALL_64_after_hwframe+0x63/0xcd The buggy address belongs to the object at ffff8880123ff800 which belongs to the cache kmalloc-256 of size 256 The buggy address is located 150 bytes inside of 256-byte region [ffff8880123ff800, ffff8880123ff900) The buggy address belongs to the physical page: page:ffffea000048ff80 refcount:1 mapcount:0 mapping:0000000000000000 index:0x0 pfn:0x123fe head:ffffea000048ff80 order:1 compound_mapcount:0 compound_pincount:0 flags: 0xfff00000010200(slab|head|node=0|zone=1|lastcpupid=0x7ff) raw: 00fff00000010200 ffffea00004b8500 dead000000000003 ffff888012041b40 raw: 0000000000000000 0000000080100010 00000001ffffffff 0000000000000000 page dumped because: kasan: bad access detected page_owner tracks the page as allocated page last allocated via order 0, migratetype Unmovable, gfp_mask 0x0(), pid 1, tgid 1 (swapper/0), ts 1841222404, free_ts 0 create_dummy_stack mm/page_owner.c: ---truncated---		
CVE-2022-49888	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: arm64: entry: avoid kprobe recursion The cortex_a76_erratum_1463225_debug_handler() function is called when handling debug exceptions (and synchronous exceptions from BRK instructions), and so is called when a probed function executes. If the compiler does not inline cortex_a76_erratum_1463225_debug_handler(), it can be probed. If cortex_a76_erratum_1463225_debug_handler() is probed, any debug exception or software breakpoint exception will result in recursive exceptions leading to a stack overflow. This can be triggered with the ftrace multiple_probes selftest, and as per the example splat below. This is a regression caused by commit:	2025-05-01	7.8

		6459b8469753e9fe ("arm64: entry: consolidate Cortex-A76 erratum 1463225 workaround") ... which removed the NOKPROBE_SYMBOL() annotation associated with the function. My intent was that cortex_a76_erratum_1463225_debug_handler() would be inlined into its caller, el1_dbg(), which is marked noinstr and cannot be probed. Mark cortex_a76_erratum_1463225_debug_handler() as __always_inline to ensure this. Example splat prior to this patch (with recursive entries elided): | # echo p cortex_a76_erratum_1463225_debug_handler > /sys/kernel/debug/tracing/kprobe_events | # echo p do_el0_svc >> /sys/kernel/debug/tracing/kprobe_events | # echo 1 > /sys/kernel/debug/tracing/events/kprobes/enable | Insufficient stack space to handle exception! | ESR: 0x0000000096000047 -- DABT (current EL) | FAR: 0xffff800009cefff0 | Task stack: [0xffff800009cf0000..0xffff800009cf4000] | IRQ stack: [0xffff800008000000..0xffff800008004000] | Overflow stack: [0xffff00007fbc00f0..0xffff00007fbc10f0] | CPU: 0 PID: 145 Comm: sh Not tainted 6.0.0 #2 | Hardware name: linux,dummy-virt (DT) | pstate: 604003c5 (nZCv DAIF +PAN -UAO -TCO -DIT -SSBS BTYPE=--) | pc : arm64_enter_el1_dbg+0x4/0x20 | lr : el1_dbg+0x24/0x5c | sp : ffff800009cf0000 | x29: ffff800009cf0000 x28: ffff000002c74740 x27: 0000000000000000 | x26: 0000000000000000 x25: 0000000000000000 x24: 0000000000000000 | x23: 00000000604003c5 x22: ffff80000801745c x21: 0000aaaac95ac068 | x20: 00000000f2000004 x19: ffff800009cf0040 x18: 0000000000000000 | x17: 0000000000000000 x16: 0000000000000000 x15: 0000000000000000 | x14: 0000000000000000 x13: 0000000000000000 x12: 0000000000000000 | x11: 0000000000000010 x10: ffff800008c87190 x9 : ffff800008ca00d0 | x8 : 000000000000003c x7 : 0000000000000000 x6 : 0000000000000000 | x5 : 0000000000000000 x4 : 0000000000000000 x3 : 00000000000043a4 | x2 : 00000000f2000004 x1 : 00000000f2000004 x0 : ffff800009cf0040 | Kernel panic - not syncing: kernel stack overflow | CPU: 0 PID: 145 Comm: sh Not tainted 6.0.0 #2 | Hardware name: linux,dummy-virt (DT) | Call trace: | dump_backtrace+0xe4/0x104 | show_stack+0x18/0x4c | dump_stack_lvl+0x64/0x7c | dump_stack+0x18/0x38 | panic+0x14c/0x338 | test_taint+0x0/0x2c | panic_bad_stack+0x104/0x118 | handle_bad_stack+0x34/0x48 | __bad_stack+0x78/0x7c | arm64_enter_el1_dbg+0x4/0x20 | el1h_64_sync_handler+0x40/0x98 | el1h_64_sync+0x64/0x68 | cortex_a76_erratum_1463225_debug_handler+0x0/0x34 ... | el1h_64_sync_handler+0x40/0x98 | el1h_64_sync+0x64/0x68 | cortex_a76_erratum_1463225_debug_handler+0x0/0x34 ... | el1h_64_sync_handler+0x40/0x98 | el1h_64_sync+0x64/0x68 | cortex_a76_erratum_1463225_debug_handler+0x0/0x34 | el1h_64_sync_handler+0x40/0x98 | el1h_64_sync+0x64/0x68 | do_el0_svc+0x0/0x28 | el0t_64_sync_handler+0x84/0xf0 | el0t_64_sync+0x18c/0x190 | Kernel Offset: disabled | CPU features: 0x0080,00005021,19001080 | Memory Limit: none | ---[end Kernel panic - not syncing: kernel stack overflow]--- With this patch, cortex_a76_erratum_1463225_debug_handler() is inlined into el1_dbg(), and el1_dbg() cannot be probed: | # echo p cortex_a76_erratum_1463225_debug_handler > /sys/kernel/debug/tracing/kprobe_events | sh: write error: No such file or directory | # grep -w cortex_a76_errat ---truncated---		

CVE-2022-49892	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ftrace: Fix use-after-free for dynamic ftrace_ops KASAN reported a use-after-free with ftrace ops [1]. It was found from vmcore that perf had registered two ops with the same content successively, both dynamic. After unregistering the second ops, a use-after-free occurred. In ftrace_shutdown(), when the second ops is unregistered, the FTRACE_UPDATE_CALLS command is not set because there is another enabled ops with the same content. Also, both ops are dynamic and the ftrace callback function is ftrace_ops_list_func, so the FTRACE_UPDATE_TRACE_FUNC command will not be set. Eventually the value of 'command' will be 0 and ftrace_shutdown() will skip the rcu synchronization. However, ftrace may be activated. When the ops is released, another CPU may be accessing the ops. Add the missing synchronization to fix this problem. [1] BUG: KASAN: use-after-free in __ftrace_ops_list_func kernel/trace/ftrace.c:7020 [inline] BUG: KASAN: use-after-free in ftrace_ops_list_func+0x2b0/0x31c kernel/trace/ftrace.c:7049 Read of size 8 at addr ffff56551965bbc8 by task syz-executor.2/14468 CPU: 1 PID: 14468 Comm: syz-executor.2 Not tainted 5.10.0 #7 Hardware name: linux,dummy-virt (DT) Call trace: dump_backtrace+0x0/0x40c arch/arm64/kernel/stacktrace.c:132 show_stack+0x30/0x40 arch/arm64/kernel/stacktrace.c:196 __dump_stack lib/dump_stack.c:77 [inline] dump_stack+0x1b4/0x248 lib/dump_stack.c:118 print_address_description.constprop.0+0x28/0x48c mm/kasan/report.c:387 __kasan_report mm/kasan/report.c:547 [inline] kasan_report+0x118/0x210 mm/kasan/report.c:564 check_memory_region_inline mm/kasan/generic.c:187 [inline] __asan_load8+0x98/0xc0 mm/kasan/generic.c:253 __ftrace_ops_list_func kernel/trace/ftrace.c:7020 [inline] ftrace_ops_list_func+0x2b0/0x31c kernel/trace/ftrace.c:7049 ftrace_graph_call+0x0/0x4 __might_sleep+0x8/0x100 include/linux/perf_event.h:1170 __might_fault mm/memory.c:5183 [inline] __might_fault+0x58/0x70 mm/memory.c:5171 do_strncpy_from_user lib/strncpy_from_user.c:41 [inline] strncpy_from_user+0x1f4/0x4b0 lib/strncpy_from_user.c:139 getname_flags+0xb0/0x31c fs/namei.c:149 getname+0x2c/0x40 fs/namei.c:209 [...] Allocated by task 14445: kasan_save_stack+0x24/0x50 mm/kasan/common.c:48 kasan_set_track mm/kasan/common.c:56 [inline] __kasan_kmalloc mm/kasan/common.c:479 [inline] __kasan_kmalloc.constprop.0+0x110/0x13c mm/kasan/common.c:449 kasan_kmalloc+0xc/0x14 mm/kasan/common.c:493 kmem_cache_alloc_trace+0x440/0x924 mm/slub.c:2950 kmalloc include/linux/slab.h:563 [inline] kzalloc include/linux/slab.h:675 [inline] perf_event_alloc.part.0+0xb4/0x1350 kernel/events/core.c:11230 perf_event_alloc kernel/events/core.c:11733 [inline] __do_sys_perf_event_open kernel/events/core.c:11831 [inline] __se_sys_perf_event_open+0x550/0x15f4 kernel/events/core.c:11723 __arm64_sys_perf_event_open+0x6c/0x80 kernel/events/core.c:11723 [...] Freed by task 14445: kasan_save_stack+0x24/0x50 mm/kasan/common.c:48 kasan_set_track+0x24/0x34 mm/kasan/common.c:56 kasan_set_free_info+0x20/0x40 mm/kasan/generic.c:358 __kasan_slab_free.part.0+0x11c/0x1b0 mm/kasan/common.c:437 __kasan_slab_free mm/kasan/common.c:445 [inline] kasan_slab_free+0x2c/0x40 mm/kasan/common.c:446 slab_free_hook mm/slub.c:1569 [inline] slab_free_freelist_hook mm/slub.c:1608 [inline] slab_free mm/slub.c:3179 [inline] kfree+0x12c/0xc10 mm/slub.c:4176 perf_event_alloc.part.0+0xa0c/0x1350 kernel/events/core.c:11434 perf_event_alloc kernel/events/core.c:11733 [inline] __do_sys_perf_event_open kernel/events/core.c:11831 [inline]	2025-05-01	7.8

		__se_sys_perf_event_open+0x550/0x15f4 kernel/events/core.c:11723 [...]		
CVE-2022-49909	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: Bluetooth: L2CAP: fix use-after-free in l2cap_conn_del() When l2cap_recv_frame() is invoked to receive data, and the cid is L2CAP_CID_A2MP, if the channel does not exist, it will create a channel. However, after a channel is created, the hold operation of the channel is not performed. In this case, the value of channel reference counting is 1. As a result, after hci_error_reset() is triggered, l2cap_conn_del() invokes the close hook function of A2MP to release the channel. Then l2cap_chan_unlock(chan) will trigger UAF issue. The process is as follows: Receive data: l2cap_data_channel() a2mp_channel_create() --->channel ref is 2 l2cap_chan_put() --->channel ref is 1 Triger event: hci_error_reset() hci_dev_do_close() ... l2cap_disconn_cfm() l2cap_conn_del() l2cap_chan_hold() --->channel ref is 2 l2cap_chan_del() --->channel ref is 1 a2mp_chan_close_cb() --->channel ref is 0, release channel l2cap_chan_unlock() --->UAF of channel The detailed Call Trace is as follows: BUG: KASAN: use-after-free in __mutex_unlock_slowpath+0xa6/0x5e0 Read of size 8 at addr ffff8880160664b8 by task kworker/u11:1/7593 Workqueue: hci0 hci_error_reset Call Trace: <TASK> dump_stack_lvl+0xcd/0x134 print_report.cold+0x2ba/0x719 kasan_report+0xb1/0x1e0 kasan_check_range+0x140/0x190 __mutex_unlock_slowpath+0xa6/0x5e0 l2cap_conn_del+0x404/0x7b0 l2cap_disconn_cfm+0x8c/0xc0 hci_conn_hash_flush+0x11f/0x260 hci_dev_close_sync+0x5f5/0x11f0 hci_dev_do_close+0x2d/0x70 hci_error_reset+0x9e/0x140 process_one_work+0x98a/0x1620 worker_thread+0x665/0x1080 kthread+0x2e4/0x3a0 ret_from_fork+0x1f/0x30 </TASK> Allocated by task 7593: kasan_save_stack+0x1e/0x40 __kasan_kmalloc+0xa9/0xd0 l2cap_chan_create+0x40/0x930 amp_mgr_create+0x96/0x990 a2mp_channel_create+0x7d/0x150 l2cap_recv_frame+0x51b8/0x9a70 l2cap_recv_acldata+0xaa3/0xc00 hci_rx_work+0x702/0x1220 process_one_work+0x98a/0x1620 worker_thread+0x665/0x1080 kthread+0x2e4/0x3a0 ret_from_fork+0x1f/0x30 Freed by task 7593: kasan_save_stack+0x1e/0x40 kasan_set_track+0x21/0x30 kasan_set_free_info+0x20/0x30 ____kasan_slab_free+0x167/0x1c0 slab_free_freelist_hook+0x89/0x1c0 kfree+0xe2/0x580 l2cap_chan_put+0x22a/0x2d0 l2cap_conn_del+0x3fc/0x7b0 l2cap_disconn_cfm+0x8c/0xc0 hci_conn_hash_flush+0x11f/0x260 hci_dev_close_sync+0x5f5/0x11f0 hci_dev_do_close+0x2d/0x70	2025-05-01	7.8

		hci_error_reset+0x9e/0x140 process_one_work+0x98a/0x1620 worker_thread+0x665/0x1080 kthread+0x2e4/0x3a0 ret_from_fork+0x1f/0x30 Last potentially related work creation: kasan_save_stack+0x1e/0x40 __kasan_record_aux_stack+0xbe/0xd0 call_rcu+0x99/0x740 netlink_release+0xe6a/0x1cf0 __sock_release+0xcd/0x280 sock_close+0x18/0x20 __fput+0x27c/0xa90 task_work_run+0xdd/0x1a0 exit_to_user_mode_prepare+0x23c/0x250 syscall_exit_to_user_mode+0x19/0x50 do_syscall_64+0x42/0x80 entry_SYSCALL_64_after_hwframe+0x63/0xcd Second to last potentially related work creation: kasan_save_stack+0x1e/0x40 __kasan_record_aux_stack+0xbe/0xd0 call_rcu+0x99/0x740 netlink_release+0xe6a/0x1cf0 __sock_release+0xcd/0x280 sock_close+0x18/0x20 __fput+0x27c/0xa90 task_work_run+0xdd/0x1a0 exit_to_user_mode_prepare+0x23c/0x250 syscall_exit_to_user_mode+0x19/0x50 do_syscall_64+0x42/0x80 entry_SYSCALL_64_after_hwframe+0x63/0xcd		
CVE-2022-49921	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net: sched: Fix use after free in red_enqueue() We can't use "skb" again after passing it to qdisc_enqueue(). This is basically identical to commit 2f09707d0c97 ("sch_sfb: Also store skb len before calling child enqueue").	2025-05-01	7.8
CVE-2025-24206	apple - multiple products	An authentication issue was addressed with improved state management. This issue is fixed in macOS Sequoia 15.4, tvOS 18.4, macOS Ventura 13.7.5, iPadOS 17.7.6, macOS Sonoma 14.7.5, iOS 18.4 and iPadOS 18.4, visionOS 2.4. An attacker on the local network may be able to bypass authentication policy.	2025-04-29	7.7
CVE-2022-21546	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: scsi: target: Fix WRITE_SAME No Data Buffer crash In newer version of the SBC specs, we have a NDOB bit that indicates there is no data buffer that gets written out. If this bit is set using commands like "sg_write_same --ndob" we will crash in target_core_iblock/file's execute_write_same handlers when we go to access the se_cmd->t_data_sg because its NULL. This patch adds a check for the NDOB bit in the common WRITE SAME code because we don't support it. And, it adds a check for zero SG elements in each handler in case the initiator tries to send a normal WRITE SAME with no data buffer.	2025-05-02	7.7
CVE-2025-31650	apache - multiple products	Improper Input Validation vulnerability in Apache Tomcat. Incorrect error handling for some invalid HTTP priority headers resulted in incomplete clean-up of the failed request which created a memory leak. A large number of such requests could trigger an OutOfMemoryException resulting in a denial of service. This issue affects Apache Tomcat: from 9.0.76 through 9.0.102, from 10.1.10 through 10.1.39, from 11.0.0-M2 through 11.0.5. Users are recommended to upgrade to version 9.0.104, 10.1.40 or 11.0.6 which fix the issue.	2025-04-28	7.5
CVE-2025-33074	microsoft - Azure Functions	Improper verification of cryptographic signature in Microsoft Azure Functions allows an authorized attacker to execute code over a network.	2025-04-30	7.5
CVE-2025-3224	docker - desktop	A vulnerability in the update process of Docker Desktop for Windows versions prior to 4.41.0 could allow a local, low-privileged attacker to escalate privileges to SYSTEM. During an update, Docker Desktop attempts to delete files and subdirectories under the path C:\ProgramData\Docker\config with high privileges. However, this directory often does not exist by default, and C:\ProgramData\ allows normal users to create new directories. By creating a malicious Docker\config folder structure at this location, an attacker can force the privileged update process to delete or manipulate arbitrary system files, leading to Elevation of Privilege.	2025-04-28	7.3
CVE-2025-2170	sonicwall - SMA1000	A Server-side request forgery (SSRF) vulnerability has been identified in the SMA1000 Appliance Work Place interface, which in specific conditions could potentially enable a remote unauthenticated attacker to cause the appliance to make requests to an unintended location.	2025-04-30	7.2

CVE-2025-4085	mozilla - multiple products	An attacker with control over a content process could potentially leverage the privileged UITour actor to leak sensitive information or escalate privileges. This vulnerability affects Firefox < 138 and Thunderbird < 138.	2025-04-29	7.1
CVE-2022-49844	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: can: dev: fix skb drop check In commit a6d190f8c767 ("can: skb: drop tx skb if in listen only mode") the priv->ctrlmode element is read even on virtual CAN interfaces that do not create the struct can_priv at startup. This out-of-bounds read may lead to CAN frame drops for virtual CAN interfaces like vcan and vxcan. This patch mainly reverts the original commit and adds a new helper for CAN interface drivers that provide the required information in struct can_priv. [mkl: patch pch_can, too]	2025-05-01	7.1
CVE-2022-49919	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: netfilter: nf_tables: release flow rule object from commit path No need to postpone this to the commit release path, since no packets are walking over this object, this is accessed from control plane only. This helped uncovered UAF triggered by races with the netlink notifier.	2025-05-01	7.0
CVE-2025-24251	apple - multiple products	The issue was addressed with improved checks. This issue is fixed in macOS Sequoia 15.4, tvOS 18.4, macOS Ventura 13.7.5, iPadOS 17.7.6, macOS Sonoma 14.7.5, iOS 18.4 and iPadOS 18.4, watchOS 11.4, visionOS 2.4. An attacker on the local network may cause an unexpected app termination.	2025-04-29	6.5
CVE-2025-30445	apple - multiple products	A type confusion issue was addressed with improved checks. This issue is fixed in macOS Sequoia 15.4, tvOS 18.4, macOS Ventura 13.7.5, iPadOS 17.7.6, macOS Sonoma 14.7.5, iOS 18.4 and iPadOS 18.4, visionOS 2.4. An attacker on the local network may cause an unexpected app termination.	2025-04-29	6.5
CVE-2025-31203	apple - multiple products	An integer overflow was addressed with improved input validation. This issue is fixed in macOS Sequoia 15.4, tvOS 18.4, iPadOS 17.7.6, macOS Sonoma 14.7.5, iOS 18.4 and iPadOS 18.4, watchOS 11.4, visionOS 2.4. An attacker on the local network may be able to cause a denial-of-service.	2025-04-29	6.5
CVE-2025-4086	mozilla - multiple products	A specially crafted filename containing a large number of encoded newline characters could obscure the file's extension when displayed in the download dialog. *This bug only affects Thunderbird for Android. Other versions of Thunderbird are unaffected.* This vulnerability affects Firefox < 138 and Thunderbird < 138.	2025-04-29	6.5
CVE-2025-4087	mozilla - multiple products	A vulnerability was identified in Thunderbird where XPath parsing could trigger undefined behavior due to missing null checks during attribute access. This could lead to out-of-bounds read access and potentially, memory corruption. This vulnerability affects Firefox < 138, Firefox ESR < 128.10, Thunderbird < 138, and Thunderbird < 128.10.	2025-04-29	6.5
CVE-2025-4088	mozilla - multiple products	A security vulnerability in Thunderbird allowed malicious sites to use redirects to send credentialed requests to arbitrary endpoints on any site that had invoked the Storage Access API. This enabled potential Cross-Site Request Forgery attacks across origins. This vulnerability affects Firefox < 138 and Thunderbird < 138.	2025-04-29	6.5
CVE-2025-4090	mozilla - multiple products	A vulnerability existed in Thunderbird for Android where potentially sensitive library locations were logged via Logcat. This vulnerability affects Firefox < 138 and Thunderbird < 138.	2025-04-29	6.5
CVE-2025-4091	mozilla - multiple products	Memory safety bugs present in Firefox 137, Thunderbird 137, Firefox ESR 128.9, and Thunderbird 128.9. 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 affects Firefox < 138, Firefox ESR < 128.10, Thunderbird < 138, and Thunderbird < 128.10.	2025-04-29	6.5
CVE-2025-4092	mozilla - multiple products	Memory safety bugs present in Firefox 137 and Thunderbird 137. 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 affects Firefox < 138 and Thunderbird < 138.	2025-04-29	6.5
CVE-2025-4093	mozilla - multiple products	Memory safety bug present in Firefox ESR 128.9, and Thunderbird 128.9. This bug showed evidence of memory corruption and we presume that with enough effort this could have been exploited to run arbitrary code. This vulnerability affects Firefox ESR < 128.10 and Thunderbird < 128.10.	2025-04-29	6.5
CVE-2025-3599	symantec - Symantec Endpoint Protection	Symantec Endpoint Protection Windows Agent, running an ERASER Engine prior to 119.1.7.8, may be susceptible to an Elevation of Privilege vulnerability, which may allow an attacker to delete resources that are normally protected from an application or user.	2025-04-30	6.5
CVE-2025-24132	apple - multiple products	The issue was addressed with improved memory handling. This issue is fixed in AirPlay audio SDK 2.7.1, AirPlay video SDK 3.6.0.126, CarPlay Communication Plug-in R18.1. An attacker on the local network may cause an unexpected app termination.	2025-04-30	6.5
CVE-2025-30422	apple - multiple products	A buffer overflow was addressed with improved input validation. This issue is fixed in AirPlay audio SDK 2.7.1, AirPlay video SDK 3.6.0.126, CarPlay Communication Plug-in R18.1. An attacker on the local network may cause an unexpected app termination.	2025-04-30	6.5
CVE-2025-47153	debian - trixie	Certain build processes for libuv and Node.js for 32-bit systems, such as for the nodejs binary package through nodejs_20.19.0+dfsg-2_i386.deb for Debian GNU/Linux, have an inconsistent off_t size (e.g., building on i386 Debian always uses _FILE_OFFSET_BITS=64 for the libuv dynamic library, but uses the _FILE_OFFSET_BITS global system default of 32 for nodejs), leading to out-of-bounds access. NOTE: this is not a problem in the Node.js software itself. In particular, the Node.js website's download page does not offer prebuilt Node.js for Linux on i386.	2025-05-01	6.5

CVE-2025-27365	ibm - MQ Operator	IBM MQ Operator LTS 2.0.0 through 2.0.29, MQ Operator CD 3.0.0, 3.0.1, 3.1.0 through 3.1.3, 3.3.0, 3.4.0, 3.4.1, 3.5.0, 3.5.1, and MQ Operator SC2 3.2.0 through 3.2.10 Client connecting to a MQ Queue Manager can cause a SIGSEGV in the AMQRMPPA channel process terminating it.	2025-05-01	6.5
CVE-2024-55909	ibm - Concert Software	IBM Concert Software 1.0.0 through 1.0.5 could allow an authenticated user to cause a denial of service due to the expansion of archive files without controlling resource consumption.	2025-05-02	6.5
CVE-2024-55910	ibm - Concert Software	IBM Concert Software 1.0.0 through 1.0.5 is vulnerable to server-side request forgery (SSRF). This may allow an authenticated attacker to send unauthorized requests from the system, potentially leading to network enumeration or facilitating other attacks.	2025-05-02	6.5
CVE-2025-29825	microsoft - Microsoft Edge (Chromium-based)	User interface (ui) misrepresentation of critical information in Microsoft Edge (Chromium-based) allows an unauthorized attacker to perform spoofing over a network.	2025-05-02	6.5
CVE-2025-1838	ibm - Cloud Pak for Business Automation	IBM Cloud Pak for Business Automation 24.0.0 and 24.0.1 through 24.0.1 IF001 Authoring allows an authenticated user to bypass client-side data validation in an authoring user interface which could cause a denial of service.	2025-05-03	6.5
CVE-2024-10635	proofpoint - multiple products	Enterprise Protection contains an improper input validation vulnerability in attachment defense that allows an unauthenticated remote attacker to bypass attachment scanning security policy by sending a malicious S/MIME attachment with an opaque signature. When opened by a recipient in a downstream email client, the malicious attachment could cause partial loss of integrity and confidentiality to their system.	2025-04-28	6.1
CVE-2025-1551	ibm - Operational Decision Manager	IBM Operational Decision Manager 8.11.0.1, 8.11.1.0, 8.12.0.1, and 9.0.0.1 is vulnerable to cross-site scripting. This vulnerability allows an unauthenticated attacker to embed arbitrary JavaScript code in the Web UI thus altering the intended functionality potentially leading to credentials disclosure within a trusted session.	2025-04-29	6.1
CVE-2024-41753	ibm - Cloud Pak for Business Automation	IBM Cloud Pak for Business Automation 24.0.0 through 24.0.0 IF004 and 24.0.1 through 24.0.1 IF001 is vulnerable to cross-site scripting. This vulnerability allows an unauthenticated attacker to embed arbitrary JavaScript code in the Web UI thus altering the intended functionality potentially leading to credentials disclosure within a trusted session.	2025-05-03	6.1
CVE-2025-1333	ibm - MQ Operator	IBM MQ Container when used with the IBM MQ Operator LTS 2.0.0 through 2.0.29, MQ Operator CD 3.0.0, 3.0.1, 3.1.0 through 3.1.3, 3.3.0, 3.4.0, 3.4.1, 3.5.0, 3.5.1, and MQ Operator SC2 3.2.0 through 3.2.10 and configured with Cloud Pak for Integration Keycloak could disclose sensitive information to a privileged user.	2025-05-01	6
CVE-2025-4082	mozilla - multiple products	Modification of specific WebGL shader attributes could trigger an out-of-bounds read, which, when chained with other vulnerabilities, could be used to escalate privileges. *This bug only affects Thunderbird for macOS. Other versions of Thunderbird are unaffected.* This vulnerability affects Firefox < 138, Firefox ESR < 128.10, Firefox ESR < 115.23, Thunderbird < 138, and Thunderbird < 128.10.	2025-04-29	5.9
CVE-2024-55912	ibm - Concert Software	IBM Concert Software 1.0.0 through 1.0.5 uses weaker than expected cryptographic algorithms that could allow an attacker to decrypt highly sensitive information.	2025-05-02	5.9
CVE-2025-24179	apple - multiple products	A null pointer dereference was addressed with improved input validation. This issue is fixed in iOS 18.3 and iPadOS 18.3, visionOS 2.3, macOS Ventura 13.7.5, iPadOS 17.7.6, macOS Sonoma 14.7.5, macOS Sequoia 15.3, tvOS 18.3. An attacker on the local network may be able to cause a denial-of-service.	2025-04-29	5.7
CVE-2025-24270	apple - multiple products	This issue was addressed by removing the vulnerable code. This issue is fixed in macOS Sequoia 15.4, tvOS 18.4, macOS Ventura 13.7.5, iPadOS 17.7.6, macOS Sonoma 14.7.5, iOS 18.4 and iPadOS 18.4, visionOS 2.4. An attacker on the local network may be able to leak sensitive user information.	2025-04-29	5.7
CVE-2025-31197	apple - multiple products	The issue was addressed with improved checks. This issue is fixed in macOS Sequoia 15.4, tvOS 18.4, macOS Ventura 13.7.5, iPadOS 17.7.6, macOS Sonoma 14.7.5, iOS 18.4 and iPadOS 18.4, visionOS 2.4. An attacker on the local network may cause an unexpected app termination.	2025-04-29	5.7
CVE-2025-4084	mozilla - multiple products	Due to insufficient escaping of the special characters in the "copy as cURL" feature, an attacker could trick a user into using this command, potentially leading to local code execution on the user's system. *This bug only affects Firefox for Windows. Other versions of Firefox are unaffected.* This vulnerability affects Firefox ESR < 128.10, Firefox ESR < 115.23, and Thunderbird < 128.10.	2025-04-29	5.7
CVE-2025-31202	apple - multiple products	A null pointer dereference was addressed with improved input validation. This issue is fixed in iOS 18.4 and iPadOS 18.4, macOS Sequoia 15.4, tvOS 18.4, visionOS 2.4. An attacker on the local network may be able to cause a denial-of-service.	2025-04-29	5.5
CVE-2024-58099	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: vmxnet3: Fix packet corruption in vmxnet3_xdp_xmit_frame Andrew and Nikolay reported connectivity issues with Cilium's service load-balancing in case of vmxnet3. If a BPF program for native XDP adds an encapsulation header such as IPIP and transmits the packet out the same interface, then in case of vmxnet3 a corrupted packet is being sent and subsequently dropped on the path. vmxnet3_xdp_xmit_frame() which is called e.g. via vmxnet3_run_xdp() through vmxnet3_xdp_xmit_back() calculates an incorrect DMA address: page = virt_to_page(xdpf->data); tbi->dma_addr = page_pool_get_dma_addr(page) +	2025-04-29	5.5

		VMXNET3_XDP_HEADROOM; dma_sync_single_for_device(&adapter->pdev->dev, tbi->dma_addr, buf_size, DMA_TO_DEVICE); The above assumes a fixed offset (VMXNET3_XDP_HEADROOM), but the XDP BPF program could have moved xdp->data. While the passed buf_size is correct (xdpf->len), the dma_addr needs to have a dynamic offset which can be calculated as xdpf->data - (void *)xdpf, that is, xdp->data - xdp->data_hard_start.		
CVE-2025-24091	apple - multiple products	An app could impersonate system notifications. Sensitive notifications now require restricted entitlements. This issue is fixed in iOS 18.3 and iPadOS 18.3, iPadOS 17.7.3. An app may be able to cause a denial-of-service.	2025-04-30	5.5
CVE-2022-49837	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: bpf: Fix memory leaks in __check_func_call kmemleak reports this issue: unreferenced object 0xffff88817139d000 (size 2048): comm "test_progs", pid 33246, jiffies 4307381979 (age 45851.820s) hex dump (first 32 bytes): 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .... 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .... backtrace: [<0000000045f075f0>] kmalloc_trace+0x27/0xa0 [<0000000098b7c90a>] __check_func_call+0x316/0x1230 [<00000000b4c3c403>] check_helper_call+0x172e/0x4700 [<00000000aa3875b7>] do_check+0x21d8/0x45e0 [<000000001147357b>] do_check_common+0x767/0xaf0 [<00000000b5a595b4>] bpf_check+0x43e3/0x5bc0 [<0000000011e391b1>] bpf_prog_load+0xf26/0x1940 [<0000000007f765c0>] __sys_bpf+0xd2c/0x3650 [<00000000839815d6>] __x64_sys_bpf+0x75/0xc0 [<00000000946ee250>] do_syscall_64+0x3b/0x90 [<0000000000506b7f>] entry_SYSCALL_64_after_hwframe+0x63/0xcd The root case here is: In function prepare_func_exit(), the callee is not released in the abnormal scenario after "state->curframe--;". To fix, move "state->curframe--;" to the very bottom of the function, right when we free callee and reset frame[] pointer to NULL, as Andrii suggested. In addition, function __check_func_call() has a similar problem. In the abnormal scenario before "state->curframe++;", the callee also should be released by free_func_state().	2025-05-01	5.5
CVE-2022-49839	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: scsi: scsi_transport_sas: Fix error handling in sas_phy_add() If transport_add_device() fails in sas_phy_add(), the kernel will crash trying to delete the device in transport_remove_device() called from sas_remove_host(). Unable to handle kernel NULL pointer dereference at virtual address 0000000000000108 CPU: 61 PID: 42829 Comm: rmmod Kdump: loaded Tainted: G W 6.1.0-rc1+ #173 pstate: 60000005 (nZCv daif -PAN -UAO -TCO -DIT -SSBS BTYPE=--) pc : device_del+0x54/0x3d0 lr : device_del+0x37c/0x3d0 Call trace: device_del+0x54/0x3d0 attribute_container_class_device_del+0x28/0x38 transport_remove_classdev+0x6c/0x80 attribute_container_device_trigger+0x108/0x110 transport_remove_device+0x28/0x38 sas_phy_delete+0x30/0x60 [scsi_transport_sas] do_sas_phy_delete+0x6c/0x80 [scsi_transport_sas] device_for_each_child+0x68/0xb0 sas_remove_children+0x40/0x50 [scsi_transport_sas] sas_remove_host+0x20/0x38 [scsi_transport_sas] hisi_sas_remove+0x40/0x68 [hisi_sas_main] hisi_sas_v2_remove+0x20/0x30 [hisi_sas_v2_hw] platform_remove+0x2c/0x60 Fix this by checking and handling return value of transport_add_device() in sas_phy_add().	2025-05-01	5.5
CVE-2022-49845	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: can: j1939: j1939_send_one(): fix missing CAN header initialization The read access to struct canxl_frame::len inside of a j1939 created	2025-05-01	5.5

		skbuff revealed a missing initialization of reserved and later filled elements in struct can_frame. This patch initializes the 8 byte CAN header with zero.		
CVE-2022-49848	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: phy: qcom-qmp-combo: fix NULL-deref on runtime resume Commit fc64623637da ("phy: qcom-qmp-combo,usb: add support for separate PCS_USB region") started treating the PCS_USB registers as potentially separate from the PCS registers but used the wrong base when no PCS_USB offset has been provided. Fix the PCS_USB base used at runtime resume to prevent dereferencing a NULL pointer on platforms that do not provide a PCS_USB offset (e.g. SC7180).	2025-05-01	5.5
CVE-2022-49850	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: nilfs2: fix deadlock in nilfs_count_free_blocks() A semaphore deadlock can occur if nilfs_get_block() detects metadata corruption while locating data blocks and a superblock writeback occurs at the same time: task 1 task 2 ------ ------ * A file operation * nilfs_truncate() nilfs_get_block() down_read(rwsem A) <-- nilfs_bmap_lookup_contig() ... generic_shutdown_super() nilfs_put_super() * Prepare to write superblock * down_write(rwsem B) <-- nilfs_cleanup_super() * Detect b-tree corruption * nilfs_set_log_cursor() nilfs_bmap_convert_error() nilfs_count_free_blocks() __nilfs_error() down_read(rwsem A) <-- nilfs_set_error() down_write(rwsem B) <-- *** DEADLOCK *** Here, nilfs_get_block() readlocks rwsem A (= NILFS_MDT(dat_inode)->mi_sem) and then calls nilfs_bmap_lookup_contig(), but if it fails due to metadata corruption, __nilfs_error() is called from nilfs_bmap_convert_error() inside the lock section. Since __nilfs_error() calls nilfs_set_error() unless the filesystem is read-only and nilfs_set_error() attempts to writelock rwsem B (= nilfs->ns_sem) to write back superblock exclusively, hierarchical lock acquisition occurs in the order rwsem A -> rwsem B. Now, if another task starts updating the superblock, it may writelock rwsem B during the lock sequence above, and can deadlock trying to readlock rwsem A in nilfs_count_free_blocks(). However, there is actually no need to take rwsem A in nilfs_count_free_blocks() because it, within the lock section, only reads a single integer data on a shared struct with nilfs_sufile_get_ncleansegs(). This has been the case after commit aa474a220180 ("nilfs2: add local variable to cache the number of clean segments"), that is, even before this bug was introduced. So, this resolves the deadlock problem by just not taking the semaphore in nilfs_count_free_blocks().	2025-05-01	5.5
CVE-2022-49853	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net: macvlan: fix memory leaks of macvlan_common_newlink kmemleak reports memory leaks in macvlan_common_newlink, as follows: ip link add link eth0 name .. type macvlan mode source macaddr add <MAC-ADDR> kmemleak reports: unreferenced object 0xffff8880109bb140 (size 64): comm "ip", pid 284, jiffies 4294986150 (age 430.108s) hex dump (first 32 bytes):	2025-05-01	5.5

		00 00 00 00 00 00 00 00 b8 aa 5a 12 80 88 ff ff ....Z..... 80 1b fa 0d 80 88 ff ff 1e ff ac af c7 c1 6b 6b ....kk backtrace: [<ffffffff813e06a7>] kmem_cache_alloc_trace+0x1c7/0x300 [<ffffffff81b66025>] macvlan_hash_add_source+0x45/0xc0 [<ffffffff81b66a67>] macvlan_changelink_sources+0xd7/0x170 [<ffffffff81b6775c>] macvlan_common_newlink+0x38c/0x5a0 [<ffffffff81b6797e>] macvlan_newlink+0xe/0x20 [<ffffffff81d97f8f>] __rtnl_newlink+0x7af/0xa50 [<ffffffff81d98278>] rtnl_newlink+0x48/0x70 ... In the scenario where the macvlan mode is configured as 'source', macvlan_changelink_sources() will be execured to reconfigure list of remote source mac addresses, at the same time, if register_netdevice() return an error, the resource generated by macvlan_changelink_sources() is not cleaned up. Using this patch, in the case of an error, it will execute macvlan_flush_sources() to ensure that the resource is cleaned up.		
CVE-2022-49854	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: mctp: Fix an error handling path in mctp_init() If mctp_neigh_init() return error, the routes resources should be released in the error handling path. Otherwise some resources leak.	2025-05-01	5.5
CVE-2022-49855	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net: wwan: iosm: fix memory leak in ipc_pcie_read_bios_cfg ipc_pcie_read_bios_cfg() is using the acpi_evaluate_dsm() to obtain the wwan power state configuration from BIOS but is not freeing the acpi_object. The acpi_evaluate_dsm() returned acpi_object to be freed. Free the acpi_object after use.	2025-05-01	5.5
CVE-2022-49857	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net: marvell: prestera: fix memory leak in prestera_rxtx_switch_init() When prestera_sdma_switch_init() failed, the memory pointed to by sw->rxtx isn't released. Fix it. Only be compiled, not be tested.	2025-05-01	5.5
CVE-2022-49860	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: dmaengine: ti: k3-udma-glue: fix memory leak when register device fail If device_register() fails, it should call put_device() to give up reference, the name allocated in dev_set_name() can be freed in callback function kobject_cleanup().	2025-05-01	5.5
CVE-2022-49861	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: dmaengine: mv_xor_v2: Fix a resource leak in mv_xor_v2_remove() A clk_prepare_enable() call in the probe is not balanced by a corresponding clk_disable_unprepare() in the remove function. Add the missing call.	2025-05-01	5.5
CVE-2022-49862	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: tipc: fix the msg->req tlv len check in tipc_nl_compat_name_table_dump_header This is a follow-up for commit 974cb0e3e7c9 ("tipc: fix uninit-value in tipc_nl_compat_name_table_dump") where it should have type casted sizeof(..) to int to work when TLV_GET_DATA_LEN() returns a negative value. syzbot reported a call trace because of it: BUG: KMSAN: uninit-value in ... tipc_nl_compat_name_table_dump+0x841/0xea0 net/tipc/netlink_compat.c:934 __tipc_nl_compat_dumpit+0xab2/0x1320 net/tipc/netlink_compat.c:238 tipc_nl_compat_dumpit+0x991/0xb50 net/tipc/netlink_compat.c:321 tipc_nl_compat_recv+0xb6e/0x1640 net/tipc/netlink_compat.c:1324 genl_family_rcv_msg_doit net/netlink/genetlink.c:731 [inline] genl_family_rcv_msg net/netlink/genetlink.c:775 [inline] genl_rcv_msg+0x103f/0x1260 net/netlink/genetlink.c:792 netlink_rcv_skb+0x3a5/0x6c0 net/netlink/af_netlink.c:2501 genl_rcv+0x3c/0x50 net/netlink/genetlink.c:803 netlink_unicast_kernel net/netlink/af_netlink.c:1319 [inline] netlink_unicast+0xf3b/0x1270 net/netlink/af_netlink.c:1345	2025-05-01	5.5

		netlink_sendmsg+0x1288/0x1440 net/netlink/af_netlink.c:1921 sock_sendmsg_nosec net/socket.c:714 [inline] sock_sendmsg net/socket.c:734 [inline]		
CVE-2022-49863	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: can: af_can: fix NULL pointer dereference in can_rx_register() It causes NULL pointer dereference when testing as following: (a) use syscall(__NR_socket, 0x10ul, 3ul, 0) to create netlink socket. (b) use syscall(__NR_sendmsg, ...) to create bond link device and vxcan link device, and bind vxcan device to bond device (can also use ifenslave command to bind vxcan device to bond device). (c) use syscall(__NR_socket, 0x1dul, 3ul, 1) to create CAN socket. (d) use syscall(__NR_bind, ...) to bind the bond device to CAN socket. The bond device invokes the can-raw protocol registration interface to receive CAN packets. However, ml_priv is not allocated to the dev, dev_rcv_lists is assigned to NULL in can_rx_register(). In this case, it will occur the NULL pointer dereference issue. The following is the stack information: BUG: kernel NULL pointer dereference, address: 0000000000000008 PGD 122a4067 P4D 122a4067 PUD 1223c067 PMD 0 Oops: 0000 [#1] PREEMPT SMP RIP: 0010:can_rx_register+0x12d/0x1e0 Call Trace: <TASK> raw_enable_filters+0x8d/0x120 raw_enable_allfilters+0x3b/0x130 raw_bind+0x118/0x4f0 __sys_bind+0x163/0x1a0 __x64_sys_bind+0x1e/0x30 do_syscall_64+0x35/0x80 entry_SYSCALL_64_after_hwframe+0x63/0xcd </TASK>	2025-05-01	5.5
CVE-2022-49864	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: drm/amdkfd: Fix NULL pointer dereference in svm_migrate_to_ram() ./drivers/gpu/drm/amd/amdkfd/kfd_migrate.c:985:58-62: ERROR: p is NULL but dereferenced.	2025-05-01	5.5
CVE-2022-49866	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net: wwan: mhi: fix memory leak in mhi_mbim_dellink MHI driver registers network device without setting the needs_free_netdev flag, and does NOT call free_netdev() when unregisters network device, which causes a memory leak. This patch sets needs_free_netdev to true when registers network device, which makes netdev subsystem call free_netdev() automatically after unregister_netdevice().	2025-05-01	5.5
CVE-2022-49867	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net: wwan: iosm: fix memory leak in ipc_wwan_dellink IOSM driver registers network device without setting the needs_free_netdev flag, and does NOT call free_netdev() when unregisters network device, which causes a memory leak. This patch sets needs_free_netdev to true when registers network device, which makes netdev subsystem call free_netdev() automatically after unregister_netdevice().	2025-05-01	5.5
CVE-2022-49869	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: bnxt_en: Fix possible crash in bnxt_hwrm_set_coal() During the error recovery sequence, the rtnl_lock is not held for the entire duration and some datastructures may be freed during the sequence. Check for the BNXT_STATE_OPEN flag instead of netif_running() to ensure that the device is fully operational before proceeding to reconfigure the coalescing settings. This will fix a possible crash like this: BUG: unable to handle kernel NULL pointer dereference at 0000000000000000 PGD 0 P4D 0 Oops: 0000 [#1] SMP NOPTI CPU: 10 PID: 181276 Comm: ethtool Kdump: loaded Tainted: G IOE --------- - - 4.18.0-348.el8.x86_64 #1 Hardware name: Dell Inc. PowerEdge R740/0F9N89, BIOS 2.3.10 08/15/2019	2025-05-01	5.5

		RIP: 0010:bnxt_hwrm_set_coal+0x1fb/0x2a0 [bnxt_en] Code: c2 66 83 4e 22 08 66 89 46 1c e8 10 cb 00 00 41 83 c6 01 44 39 b3 68 01 00 00 0f 8e a3 00 00 00 48 8b 93 c8 00 00 00 49 63 c6 <48> 8b 2c c2 48 8b 85 b8 02 00 00 48 85 c0 74 2e 48 8b 74 24 08 f6 RSP: 0018:ffffb11c8dcaba50 EFLAGS: 00010246 RAX: 0000000000000000 RBX: ffff8d168a8b0ac0 RCX: 00000000000000c5 RDX: 0000000000000000 RSI: ffff8d162f72c000 RDI: ffff8d168a8b0b28 RBP: 0000000000000000 R08: b6e1f68a12e9a7eb R09: 0000000000000000 R10: 0000000000000001 R11: 0000000000000037 R12: ffff8d168a8b109c R13: ffff8d168a8b10aa R14: 0000000000000000 R15: ffffffffc01ac4e0 FS: 00007f3852e4c740(0000) GS:ffff8d24c0080000(0000) knlGS:0000000000000000 CS: 0010 DS: 0000 ES: 0000 CR0: 0000000080050033 CR2: 0000000000000000 CR3: 000000041b3ee003 CR4: 00000000007706e0 DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000 DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400 PKRU: 55555554 Call Trace: ethnl_set_coalesce+0x3ce/0x4c0 genl_family_rcv_msg_doit.isra.15+0x10f/0x150 genl_family_rcv_msg+0xb3/0x160 ? coalesce_fill_reply+0x480/0x480 genl_rcv_msg+0x47/0x90 ? genl_family_rcv_msg+0x160/0x160 netlink_rcv_skb+0x4c/0x120 genl_rcv+0x24/0x40 netlink_unicast+0x196/0x230 netlink_sendmsg+0x204/0x3d0 sock_sendmsg+0x4c/0x50 __sys_sendto+0xee/0x160 ? syscall_trace_enter+0x1d3/0x2c0 ? __audit_syscall_exit+0x249/0x2a0 __x64_sys_sendto+0x24/0x30 do_syscall_64+0x5b/0x1a0 entry_SYSCALL_64_after_hwframe+0x65/0xca RIP: 0033:0x7f38524163bb		
CVE-2022-49871	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net: tun: Fix memory leaks of napi_get_frags kmemleak reports after running test_progs: unreferenced object 0xffff8881b1672dc0 (size 232): comm "test_progs", pid 394388, jiffies 4354712116 (age 841.975s) hex dump (first 32 bytes): e0 84 d7 a8 81 88 ff ff 80 2c 67 b1 81 88 ff ff .....,g..... 00 40 c5 9b 81 88 ff ff 00 00 00 00 00 00 00 00 .@.............. backtrace: [<00000000c8f01748>] napi_skb_cache_get+0xd4/0x150 [<0000000041c7fc09>] __napi_build_skb+0x15/0x50 [<00000000431c7079>] __napi_alloc_skb+0x26e/0x540 [<000000003ecfa30e>] napi_get_frags+0x59/0x140 [<0000000099b2199e>] tun_get_user+0x183d/0x3bb0 [tun] [<000000008a5adef0>] tun_chr_write_iter+0xc0/0x1b1 [tun] [<0000000049993ff4>] do_iter_readv_writev+0x19f/0x320 [<000000008f338ea2>] do_iter_write+0x135/0x630 [<000000008a3377a4>] vfs_writev+0x12e/0x440 [<00000000a6b5639a>] do_writev+0x104/0x280 [<00000000ccf065d8>] do_syscall_64+0x3b/0x90 [<00000000d776e329>] entry_SYSCALL_64_after_hwframe+0x63/0xcd The issue occurs in the following scenarios: tun_get_user() napi_gro_frags() napi_frags_finish() case GRO_NORMAL: gro_normal_one() list_add_tail(&skb->list, &napi->rx_list); <-- While napi->rx_count < READ_ONCE(gro_normal_batch), <-- gro_normal_list() is not called, napi->rx_list is not empty <-- not ask to complete the gro work, will cause memory leaks in <-- following tun_napi_del() ... tun_napi_del() netif_napi_del() __netif_napi_del() <-- &napi->rx_list is not empty, which caused memory leaks To fix, add napi_complete() after napi_gro_frags().	2025-05-01	5.5
CVE-2022-49873	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: bpf: Fix wrong reg type conversion in release_reference()	2025-05-01	5.5

		Some helper functions will allocate memory. To avoid memory leaks, the verifier requires the eBPF program to release these memories by calling the corresponding helper functions. When a resource is released, all pointer registers corresponding to the resource should be invalidated. The verifier use release_references() to do this job, by apply __mark_reg_unknown() to each relevant register. It will give these registers the type of SCALAR_VALUE. A register that will contain a pointer value at runtime, but of type SCALAR_VALUE, which may allow the unprivileged user to get a kernel pointer by storing this register into a map. Using __mark_reg_not_init() while NOT allow_ptr_leaks can mitigate this problem.		
CVE-2022-49874	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: HID: hyperv: fix possible memory leak in mousevsc_probe() If hid_add_device() returns error, it should call hid_destroy_device() to free hid_dev which is allocated in hid_allocate_device().	2025-05-01	5.5
CVE-2022-49875	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: bpftool: Fix NULL pointer dereference when pin {PROG, MAP, LINK} without FILE When using bpftool to pin {PROG, MAP, LINK} without FILE, segmentation fault will occur. The reson is that the lack of FILE will cause strlen to trigger NULL pointer dereference. The corresponding stacktrace is shown below: do_pin do_pin_any do_pin_fd mount_bpffs_for_pin strlen(name) <- NULL pointer dereference Fix it by adding validation to the common process.	2025-05-01	5.5
CVE-2022-49876	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: wifi: mac80211: fix general-protection-fault in ieee80211_subif_start_xmit() When device is running and the interface status is changed, the gpf issue is triggered. The problem triggering process is as follows: Thread A: Thread B ieee80211_runtime_change_iftype() process_one_work() ... ... ieee80211_do_stop() ... ... sdata->bss = NULL ... ... ieee80211_subif_start_xmit() ieee80211_multicast_to_unicast //!sdata->bss->multicast_to_unicast cause gpf issue When the interface status is changed, the sending queue continues to send packets. After the bss is set to NULL, the bss is accessed. As a result, this causes a general-protection-fault issue. The following is the stack information: general protection fault, probably for non-canonical address 0xdffffc000000002f: 0000 [#1] PREEMPT SMP KASAN KASAN: null-ptr-deref in range [0x0000000000000178-0x000000000000017f] Workqueue: mld mld_ifc_work RIP: 0010:ieee80211_subif_start_xmit+0x25b/0x1310 Call Trace: <TASK> dev_hard_start_xmit+0x1be/0x990 __dev_queue_xmit+0x2c9a/0x3b60 ip6_finish_output2+0xf92/0x1520 ip6_finish_output+0x6af/0x11e0 ip6_output+0x1ed/0x540 mld_sendpack+0xa09/0xe70 mld_ifc_work+0x71c/0xdb0 process_one_work+0x9bf/0x1710 worker_thread+0x665/0x1080 kthread+0x2e4/0x3a0 ret_from_fork+0x1f/0x30 </TASK>	2025-05-01	5.5
CVE-2022-49878	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved:	2025-05-01	5.5

		bpf, verifier: Fix memory leak in array reallocation for stack state If an error (NULL) is returned by krealloc(), callers of realloc_array() were setting their allocation pointers to NULL, but on error krealloc() does not touch the original allocation. This would result in a memory resource leak. Instead, free the old allocation on the error handling path. The memory leak information is as follows as also reported by Zhengchao: unreferenced object 0xffff888019801800 (size 256): comm "bpf_repo", pid 6490, jiffies 4294959200 (age 17.170s) hex dump (first 32 bytes): 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .... 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .... backtrace: [<00000000b211474b>] __kmalloc_node_track_caller+0x45/0xc0 [<0000000086712a0b>] krealloc+0x83/0xd0 [<00000000139aab02>] realloc_array+0x82/0xe2 [<00000000b1ca41d1>] grow_stack_state+0xfb/0x186 [<00000000cd6f36d2>] check_mem_access.cold+0x141/0x1341 [<0000000081780455>] do_check_common+0x5358/0xb350 [<0000000015f6b091>] bpf_check.cold+0xc3/0x29d [<000000002973c690>] bpf_prog_load+0x13db/0x2240 [<00000000028d1644>] __sys_bpf+0x1605/0x4ce0 [<00000000053f29bd>] __x64_sys_bpf+0x75/0xb0 [<0000000056fedaf5>] do_syscall_64+0x35/0x80 [<000000002bd58261>] entry_SYSCALL_64_after_hwframe+0x63/0xcd		
CVE-2022-49880	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ext4: fix warning in 'ext4_da_release_space' Syzkaller report issue as follows: EXT4-fs (loop0): Free/Dirty block details EXT4-fs (loop0): free_blocks=0 EXT4-fs (loop0): dirty_blocks=0 EXT4-fs (loop0): Block reservation details EXT4-fs (loop0): i_reserved_data_blocks=0 EXT4-fs warning (device loop0): ext4_da_release_space:1527: ext4_da_release_space: ino 18, to_free 1 with only 0 reserved data blocks ------------[cut here]------------ WARNING: CPU: 0 PID: 92 at fs/ext4/inode.c:1528 ext4_da_release_space+0x25e/0x370 fs/ext4/inode.c:1524 Modules linked in: CPU: 0 PID: 92 Comm: kworker/u4:4 Not tainted 6.0.0-syzkaller-09423-g493ffd6605b2 #0 Hardware name: Google Google Compute Engine/Google Compute Engine, BIOS Google 09/22/2022 Workqueue: writeback wb_workfn (flush-7:0) RIP: 0010:ext4_da_release_space+0x25e/0x370 fs/ext4/inode.c:1528 RSP: 0018:ffffc900015f6c90 EFLAGS: 00010296 RAX: 42215896cd52ea00 RBX: 0000000000000000 RCX: 42215896cd52ea00 RDX: 0000000000000000 RSI: 0000000080000001 RDI: 0000000000000000 RBP: 1ffff1100e907d96 R08: ffffffff816aa79d R09: fffff520002bece5 R10: fffff520002bece5 R11: 1ffff920002bece4 R12: ffff888021fd2000 R13: ffff88807483ecb0 R14: 0000000000000001 R15: ffff88807483e740 FS: 0000000000000000(0000) GS:ffff8880b9a00000(0000) knlGS:0000000000000000 CS: 0010 DS: 0000 ES: 0000 CR0: 0000000080050033 CR2: 00005555569ba628 CR3: 000000000c88e000 CR4: 00000000003506f0 DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000 DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400 Call Trace: <TASK> ext4_es_remove_extent+0x1ab/0x260 fs/ext4/extents_status.c:1461 mpage_release_unused_pages+0x24d/0xef0 fs/ext4/inode.c:1589 ext4_writepages+0x12eb/0x3be0 fs/ext4/inode.c:2852 do_writepages+0x3c3/0x680 mm/page-writeback.c:2469 __writeback_single_inode+0xd1/0x670 fs/fs-writeback.c:1587 writeback_sb_inodes+0xb3b/0x18f0 fs/fs-writeback.c:1870 wb_writeback+0x41f/0x7b0 fs/fs-writeback.c:2044 wb_do_writeback fs/fs-writeback.c:2187 [inline] wb_workfn+0x3cb/0xef0 fs/fs-writeback.c:2227 process_one_work+0x877/0xdb0 kernel/workqueue.c:2289 worker_thread+0xb14/0x1330 kernel/workqueue.c:2436 kthread+0x266/0x300 kernel/kthread.c:376 ret_from_fork+0x1f/0x30 arch/x86/entry/entry_64.S:306 </TASK> Above issue may happens as follows: ext4_da_write_begin ext4_create_inline_data ext4_clear_inode_flag(inode, EXT4_INODE_EXTENTS);	2025-05-01	5.5

		ext4_set_inode_flag(inode, EXT4_INODE_INLINE_DATA);
		__ext4_ioctl
		ext4_ext_migrate -> will lead to eh->eh_entries not zero, and set extent flag
		ext4_da_write_begin
		ext4_da_convert_inline_data_to_extent
		ext4_da_write_inline_data_begin
		ext4_da_map_blocks
		ext4_insert_delayed_block
		if (!ext4_es_scan_clu(inode, &ext4_es_is_delonly, lblk))
		if (!ext4_es_scan_clu(inode, &ext4_es_is_mapped, lblk))
		ext4_clu_mapped(inode, EXT4_B2C(sbi, lblk)); -> will return 1
		allocated = true;
		ext4_es_insert_delayed_block(inode, lblk, allocated);
		ext4_writepages
		mpage_map_and_submit_extent(handle, &mpd, &give_up_on_write); -> return -ENOSPC
		mpage_release_unused_pages(&mpd, give_up_on_write); -> give_up_on_write == 1
		ext4_es_remove_extent
		ext4_da_release_space(inode, reserved);
		if (unlikely(to_free > ei->i_reserved_data_blocks))
		-> to_free == 1 but ei->i_reserved_data_blocks == 0
		-> then trigger warning as above
		
		To solve above issue, forbid inode do migrate which has inline data.		

CVE-2022-49881	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: wifi: cfg80211: fix memory leak in query_regdb_file() In the function query_regdb_file() the alpha2 parameter is duplicated using kmemdup() and subsequently freed in regdb_fw_cb(). However, request_firmware_nowait() can fail without calling regdb_fw_cb() and thus leak memory.	2025-05-01	5.5
CVE-2022-49885	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ACPI: APEI: Fix integer overflow in ghes_estatus_pool_init() Change num_ghes from int to unsigned int, preventing an overflow and causing subsequent vmalloc() to fail. The overflow happens in ghes_estatus_pool_init() when calculating len during execution of the statement below as both multiplication operands here are signed int: len += (num_ghes * GHES_ESOURCE_PREALLOC_MAX_SIZE); The following call trace is observed because of this bug: [9.317108] swapper/0: vmalloc error: size 18446744071562596352, exceeds total pages, mode:0xcc0(GFP_KERNEL), nodemask=(null),cpuset=/,mems_allowed=0-1 [9.317131] Call Trace: [9.317134] <TASK> [9.317137] dump_stack_lvl+0x49/0x5f [9.317145] dump_stack+0x10/0x12 [9.317146] warn_alloc.cold+0x7b/0xdf [9.317150] ? __device_attach+0x16a/0x1b0 [9.317155] __vmalloc_node_range+0x702/0x740 [9.317160] ? device_add+0x17f/0x920 [9.317164] ? dev_set_name+0x53/0x70 [9.317166] ? platform_device_add+0xf9/0x240 [9.317168] __vmalloc_node+0x49/0x50 [9.317170] ? ghes_estatus_pool_init+0x43/0xa0 [9.317176] vmalloc+0x21/0x30 [9.317177] ghes_estatus_pool_init+0x43/0xa0 [9.317179] acpi_hest_init+0x129/0x19c [9.317185] acpi_init+0x434/0x4a4 [9.317188] ? acpi_sleep_proc_init+0x2a/0x2a [9.317190] do_one_initcall+0x48/0x200 [9.317195] kernel_init_freeable+0x221/0x284 [9.317200] ? rest_init+0xe0/0xe0 [9.317204] kernel_init+0x1a/0x130 [9.317205] ret_from_fork+0x22/0x30 [9.317208] </TASK> [rjw: Subject and changelog edits]	2025-05-01	5.5
CVE-2022-49887	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: media: meson: vdec: fix possible refcount leak in vdec_probe() v4l2_device_unregister need to be called to put the refcount got by v4l2_device_register when vdec_probe fails or vdec_remove is called.	2025-05-01	5.5
CVE-2022-49889	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved:	2025-05-01	5.5

		ring-buffer: Check for NULL cpu_buffer in ring_buffer_wake_waiters() On some machines the number of listed CPUs may be bigger than the actual CPUs that exist. The tracing subsystem allocates a per_cpu directory with access to the per CPU ring buffer via a cpuX file. But to save space, the ring buffer will only allocate buffers for online CPUs, even though the CPU array will be as big as the nr_cpu_ids. With the addition of waking waiters on the ring buffer when closing the file, the ring_buffer_wake_waiters() now needs to make sure that the buffer is allocated (with the irq_work allocated with it) before trying to wake waiters, as it will cause a NULL pointer dereference. While debugging this, I added a NULL check for the buffer itself (which is OK to do), and also NULL pointer checks against buffer->buffers (which is not fine, and will WARN) as well as making sure the CPU number passed in is within the nr_cpu_ids (which is also not fine if it isn't). Bugzilla: https://bugzilla.opensuse.org/show_bug.cgi?id=1204705		
CVE-2022-49890	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: capabilities: fix potential memleak on error path from vfs_getxattr_alloc() In cap_inode_getsecurity(), we will use vfs_getxattr_alloc() to complete the memory allocation of tmpbuf, if we have completed the memory allocation of tmpbuf, but failed to call handler->get(...), there will be a memleak in below logic: |-- ret = (int)vfs_getxattr_alloc(mnt_userns, ...) | /* ^^^ alloc for tmpbuf */ |-- value = krealloc(*xattr_value, error + 1, flags) | /* ^^^ alloc memory */ |-- error = handler->get(handler, ...) | /* error! */ |-- *xattr_value = value | /* xattr_value is &tmpbuf (memory leak!) */ So we will try to free(tmpbuf) after vfs_getxattr_alloc() fails to fix it. [PM: subject line and backtrace tweaks]	2025-05-01	5.5
CVE-2022-49891	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: tracing: kprobe: Fix memory leak in test_gen_kprobe/kretprobe_cmd() test_gen_kprobe_cmd() only free buf in fail path, hence buf will leak when there is no failure. Move kfree(buf) from fail path to common path to prevent the memleak. The same reason and solution in test_gen_kretprobe_cmd(). unreferenced object 0xffff888143b14000 (size 2048): comm "insmod", pid 52490, jiffies 4301890980 (age 40.553s) hex dump (first 32 bytes): 70 3a 6b 70 72 6f 62 65 73 2f 67 65 6e 5f 6b 70 p:kprobes/gen_kp 72 6f 62 65 5f 74 65 73 74 20 64 6f 5f 73 79 73 robe_test do_sys backtrace: [<000000006d7b836b>] kmalloc_trace+0x27/0xa0 [<0000000009528b5b>] 0xffffffffa059006f [<000000008408b580>] do_one_initcall+0x87/0x2a0 [<00000000c4980a7e>] do_init_module+0xdf/0x320 [<00000000d775aad0>] load_module+0x3006/0x3390 [<00000000e9a74b80>] __do_sys_finit_module+0x113/0x1b0 [<000000003726480d>] do_syscall_64+0x35/0x80 [<000000003441e93b>] entry_SYSCALL_64_after_hwframe+0x46/0xb0	2025-05-01	5.5
CVE-2022-49894	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: cxl/region: Fix region HPA ordering validation Some regions may not have any address space allocated. Skip them when validating HPA order otherwise a crash like the following may result: devm_cxl_add_region: cxl_acpi cxl_acpi.0: decoder3.4: created region9 BUG: kernel NULL pointer dereference, address: 0000000000000000 [..] RIP: 0010:store_targetN+0x655/0x1740 [cxl_core] [..] Call Trace: <TASK> kernfs_fop_write_iter+0x144/0x200 vfs_write+0x24a/0x4d0 ksys_write+0x69/0xf0	2025-05-01	5.5

		do_syscall_64+0x3a/0x90 store_targetN+0x655/0x1740: alloc_region_ref at drivers/cxl/core/region.c:676 (inlined by) cxl_port_attach_region at drivers/cxl/core/region.c:850 (inlined by) cxl_region_attach at drivers/cxl/core/region.c:1290 (inlined by) attach_target at drivers/cxl/core/region.c:1410 (inlined by) store_targetN at drivers/cxl/core/region.c:1453		
CVE-2022-49895	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: cxl/region: Fix decoder allocation crash When an intermediate port's decoders have been exhausted by existing regions, and creating a new region with the port in question in it's hierarchical path is attempted, cxl_port_attach_region() fails to find a port decoder (as would be expected), and drops into the failure / cleanup path. However, during cleanup of the region reference, a sanity check attempts to dereference the decoder, which in the above case didn't exist. This causes a NULL pointer dereference BUG. To fix this, refactor the decoder allocation and de-allocation into helper routines, and in this 'free' routine, check that the decoder, @cxld, is valid before attempting any operations on it.	2025-05-01	5.5
CVE-2022-49896	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: cxl/pmem: Fix cxl_pmem_region and cxl_memdev leak When a cxl_nvdimm object goes through a ->remove() event (device physically removed, nvdimm-bridge disabled, or nvdimm device disabled), then any associated regions must also be disabled. As highlighted by the cxl-create-region.sh test [1], a single device may host multiple regions, but the driver was only tracking one region at a time. This leads to a situation where only the last enabled region per nvdimm device is cleaned up properly. Other regions are leaked, and this also causes cxl_memdev reference leaks. Fix the tracking by allowing cxl_nvdimm objects to track multiple region associations.	2025-05-01	5.5
CVE-2022-49899	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: fscrypt: stop using keyrings subsystem for fscrypt_master_key The approach of fs/crypto/ internally managing the fscrypt_master_key structs as the payloads of "struct key" objects contained in a "struct key" keyring has outlived its usefulness. The original idea was to simplify the code by reusing code from the keyrings subsystem. However, several issues have arisen that can't easily be resolved: - When a master key struct is destroyed, blk_crypto_evict_key() must be called on any per-mode keys embedded in it. (This started being the case when inline encryption support was added.) Yet, the keyrings subsystem can arbitrarily delay the destruction of keys, even past the time the filesystem was unmounted. Therefore, currently there is no easy way to call blk_crypto_evict_key() when a master key is destroyed. Currently, this is worked around by holding an extra reference to the filesystem's request_queue(s). But it was overlooked that the request_queue reference is *not* guaranteed to pin the corresponding blk_crypto_profile too; for device-mapper devices that support inline crypto, it doesn't. This can cause a use-after-free. - When the last inode that was using an incompletely-removed master key is evicted, the master key removal is completed by removing the key struct from the keyring. Currently this is done via key_invalidate(). Yet, key_invalidate() takes the key semaphore. This can deadlock when called from the shrinker, since in fscrypt_ioctl_add_key(), memory is allocated with GFP_KERNEL under the same semaphore. - More generally, the fact that the keyrings subsystem can arbitrarily delay the destruction of keys (via garbage collection delay, or via random processes getting temporary key references) is undesirable, as it means we can't strictly guarantee that all secrets are ever wiped. - Doing the master key lookups via the keyrings subsystem results in the key_permission LSM hook being called. fscrypt doesn't want this, as all access control for encrypted files is designed to happen via the files themselves, like any other files. The workaround which SELinux users are using is to change their SELinux policy to grant key search access to all domains. This works, but it is an odd extra step that shouldn't really have to be done.	2025-05-01	5.5

		The fix for all these issues is to change the implementation to what I should have done originally: don't use the keyrings subsystem to keep track of the filesystem's fscrypt_master_key structs. Instead, just store them in a regular kernel data structure, and rework the reference counting, locking, and lifetime accordingly. Retain support for RCU-mode key lookups by using a hash table. Replace fscrypt_sb_free() with fscrypt_sb_delete(), which releases the keys synchronously and runs a bit earlier during unmount, so that block devices are still available. A side effect of this patch is that neither the master keys themselves nor the filesystem keyrings will be listed in /proc/keys anymore. ("Master key users" and the master key users keyrings will still be listed.) However, this was mostly an implementation detail, and it was intended just for debugging purposes. I don't know of anyone using it. This patch does *not* change how "master key users" (->mk_users) works; that still uses the keyrings subsystem. That is still needed for key quotas, and changing that isn't necessary to solve the issues listed above. If we decide to change that too, it would be a separate patch. I've marked this as fixing the original commit that added the fscrypt keyring, but as noted above the most important issue that this patch fixes wasn't introduced until the addition of inline encryption support.		

CVE-2022-49901	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved:
		
		blk-mq: Fix kmemleak in blk_mq_init_allocated_queue
		
		There is a kmemleak caused by modprobe null_blk.ko
		
		unreferenced object 0xffff8881acb1f000 (size 1024):
		comm "modprobe", pid 836, jiffies 4294971190 (age 27.068s)
		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 00 53 99 9e ff ff ff ff .....S.....
		backtrace:
		[<000000004a10c249>] kmalloc_node_trace+0x22/0x60
		[<00000000648f7950>] blk_mq_alloc_and_init_hctx+0x289/0x350
		[<00000000af06de0e>] blk_mq_realloc_hw_ctxs+0x2fe/0x3d0
		[<00000000e00c1872>] blk_mq_init_allocated_queue+0x48c/0x1440
		[<00000000d16b4e68>] __blk_mq_alloc_disk+0xc8/0x1c0
		[<00000000d10c98c3>] 0xffffffffc450d69d
		[<00000000b9299f48>] 0xffffffffc4538392
		[<0000000061c39ed6>] do_one_initcall+0xd0/0x4f0
		[<00000000b389383b>] do_init_module+0x1a4/0x680
		[<0000000087cf3542>] load_module+0x6249/0x7110
		[<00000000beba61b8>] __do_sys_finit_module+0x140/0x200
		[<00000000fdcfff51>] do_syscall_64+0x35/0x80
		[<000000003c0f1f71>] entry_SYSCALL_64_after_hwframe+0x46/0xb0
		
		That is because q->ma_ops is set to NULL before blk_release_queue is called.
		
		blk_mq_init_queue_data
		blk_mq_init_allocated_queue
		blk_mq_realloc_hw_ctxs
		for (i = 0; i < set->nr_hw_queues; i++) {
		old_hctx = xa_load(&q->hctx_table, i);
		if (!blk_mq_alloc_and_init_hctx(.., i, ..)) [1]
		if (!old_hctx)
		break;
		
		xa_for_each_start(&q->hctx_table, j, hctx, j)
		blk_mq_exit_hctx(q, set, hctx, j); [2]
		
		if (!q->nr_hw_queues) [3]
		goto err_hctxs;
		
		err_exit:
		q->mq_ops = NULL; [4]
		
		blk_put_queue
		blk_release_queue
		if (queue_is_mq(q)) [5]
		blk_mq_release(q);
		
		[1]: blk_mq_alloc_and_init_hctx failed at i != 0.
		[2]: The hctxs allocated by [1] are moved to q->unused_hctx_list and will be cleaned up in blk_mq_release.
		[3]: q->nr_hw_queues is 0.
		[4]: Set q->mq_ops to NULL.
			2025-05-01	5.5

		[5]: queue_is_mq returns false due to [4]. And blk_mq_release will not be called. The hctxs in q->unused_hctx_list are leaked. To fix it, call blk_release_queue in exception path.		
CVE-2022-49902	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: block: Fix possible memory leak for rq_wb on add_disk failure kmemleak reported memory leaks in device_add_disk(): kmemleak: 3 new suspected memory leaks unreferenced object 0xffff88800f420800 (size 512): comm "modprobe", pid 4275, jiffies 4295639067 (age 223.512s) hex dump (first 32 bytes): 04 00 00 00 08 00 00 00 01 00 00 00 00 00 00 00 .... 00 e1 f5 05 00 00 00 00 00 00 00 00 00 00 00 00 .... backtrace: [<00000000d3662699>] kmalloc_trace+0x26/0x60 [<00000000edc7aadc>] wbt_init+0x50/0x6f0 [<0000000069601d16>] wbt_enable_default+0x157/0x1c0 [<0000000028fc393f>] blk_register_queue+0x2a4/0x420 [<000000007345a042>] device_add_disk+0x6fd/0xe40 [<0000000060e6aab0>] nbd_dev_add+0x828/0xbf0 [nbd] ... It is because the memory allocated in wbt_enable_default() is not released in device_add_disk() error path. Normally, these memory are freed in: del_gendisk() rq_qos_exit() rqos->ops->exit(rqos); wbt_exit() So rq_qos_exit() is called to free the rq_wb memory for wbt_init(). However in the error path of device_add_disk(), only blk_unregister_queue() is called and make rq_wb memory leaked. Add rq_qos_exit() to the error path to fix it.	2025-05-01	5.5
CVE-2022-49904	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net, neigh: Fix null-ptr-deref in neigh_table_clear() When IPv6 module gets initialized but hits an error in the middle, kenel panic with: KASAN: null-ptr-deref in range [0x0000000000000598-0x000000000000059f] CPU: 1 PID: 361 Comm: insmod Hardware name: QEMU Standard PC (i440FX + PIIX, 1996) RIP: 0010:__neigh_ifdown.isra.0+0x24b/0x370 RSP: 0018:ffff888012677908 EFLAGS: 00000202 ... Call Trace: <TASK> neigh_table_clear+0x94/0x2d0 ndisc_cleanup+0x27/0x40 [ipv6] inet6_init+0x21c/0x2cb [ipv6] do_one_initcall+0xd3/0x4d0 do_init_module+0x1ae/0x670 ... Kernel panic - not syncing: Fatal exception When ipv6 initialization fails, it will try to cleanup and calls: neigh_table_clear() neigh_ifdown(tbl, NULL) pneigh_queue_purge(&tbl->proxy_queue, dev_net(dev == NULL)) # dev_net(NULL) triggers null-ptr-deref. Fix it by passing NULL to pneigh_queue_purge() in neigh_ifdown() if dev is NULL, to make kernel not panic immediately.	2025-05-01	5.5
CVE-2022-49906	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: ibmvnic: Free rwi on reset success Free the rwi structure in the event that the last rwi in the list processed successfully. The logic in commit 4f408e1fa6e1 ("ibmvnic: retry reset if there are no other resets") introduces an issue that results in a 32 byte memory leak whenever the last rwi in the list gets processed.	2025-05-01	5.5

CVE-2022-49908	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: Bluetooth: L2CAP: Fix memory leak in vhci_write Syzkaller reports a memory leak as follows: ========== BUG: memory leak unreferenced object 0xffff88810d81ac00 (size 240): [...] hex dump (first 32 bytes): 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .... 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .... backtrace: [<ffffffff838733d9>] __alloc_skb+0x1f9/0x270 net/core/skbuff.c:418 [<ffffffff833f742f>] alloc_skb include/linux/skbuff.h:1257 [inline] [<ffffffff833f742f>] bt_skb_alloc include/net/bluetooth/bluetooth.h:469 [inline] [<ffffffff833f742f>] vhci_get_user drivers/bluetooth/hci_vhci.c:391 [inline] [<ffffffff833f742f>] vhci_write+0x5f/0x230 drivers/bluetooth/hci_vhci.c:511 [<ffffffff815e398d>] call_write_iter include/linux/fs.h:2192 [inline] [<ffffffff815e398d>] new_sync_write fs/read_write.c:491 [inline] [<ffffffff815e398d>] vfs_write+0x42d/0x540 fs/read_write.c:578 [<ffffffff815e3cdd>] ksys_write+0x9d/0x160 fs/read_write.c:631 [<ffffffff845e0645>] do_syscall_x64 arch/x86/entry/common.c:50 [inline] [<ffffffff845e0645>] do_syscall_64+0x35/0xb0 arch/x86/entry/common.c:80 [<ffffffff84600087>] entry_SYSCALL_64_after_hwframe+0x63/0xcd ========== HCI core will uses hci_rx_work() to process frame, which is queued to the hdev->rx_q tail in hci_recv_frame() by HCI driver. Yet the problem is that, HCI core may not free the skb after handling ACL data packets. To be more specific, when start fragment does not contain the L2CAP length, HCI core just copies skb into conn->rx_skb and finishes frame process in l2cap_recv_acldata(), without freeing the skb, which triggers the above memory leak. This patch solves it by releasing the relative skb, after processing the above case in l2cap_recv_acldata().	2025-05-01	5.5
CVE-2022-49915	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: mISDN: fix possible memory leak in mISDN_register_device() Afer commit 1fa5ae857bb1 ("driver core: get rid of struct device's bus_id string array"), the name of device is allocated dynamically, add put_device() to give up the reference, so that the name can be freed in kobject_cleanup() when the refcount is 0. Set device class before put_device() to avoid null release() function WARN message in device_release().	2025-05-01	5.5
CVE-2022-49916	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: rose: Fix NULL pointer dereference in rose_send_frame() The syzkaller reported an issue: KASAN: null-ptr-deref in range [0x0000000000000380-0x0000000000000387] CPU: 0 PID: 4069 Comm: kworker/0:15 Not tainted 6.0.0-syzkaller-02734-g0326074ff465 #0 Hardware name: Google Google Compute Engine/Google Compute Engine, BIOS Google 09/22/2022 Workqueue: rcu_gp srcu_invoke_callbacks RIP: 0010:rose_send_frame+0x1dd/0x2f0 net/rose/rose_link.c:101 Call Trace: <IRQ> rose_transmit_clear_request+0x1d5/0x290 net/rose/rose_link.c:255 rose_rx_call_request+0x4c0/0x1bc0 net/rose/af_rose.c:1009 rose_loopback_timer+0x19e/0x590 net/rose/rose_loopback.c:111 call_timer_fn+0x1a0/0x6b0 kernel/time/timer.c:1474 expire_timers kernel/time/timer.c:1519 [inline] __run_timers.part.0+0x674/0xa80 kernel/time/timer.c:1790 __run_timers kernel/time/timer.c:1768 [inline] run_timer_softirq+0xb3/0x1d0 kernel/time/timer.c:1803 __do_softirq+0x1d0/0x9c8 kernel/softirq.c:571 [...] </IRQ> It triggers NULL pointer dereference when 'neigh->dev->dev_addr' is called in the rose_send_frame(). It's the first occurrence of the `neigh` is in rose_loopback_timer() as `rose_loopback_neigh', and the 'dev' in 'rose_loopback_neigh' is initialized sa nullptr. It had been fixed by commit 3b3fd068c56e3fbea30090859216a368398e39bf ("rose: Fix Null pointer dereference in rose_send_frame()") ever. But it's introduced by commit 3c53cd65dece47dd1f9d3a809f32e59d1d87b2b8	2025-05-01	5.5

		("rose: check NULL rose_loopback_neigh->loopback") again. We fix it by add NULL check in rose_transmit_clear_request(). When the 'dev' in 'neigh' is NULL, we don't reply the request and just clear it. syzkaller don't provide repro, and I provide a syz repro like: r0 = syz_init_net_socket$bt_sco(0x1f, 0x5, 0x2) ioctl$sock_inet_SIOCSIFFLAGS(r0, 0x8914, &(0x7f0000000180)={'rose0\x00', 0x201}) r1 = syz_init_net_socket$rose(0xb, 0x5, 0x0) bind$rose(r1, &(0x7f00000000c0)=@full={0xb, @dev, @null, 0x0, [@null, @null, @netrom, @netrom, @default, @null]}, 0x40) connect$rose(r1, &(0x7f0000000240)=@short={0xb, @dev={0xbb, 0xbb, 0xbb, 0x1, 0x0}, @remote={0xcc, 0xcc, 0xcc, 0xcc, 0xcc, 0xcc, 0x1}, 0x1, @netrom={0xbb, 0xbb, 0xbb, 0xbb, 0xbb, 0x0, 0x0}}, 0x1c)		
CVE-2022-49922	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: nfc: nfcmrvl: Fix potential memory leak in nfcmrvl_i2c_nci_send() nfcmrvl_i2c_nci_send() will be called by nfcmrvl_nci_send(), and skb should be freed in nfcmrvl_i2c_nci_send(). However, nfcmrvl_nci_send() will only free skb when i2c_master_send() return >=0, which means skb will memleak when i2c_master_send() failed. Free skb no matter whether i2c_master_send() succeeds.	2025-05-01	5.5
CVE-2022-49923	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: nfc: nxp-nci: Fix potential memory leak in nxp_nci_send() nxp_nci_send() will call nxp_nci_i2c_write(), and only free skb when nxp_nci_i2c_write() failed. However, even if the nxp_nci_i2c_write() run succeeds, the skb will not be freed in nxp_nci_i2c_write(). As the result, the skb will memleak. nxp_nci_send() should also free the skb when nxp_nci_i2c_write() succeeds.	2025-05-01	5.5
CVE-2022-49924	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: nfc: fdp: Fix potential memory leak in fdp_nci_send() fdp_nci_send() will call fdp_nci_i2c_write that will not free skb in the function. As a result, when fdp_nci_i2c_write() finished, the skb will memleak. fdp_nci_send() should free skb after fdp_nci_i2c_write() finished.	2025-05-01	5.5
CVE-2022-49925	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: RDMA/core: Fix null-ptr-deref in ib_core_cleanup() KASAN reported a null-ptr-deref error: KASAN: null-ptr-deref in range [0x0000000000000118-0x000000000000011f] CPU: 1 PID: 379 Hardware name: QEMU Standard PC (i440FX + PIIX, 1996) RIP: 0010:destroy_workqueue+0x2f/0x740 RSP: 0018:ffff888016137df8 EFLAGS: 00000202 ... Call Trace: ib_core_cleanup+0xa/0xa1 [ib_core] __do_sys_delete_module.constprop.0+0x34f/0x5b0 do_syscall_64+0x3a/0x90 entry_SYSCALL_64_after_hwframe+0x63/0xcd RIP: 0033:0x7fa1a0d221b7 ... It is because the fail of roce_gid_mgmt_init() is ignored: ib_core_init() roce_gid_mgmt_init() gid_cache_wq = alloc_ordered_workqueue # fail ... ib_core_cleanup() roce_gid_mgmt_cleanup() destroy_workqueue(gid_cache_wq) # destroy an unallocated wq Fix this by catching the fail of roce_gid_mgmt_init() in ib_core_init().	2025-05-01	5.5
CVE-2022-49926	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: net: dsa: Fix possible memory leaks in dsa_loop_init() kmemleak reported memory leaks in dsa_loop_init(): kmemleak: 12 new suspected memory leaks	2025-05-01	5.5

		unreferenced object 0xffff8880138ce000 (size 2048):
		comm "modprobe", pid 390, jiffies 4295040478 (age 238.976s)
		backtrace:
		[<000000006a94f1d5>] kmalloc_trace+0x26/0x60
		[<00000000a9c44622>] phy_device_create+0x5d/0x970
		[<00000000d0ee2afc>] get_phy_device+0xf3/0x2b0
		[<00000000dca0c71f>] __fixed_phy_register.part.0+0x92/0x4e0
		[<000000008a834798>] fixed_phy_register+0x84/0xb0
		[<0000000055223fcb>] dsa_loop_init+0xa9/0x116 [dsa_loop]
		...
		
		There are two reasons for memleak in dsa_loop_init().
		
		First, fixed_phy_register() create and register phy_device:
		
		fixed_phy_register()
		get_phy_device()
		phy_device_create() # freed by phy_device_free()
		phy_device_register() # freed by phy_device_remove()
		
		But fixed_phy_unregister() only calls phy_device_remove().
		So the memory allocated in phy_device_create() is leaked.
		
		Second, when mdio_driver_register() fail in dsa_loop_init(),
		it just returns and there is no cleanup for phydevs.
		
		Fix the problems by catching the error of mdio_driver_register()
		in dsa_loop_init(), then calling both fixed_phy_unregister() and
		phy_device_free() to release phydevs.
		Also add a function for phydevs cleanup to avoid duplacate.		

CVE-2022-49927	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: nfs4: Fix kmemleak when allocate slot failed If one of the slot allocate failed, should cleanup all the other allocated slots, otherwise, the allocated slots will leak: unreferenced object 0xffff8881115aa100 (size 64): comm ""mount.nfs"", pid 679, jiffies 4294744957 (age 115.037s) hex dump (first 32 bytes): 00 cc 19 73 81 88 ff ff 00 a0 5a 11 81 88 ff ff ...s.....Z..... 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .... backtrace: [<000000007a4c434a>] nfs4_find_or_create_slot+0x8e/0x130 [<00000005472a39c>] nfs4_realloc_slot_table+0x23f/0x270 [<00000000cd8ca0eb>] nfs40_init_client+0x4a/0x90 [<00000000128486db>] nfs4_init_client+0xce/0x270 [<000000008d2cacad>] nfs4_set_client+0x1a2/0x2b0 [<000000000e593b52>] nfs4_create_server+0x300/0x5f0 [<00000000e4425dd2>] nfs4_try_get_tree+0x65/0x110 [<00000000d3a6176f>] vfs_get_tree+0x41/0xf0 [<0000000016b5ad4c>] path_mount+0x9b3/0xdd0 [<00000000494cae71>] __x64_sys_mount+0x190/0x1d0 [<000000005d56bdec>] do_syscall_64+0x35/0x80 [<00000000687c9ae4>] entry_SYSCALL_64_after_hwframe+0x46/0xb0	2025-05-01	5.5
CVE-2022-49928	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: SUNRPC: Fix null-ptr-deref when xps sysfs alloc failed There is a null-ptr-deref when xps sysfs alloc failed: BUG: KASAN: null-ptr-deref in sysfs_do_create_link_sd+0x40/0xd0 Read of size 8 at addr 0000000000000030 by task gssproxy/457 CPU: 5 PID: 457 Comm: gssproxy Not tainted 6.0.0-09040-g02357b27ee03 #9 Call Trace: <TASK> dump_stack_lvl+0x34/0x44 kasan_report+0xa3/0x120 sysfs_do_create_link_sd+0x40/0xd0 rpc_sysfs_client_setup+0x161/0x1b0 rpc_new_client+0x3fc/0x6e0 rpc_create_xprt+0x71/0x220 rpc_create+0x1d4/0x350 gssp_rpc_create+0xc3/0x160 set_gssp_clnt+0xbc/0x140 write_gssp+0x116/0x1a0 proc_reg_write+0xd6/0x130 vfs_write+0x177/0x690 ksys_write+0xb9/0x150 do_syscall_64+0x35/0x80 entry_SYSCALL_64_after_hwframe+0x46/0xb0	2025-05-01	5.5

		When the xprt_switch sysfs alloc failed, should not add xprt and switch sysfs to it, otherwise, maybe null-ptr-deref; also initialize the 'xps_sysfs' to NULL to avoid oops when destroy it.		
CVE-2022-49930	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: RDMA/hns: Fix NULL pointer problem in free_mr_init() Lock grab occurs in a concurrent scenario, resulting in stepping on a NULL pointer. It should be init mutex_init() first before use the lock. Unable to handle kernel NULL pointer dereference at virtual address 0000000000000000 Call trace: __mutex_lock.constprop.0+0xd0/0x5c0 __mutex_lock_slowpath+0x1c/0x2c mutex_lock+0x44/0x50 free_mr_send_cmd_to_hw+0x7c/0x1c0 [hns_roce_hw_v2] hns_roce_v2_dereg_mr+0x30/0x40 [hns_roce_hw_v2] hns_roce_dereg_mr+0x4c/0x130 [hns_roce_hw_v2] ib_dereg_mr_user+0x54/0x124 uverbs_free_mr+0x24/0x30 destroy_hw_idr_uobject+0x38/0x74 uverbs_destroy_uobject+0x48/0x1c4 uobj_destroy+0x74/0xcc ib_uverbs_cmd_verbs+0x368/0xbb0 ib_uverbs_ioctl+0xec/0x1a4 __arm64_sys_ioctl+0xb4/0x100 invoke_syscall+0x50/0x120 el0_svc_common.constprop.0+0x58/0x190 do_el0_svc+0x30/0x90 el0_svc+0x2c/0xb4 el0t_64_sync_handler+0x1a4/0x1b0 el0t_64_sync+0x19c/0x1a0	2025-05-01	5.5
CVE-2022-49931	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: IB/hfi1: Correctly move list in sc_disable() Commit 13bac861952a ("IB/hfi1: Fix abba locking issue with sc_disable()") incorrectly tries to move a list from one list head to another. The result is a kernel crash. The crash is triggered when a link goes down and there are waiters for a send to complete. The following signature is seen: BUG: kernel NULL pointer dereference, address: 0000000000000030 [...] Call Trace: sc_disable+0x1ba/0x240 [hfi1] pio_freeze+0x3d/0x60 [hfi1] handle_freeze+0x27/0x1b0 [hfi1] process_one_work+0x1b0/0x380 ? process_one_work+0x380/0x380 worker_thread+0x30/0x360 ? process_one_work+0x380/0x380 kthread+0xd7/0x100 ? kthread_complete_and_exit+0x20/0x20 ret_from_fork+0x1f/0x30 The fix is to use the correct call to move the list.	2025-05-01	5.5
CVE-2025-24271	apple - multiple products	An access issue was addressed with improved access restrictions. This issue is fixed in macOS Sequoia 15.4, tvOS 18.4, macOS Ventura 13.7.5, iPadOS 17.7.6, macOS Sonoma 14.7.5, iOS 18.4 and iPadOS 18.4, visionOS 2.4. An unauthenticated user on the same network as a signed-in Mac could send it AirPlay commands without pairing.	2025-04-29	5.4
CVE-2025-3910	red hat - multiple products	A flaw was found in Keycloak. The org.keycloak.authorization package may be vulnerable to circumventing required actions, allowing users to circumvent requirements such as setting up two-factor authentication.	2025-04-29	5.4
CVE-2025-3891	red hat - multiple products	A flaw was found in the mod_auth_openidc module for Apache httpd. This flaw allows a remote, unauthenticated attacker to trigger a denial of service by sending an empty POST request when the OIDCPreservePost directive is enabled. The server crashes consistently, affecting availability.	2025-04-29	5.3
CVE-2025-4121	netgear - JWNR2000v2	A vulnerability was found in Netgear JWNR2000v2 1.0.0.11. It has been declared as critical. Affected by this vulnerability is the function cmd_wireless. The manipulation of the argument host leads to command injection. The attack can be launched remotely. The vendor was contacted early about this disclosure but did not respond in any way.	2025-04-30	5.3
CVE-2025-4122	netgear - JWNR2000v2	A vulnerability was found in Netgear JWNR2000v2 1.0.0.11. It has been rated as critical. Affected by this issue is the function sub_435E04. The manipulation of the argument host leads to command injection. The attack may be launched remotely. The vendor was contacted early about this disclosure but did not respond in any way.	2025-04-30	5.3
CVE-2025-4135	netgear - WG302v2	A vulnerability was found in Netgear WG302v2 up to 5.2.9 and classified as critical. Affected by this issue is the function ui_get_input_value. The manipulation of the argument host leads to	2025-04-30	5.3

		command injection. The attack may be launched remotely. The vendor was contacted early about this disclosure but did not respond in any way.		
CVE-2024-52903	ibm - Db2 for Linux, UNIX and Windows	IBM Db2 for Linux, UNIX and Windows 12.1.0 and 12.1.1 is vulnerable to a denial of service as the server may crash under certain conditions with a specially crafted query.	2025-05-01	5.3
CVE-2024-55913	ibm - Concert Software	IBM Concert Software 1.0.0 through 1.0.5 could allow a remote attacker to traverse directories on the system. An attacker could send a specially crafted URL request containing "dot dot" sequences (/../) to view arbitrary files on the system.	2025-05-02	5.3
CVE-2025-3911	docker - Docker Desktop	Recording of environment variables, configured for running containers, in Docker Desktop application logs could lead to unintentional disclosure of sensitive information such as api keys, passwords, etc. A malicious actor with read access to these logs could obtain sensitive credentials information and further use it to gain unauthorized access to other systems. Starting with version 4.41.0, Docker Desktop no longer logs environment variables set by the user.	2025-04-29	5.2
CVE-2025-4089	mozilla - multiple products	Due to insufficient escaping of special characters in the "copy as cURL" feature, an attacker could trick a user into using this command, potentially leading to local code execution on the user's system. This vulnerability affects Firefox < 138 and Thunderbird < 138.	2025-04-29	5.1
CVE-2025-4117	netgear - JWNR2000v2	A vulnerability, which was classified as critical, was found in Netgear JWNR2000v2 1.0.0.11. This affects the function sub_41A914. The manipulation of the argument host leads to buffer overflow. The vendor was contacted early about this disclosure but did not respond in any way.	2025-04-30	5.1
CVE-2025-0716	google - AngularJS	Improper sanitization of the value of the 'href' and 'xlink:href' attributes in '<image>' SVG elements in AngularJS allows attackers to bypass common image source restrictions. This can lead to a form of Content Spoofing https://owasp.org/www-community/attacks/Content_Spoofing and also negatively affect the application's performance and behavior by using too large or slow-to-load images. This issue affects all versions of AngularJS. Note: The AngularJS project is End-of-Life and will not receive any updates to address this issue. For more information see here https://docs.angularjs.org/misc/version-support-status .	2025-04-29	4.8
CVE-2022-49920	linux - multiple products	In the Linux kernel, the following vulnerability has been resolved: netfilter: nf_tables: netlink notifier might race to release objects commit release path is invoked via call_rcu and it runs lockless to release the objects after rcu grace period. The netlink notifier handler might win race to remove objects that the transaction context is still referencing from the commit release path. Call rcu_barrier() to ensure pending rcu callbacks run to completion if the list of transactions to be destroyed is not empty.	2025-05-01	4.7
CVE-2025-4035	red hat - multiple products	A flaw was found in libsoup. When handling cookies, libsoup clients mistakenly allow cookies to be set for public suffix domains if the domain contains at least two components and includes an uppercase character. This bypasses public suffix protections and could allow a malicious website to set cookies for domains it does not own, potentially leading to integrity issues such as session fixation.	2025-04-29	4.3
CVE-2025-4095	docker - Docker Desktop	Registry Access Management (RAM) is a security feature allowing administrators to restrict access for their developers to only allowed registries. When a MacOS configuration profile is used to enforce organization sign-in, the RAM policies are not being applied, which would allow Docker Desktop users to pull down unapproved, and potentially malicious images from any registry.	2025-04-29	4.3
CVE-2025-1495	ibm - IBM Business Automation Workflow	IBM Business Automation Workflow 24.0.0 and 24.0.1 through 24.0.1 IF001 Center may leak sensitive information due to missing authorization validation.	2025-05-03	4.3
CVE-2025-23377	dell - PowerProtect Data Manager	Dell PowerProtect Data Manager Reporting, version(s) 19.17, 19.18 contain(s) an Improper Encoding or Escaping of Output vulnerability. A high privileged attacker with local access could potentially exploit this vulnerability to inject arbitrary web script or html in reporting outputs.	2025-04-28	4.2
CVE-2025-47229	gnu - PSPP	libpspp-core.a in GNU PSPP through 2.0.1 allows attackers to cause a denial of service (var_set_leave_quiet assertion failure and application exit) via crafted input data, such as data that triggers a call from src/data/dictionary.c code into src/data/variable.c code.	2025-05-03	2.9
CVE-2025-23376	dell - PowerProtect Data Manager Reporting	Dell PowerProtect Data Manager Reporting, version(s) 19.16, 19.17, 19.18, contain(s) an Improper Neutralization of Special Elements Used in a Template Engine vulnerability. A high privileged attacker with local access could potentially exploit this vulnerability, leading to information disclosure.	2025-04-28	2.3

Where NCA provides the vulnerability information as published by NIST's NVD. In addition, it is the entity's or individual's responsibility to ensure the implementation of appropriate recommendations.

وحيث تقدم الهيئة تفاصيل الثغرات كما تم نشرها من قبل NIST's NVD. وإذ تبقى مسؤولية الجهة أو الشخص قائمة للتأكد من تطبيق التوصيات المناسبة.