```
commit db0655e705be645ad673b0a70160921e088517c0
Author: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
Date:   Tue Oct 29 09:22:48 2019 +0100

    Linux 5.3.8

commit bbe837675455975d4a52ccfcc47006107b12a893
Author: Greg KH <gregkh@linuxfoundation.org>
Date:   Tue Oct 1 18:56:11 2019 +0200

    RDMA/cxgb4: Do not dma memory off of the stack

    commit 3840c5b78803b2b6cc1ff820100a74a092c40cbb upstream.

    Nicolas pointed out that the cxgb4 driver is doing dma off of the stack,
    which is generally considered a very bad thing.  On some architectures it
    could be a security problem, but odds are none of them actually run this
    driver, so it's just a "normal" bug.

    Resolve this by allocating the memory for a message off of the heap
    instead of the stack.  kmalloc() always will give us a proper memory
    location that DMA will work correctly from.

    Link: https://lore.kernel.org/r/20191001165611.GA3542072@kroah.com
    Reported-by: Nicolas Waisman <nico@semmle.com>
    Tested-by: Potnuri Bharat Teja <bharat@chelsio.com>
    Signed-off-by: Jason Gunthorpe <jgg@mellanox.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 37b4a8252dfd6060d2ffe9beecab4e58330d62a3
Author: Tejun Heo <tj@kernel.org>
Date:   Tue Oct 15 08:49:27 2019 -0700

    blk-rq-qos: fix first node deletion of rq_qos_del()

    commit 307f4065b9d7c1e887e8bdfb2487e4638559fea1 upstream.

    rq_qos_del() incorrectly assigns the node being deleted to the head if
    it was the first on the list in the !prev path.  Fix it by iterating
    with ** instead.

    Signed-off-by: Tejun Heo <tj@kernel.org>
    Cc: Josef Bacik <josef@toxicpanda.com>
    Fixes: a79050434b45 ("blk-rq-qos: refactor out common elements of blk-wbt")
    Cc: stable@vger.kernel.org # v4.19+
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit d07e3066ffa045e7ec5f81ca67951b03f3bfb97c
Author: Chris Goldsworthy <cgoldswo@codeaurora.org>
Date:   Sat Oct 19 18:57:24 2019 -0700

    of: reserved_mem: add missing of_node_put() for proper ref-counting

    commit 5dba51754b04a941a1064f584e7a7f607df3f9bc upstream.

    Commit d698a388146c ("of: reserved-memory: ignore disabled memory-region
    nodes") added an early return in of_reserved_mem_device_init_by_idx(), but
    didn't call of_node_put() on a device_node whose ref-count was incremented
    in the call to of_parse_phandle() preceding the early exit.

    Fixes: d698a388146c ("of: reserved-memory: ignore disabled memory-region nodes")
    Signed-off-by: Chris Goldsworthy <cgoldswo@codeaurora.org>
    Cc: stable@vger.kernel.org
```

```
    Reviewed-by: Bjorn Andersson <bjorn.andersson@linaro.org>
    Signed-off-by: Rob Herring <robh@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 99f8ef99333f734f242b8dc0de44a474ac6a8b52
Author: Viresh Kumar <viresh.kumar@linaro.org>
Date:   Thu Oct 10 15:55:33 2019 +0530

    opp: of: drop incorrect lockdep_assert_held()

    commit f2edbb6699b0bc6e4f789846b99007200546c6c2 upstream.

    _find_opp_of_np() doesn't traverse the list of OPP tables but instead
    just the entries within an OPP table and so only requires to lock the
    OPP table itself.

    The lockdep_assert_held() was added there by mistake and isn't really
    required.

    Fixes: 5d6d106fa455 ("OPP: Populate required opp tables from "required-opps" property")
    Cc: v5.0+ <stable@vger.kernel.org> # v5.0+
    Reported-by: Niklas Cassel <niklas.cassel@linaro.org>
    Signed-off-by: Viresh Kumar <viresh.kumar@linaro.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e00907058806754b5b6548b57f3bd5f32bfdb34c
Author: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
Date:   Mon Oct 14 13:25:00 2019 +0200

    PCI: PM: Fix pci_power_up()

    commit 45144d42f299455911cc29366656c7324a3a7c97 upstream.

    There is an arbitrary difference between the system resume and
    runtime resume code paths for PCI devices regarding the delay to
    apply when switching the devices from D3cold to D0.

    Namely, pci_restore_standard_config() used in the runtime resume
    code path calls pci_set_power_state() which in turn invokes
    __pci_start_power_transition() to power up the device through the
    platform firmware and that function applies the transition delay
    (as per PCI Express Base Specification Revision 2.0, Section 6.6.1).
    However, pci_pm_default_resume_early() used in the system resume
    code path calls pci_power_up() which doesn't apply the delay at
    all and that causes issues to occur during resume from
    suspend-to-idle on some systems where the delay is required.

    Since there is no reason for that difference to exist, modify
    pci_power_up() to follow pci_set_power_state() more closely and
    invoke __pci_start_power_transition() from there to call the
    platform firmware to power up the device (in case that's necessary).

    Fixes: db288c9c5f9d ("PCI / PM: restore the original behavior of pci_set_power_state()")
    Reported-by: Daniel Drake <drake@endlessm.com>
    Tested-by: Daniel Drake <drake@endlessm.com>
    Link: https://lore.kernel.org/linux-
pm/CAD8Lp44TYxrMgPLkHCqF9hv6smEurMXvmmvmtyFhZ6Q4SE+dig@mail.gmail.com/T/#m21be74af263c6a34f3
6e0fc5c77c5449d9406925
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Acked-by: Bjorn Helgaas <bhelgaas@google.com>
    Cc: 3.10+ <stable@vger.kernel.org> # 3.10+
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e8dc486e861dc824cf3b851c4bded331bef998f2
Author: Juergen Gross <jgross@suse.com>
```

```
Date:   Fri Oct 18 09:45:49 2019 +0200

    xen/netback: fix error path of xenvif_connect_data()

    commit 3d5c1a037d37392a6859afbde49be5ba6a70a6b3 upstream.

    xenvif_connect_data() calls module_put() in case of error. This is
    wrong as there is no related module_get().

    Remove the superfluous module_put().

    Fixes: 279f438e36c0a7 ("xen-netback: Don't destroy the netdev until the vif is shut
down")
    Cc: <stable@vger.kernel.org> # 3.12
    Signed-off-by: Juergen Gross <jgross@suse.com>
    Reviewed-by: Paul Durrant <paul@xen.org>
    Reviewed-by: Wei Liu <wei.liu@kernel.org>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f55937689844659bd04e754ccd1768611a65fba0
Author: Jeff Layton <jlayton@kernel.org>
Date:   Thu Sep 26 16:05:11 2019 -0400

    ceph: just skip unrecognized info in ceph_reply_info_extra

    commit 1d3f87233e26362fc3d4e59f0f31a71b570f90b9 upstream.

    In the future, we're going to want to extend the ceph_reply_info_extra
    for create replies. Currently though, the kernel code doesn't accept an
    extra blob that is larger than the expected data.

    Change the code to skip over any unrecognized fields at the end of the
    extra blob, rather than returning -EIO.

    Cc: stable@vger.kernel.org
    Signed-off-by: Jeff Layton <jlayton@kernel.org>
    Signed-off-by: Ilya Dryomov <idryomov@gmail.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit cb4b4601f910c78d2b49f637a12ef98b41cb76a9
Author: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
Date:   Wed Oct 9 01:29:10 2019 +0200

    cpufreq: Avoid cpufreq_suspend() deadlock on system shutdown

    commit 65650b35133ff20f0c9ef0abd5c3c66dbce3ae57 upstream.

    It is incorrect to set the cpufreq syscore shutdown callback pointer
    to cpufreq_suspend(), because that function cannot be run in the
    syscore stage of system shutdown for two reasons: (a) it may attempt
    to carry out actions depending on devices that have already been shut
    down at that point and (b) the RCU synchronization carried out by it
    may not be able to make progress then.

    The latter issue has been present since commit 45975c7d21a1 ("rcu:
    Define RCU-sched API in terms of RCU for Tree RCU PREEMPT builds"),
    but the former one has been there since commit 90de2a4aa9f3 ("cpufreq:
    suspend cpufreq governors on shutdown") regardless.

    Fix that by dropping cpufreq_syscore_ops altogether and making
    device_shutdown() call cpufreq_suspend() directly before shutting
    down devices, which is along the lines of what system-wide power
    management does.
```

```
    Fixes: 45975c7d21a1 ("rcu: Define RCU-sched API in terms of RCU for Tree RCU PREEMPT
builds")
    Fixes: 90de2a4aa9f3 ("cpufreq: suspend cpufreq governors on shutdown")
    Reported-by: Ville Syrjälä <ville.syrjala@linux.intel.com>
    Tested-by: Ville Syrjälä <ville.syrjala@linux.intel.com>
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Acked-by: Viresh Kumar <viresh.kumar@linaro.org>
    Cc: 4.0+ <stable@vger.kernel.org> # 4.0+
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 80f59f36cb2caea3a5342c3361eb1782e7c24b23
Author: Christophe JAILLET <christophe.jaillet@wanadoo.fr>
Date:   Sat Oct 5 13:21:01 2019 +0200

    memstick: jmb38x_ms: Fix an error handling path in 'jmb38x_ms_probe()'

    commit 28c9fac09ab0147158db0baeec630407a5e9b892 upstream.

    If 'jmb38x_ms_count_slots()' returns 0, we must undo the previous
    'pci_request_regions()' call.

    Goto 'err_out_int' to fix it.

    Fixes: 60fdd931d577 ("memstick: add support for JMicron jmb38x MemoryStick host
controller")
    Cc: stable@vger.kernel.org
    Signed-off-by: Christophe JAILLET <christophe.jaillet@wanadoo.fr>
    Signed-off-by: Ulf Hansson <ulf.hansson@linaro.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 21b3ad2f5726eddde4176b6240f5cc8081189cfe
Author: Greg Kurz <groug@kaod.org>
Date:   Fri Sep 27 13:53:43 2019 +0200

    KVM: PPC: Book3S HV: XIVE: Ensure VP isn't already in use

    commit 12ade69c1eb9958b13374edf5ef742ea20ccffde upstream.

    Connecting a vCPU to a XIVE KVM device means establishing a 1:1
    association between a vCPU id and the offset (VP id) of a VP
    structure within a fixed size block of VPs. We currently try to
    enforce the 1:1 relationship by checking that a vCPU with the
    same id isn't already connected. This is good but unfortunately
    not enough because we don't map VP ids to raw vCPU ids but to
    packed vCPU ids, and the packing function kvmppc_pack_vcpu_id()
    isn't bijective by design. We got away with it because QEMU passes
    vCPU ids that fit well in the packing pattern. But nothing prevents
    userspace to come up with a forged vCPU id resulting in a packed id
    collision which causes the KVM device to associate two vCPUs to the
    same VP. This greatly confuses the irq layer and ultimately crashes
    the kernel, as shown below.

    Example: a guest with 1 guest thread per core, a core stride of
    8 and 300 vCPUs has vCPU ids 0,8,16...2392. If QEMU is patched to
    inject at some point an invalid vCPU id 348, which is the packed
    version of itself and 2392, we get:

    genirq: Flags mismatch irq 199. 00010000 (kvm-2-2392) vs. 00010000 (kvm-2-348)
    CPU: 24 PID: 88176 Comm: qemu-system-ppc Not tainted 5.3.0-xive-nr-servers-5.3-gku+ #38
    Call Trace:
    [c000003f7f9937e0] [c000000000c0110c] dump_stack+0xb0/0xf4 (unreliable)
    [c000003f7f993820] [c0000000001cb480] __setup_irq+0xa70/0xad0
    [c000003f7f9938d0] [c0000000001cb75c] request_threaded_irq+0x13c/0x260
    [c000003f7f993940] [c00800000d44e7ac] kvmppc_xive_attach_escalation+0x104/0x270 [kvm]
    [c000003f7f9939d0] [c00800000d45013c] kvmppc_xive_connect_vcpu+0x424/0x620 [kvm]
```

```
    [c000003f7f993ac0] [c00800000d444428] kvm_arch_vcpu_ioctl+0x260/0x448 [kvm]
    [c000003f7f993b90] [c00800000d43593c] kvm_vcpu_ioctl+0x154/0x7c8 [kvm]
    [c000003f7f993d00] [c0000000004840f0] do_vfs_ioctl+0xe0/0xc30
    [c000003f7f993db0] [c000000000484d44] ksys_ioctl+0x104/0x120
    [c000003f7f993e00] [c000000000484d88] sys_ioctl+0x28/0x80
    [c000003f7f993e20] [c00000000000b278] system_call+0x5c/0x68
    xive-kvm: Failed to request escalation interrupt for queue 0 of VCPU 2392
    ------------[ cut here ]------------
    remove_proc_entry: removing non-empty directory 'irq/199', leaking at least 'kvm-2-348'
    WARNING: CPU: 24 PID: 88176 at /home/greg/Work/linux/kernel-kvm-
ppc/fs/proc/generic.c:684 remove_proc_entry+0x1ec/0x200
    Modules linked in: kvm_hv kvm dm_mod vhost_net vhost tap xt_CHECKSUM iptable_mangle
xt_MASQUERADE iptable_nat nf_nat xt_conntrack nf_conntrack nf_defrag_ipv6 nf_defrag_ipv4
ipt_REJECT nf_reject_ipv4 tun bridge stp llc ebtable_filter ebtables ip6table_filter
ip6_tables iptable_filter squashfs loop fuse i2c_dev sg ofpart ocxl powernv_flash at24 xts
mtd uio_pdrv_genirq vmx_crypto opal_prd ipmi_powernv uio ipmi_devintf ipmi_msghandler
ibmpowernv ib_iser rdma_cm iw_cm ib_cm ib_core iscsi_tcp libiscsi_tcp libiscsi
scsi_transport_iscsi ip_tables ext4 mbcache jbd2 raid10 raid456 async_raid6_recov
async_memcpy async_pq async_xor xor async_tx raid6_pq libcrc32c raid1 raid0 linear sd_mod
ast i2c_algo_bit drm_vram_helper ttm drm_kms_helper syscopyarea sysfillrect sysimgblt
fb_sys_fops drm ahci libahci libata tg3 drm_panel_orientation_quirks [last unloaded: kvm]
    CPU: 24 PID: 88176 Comm: qemu-system-ppc Not tainted 5.3.0-xive-nr-servers-5.3-gku+ #38
    NIP:  c00000000053b0cc LR: c00000000053b0c8 CTR: c0000000000ba3b0
    REGS: c000003f7f9934b0 TRAP: 0700   Not tainted  (5.3.0-xive-nr-servers-5.3-gku+)
    MSR:  9000000000029033 <SF,HV,EE,ME,IR,DR,RI,LE>  CR: 48228222  XER: 20040000
    CFAR: c000000000131a50 IRQMASK: 0
    GPR00: c00000000053b0c8 c000003f7f993740 c0000000015ec500 0000000000000057
    GPR04: 0000000000000001 0000000000000000 000049fb98484262 0000000000001bcf
    GPR08: 0000000000000007 0000000000000007 0000000000000001 9000000000001033
    GPR12: 0000000000008000 c000003ffffeb800 0000000000000000 000000012f4ce5a1
    GPR16: 000000012ef5a0c8 0000000000000000 000000012f113bb0 0000000000000000
    GPR20: 000000012f45d918 c000003f863758b0 c000003f86375870 0000000000000006
    GPR24: c000003f86375a30 0000000000000007 c0002039373d9020 c0000000014c4a48
    GPR28: 0000000000000001 c000003fe62a4f6b c00020394b2e9fab c000003fe62a4ec0
    NIP [c00000000053b0cc] remove_proc_entry+0x1ec/0x200
    LR [c00000000053b0c8] remove_proc_entry+0x1e8/0x200
    Call Trace:
    [c000003f7f993740] [c00000000053b0c8] remove_proc_entry+0x1e8/0x200 (unreliable)
    [c000003f7f9937e0] [c0000000001d3654] unregister_irq_proc+0x114/0x150
    [c000003f7f993880] [c0000000001c6284] free_desc+0x54/0xb0
    [c000003f7f9938c0] [c0000000001c65ec] irq_free_descs+0xac/0x100
    [c000003f7f993910] [c0000000001d1ff8] irq_dispose_mapping+0x68/0x80
    [c000003f7f993940] [c00800000d44e8a4] kvmppc_xive_attach_escalation+0x1fc/0x270 [kvm]
    [c000003f7f9939d0] [c00800000d45013c] kvmppc_xive_connect_vcpu+0x424/0x620 [kvm]
    [c000003f7f993ac0] [c00800000d444428] kvm_arch_vcpu_ioctl+0x260/0x448 [kvm]
    [c000003f7f993b90] [c00800000d43593c] kvm_vcpu_ioctl+0x154/0x7c8 [kvm]
    [c000003f7f993d00] [c0000000004840f0] do_vfs_ioctl+0xe0/0xc30
    [c000003f7f993db0] [c000000000484d44] ksys_ioctl+0x104/0x120
    [c000003f7f993e00] [c000000000484d88] sys_ioctl+0x28/0x80
    [c000003f7f993e20] [c00000000000b278] system_call+0x5c/0x68
    Instruction dump:
    2c230000 41820008 3923ff78 e8e900a0 3c82ff69 3c62ff8d 7fa6eb78 7fc5f378
    3884f080 3863b948 4bbf6925 60000000 <0fe00000> 4bffff7c fba10088 4bbf6e41
    ---[ end trace b925b67a74a1d8d1 ]---
    BUG: Kernel NULL pointer dereference at 0x00000010
    Faulting instruction address: 0xc00800000d44fc04
    Oops: Kernel access of bad area, sig: 11 [#1]
    LE PAGE_SIZE=64K MMU=Radix MMU=Hash SMP NR_CPUS=2048 NUMA PowerNV
    Modules linked in: kvm_hv kvm dm_mod vhost_net vhost tap xt_CHECKSUM iptable_mangle
xt_MASQUERADE iptable_nat nf_nat xt_conntrack nf_conntrack nf_defrag_ipv6 nf_defrag_ipv4
ipt_REJECT nf_reject_ipv4 tun bridge stp llc ebtable_filter ebtables ip6table_filter
ip6_tables iptable_filter squashfs loop fuse i2c_dev sg ofpart ocxl powernv_flash at24 xts
mtd uio_pdrv_genirq vmx_crypto opal_prd ipmi_powernv uio ipmi_devintf ipmi_msghandler
ibmpowernv ib_iser rdma_cm iw_cm ib_cm ib_core iscsi_tcp libiscsi_tcp libiscsi
scsi_transport_iscsi ip_tables ext4 mbcache jbd2 raid10 raid456 async_raid6_recov
```

```
 async_memcpy async_pq async_xor xor async_tx raid6_pq libcrc32c raid1 raid0 linear sd_mod
 ast i2c_algo_bit drm_vram_helper ttm drm_kms_helper syscopyarea sysfillrect sysimgblt
 fb_sys_fops drm ahci libahci libata tg3 drm_panel_orientation_quirks [last unloaded: kvm]
     CPU: 24 PID: 88176 Comm: qemu-system-ppc Tainted: G        W         5.3.0-xive-nr-
 servers-5.3-gku+ #38
     NIP:  c00800000d44fc04 LR: c00800000d44fc00 CTR: c0000000001cd970
     REGS: c000003f7f9938e0 TRAP: 0300   Tainted: G        W          (5.3.0-xive-nr-servers-
 5.3-gku+)
     MSR:  9000000000009033 <SF,HV,EE,ME,IR,DR,RI,LE>  CR: 24228882  XER: 20040000
     CFAR: c0000000001cd9ac DAR: 0000000000000010 DSISR: 40000000 IRQMASK: 0
     GPR00: c00800000d44fc00 c000003f7f993b70 c00800000d468300 0000000000000000
     GPR04: 00000000000000c7 0000000000000000 0000000000000000 c000003ffacd06d8
     GPR08: 0000000000000000 c000003ffacd0738 0000000000000000 fffffffffffffffd
     GPR12: 0000000000000040 c000003ffffeb800 0000000000000000 000000012f4ce5a1
     GPR16: 000000012ef5a0c8 0000000000000000 000000012f113bb0 0000000000000000
     GPR20: 000000012f45d918 00007ffffe0d9a80 000000012f4f5df0 000000012ef8c9f8
     GPR24: 0000000000000001 0000000000000000 c000003fe4501ed0 c000003f8b1d0000
     GPR28: c0000033314689c0 c000003fe4501c00 c000003fe4501e70 c000003fe4501e90
     NIP [c00800000d44fc04] kvmppc_xive_cleanup_vcpu+0xfc/0x210 [kvm]
     LR [c00800000d44fc00] kvmppc_xive_cleanup_vcpu+0xf8/0x210 [kvm]
     Call Trace:
     [c000003f7f993b70] [c00800000d44fc00] kvmppc_xive_cleanup_vcpu+0xf8/0x210 [kvm]
 (unreliable)
     [c000003f7f993bd0] [c00800000d450bd4] kvmppc_xive_release+0xdc/0x1b0 [kvm]
     [c000003f7f993c30] [c00800000d436a98] kvm_device_release+0xb0/0x110 [kvm]
     [c000003f7f993c70] [c00000000046730c] __fput+0xec/0x320
     [c000003f7f993cd0] [c000000000164ae0] task_work_run+0x150/0x1c0
     [c000003f7f993d30] [c000000000025034] do_notify_resume+0x304/0x440
     [c000003f7f993e20] [c00000000000dcc4] ret_from_except_lite+0x70/0x74
     Instruction dump:
     3bff0008 7fbfd040 419e0054 847e0004 2fa30000 419effec e93d0000 8929203c
     2f890000 419effb8 4800821d e8410018 <e9230010> e9490008 9b2a0039 7c0004ac
     ---[ end trace b925b67a74a1d8d2 ]---

     Kernel panic - not syncing: Fatal exception

     This affects both XIVE and XICS-on-XIVE devices since the beginning.

     Check the VP id instead of the vCPU id when a new vCPU is connected.
     The allocation of the XIVE CPU structure in kvmppc_xive_connect_vcpu()
     is moved after the check to avoid the need for rollback.

     Cc: stable@vger.kernel.org # v4.12+
     Signed-off-by: Greg Kurz <groug@kaod.org>
     Reviewed-by: Cédric Le Goater <clg@kaod.org>
     Signed-off-by: Paul Mackerras <paulus@ozlabs.org>
     Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

 commit 03ad9b7ea7162c5191fa0eeb2322fbeb7f6c89a5
 Author: Qu Wenruo <wqu@suse.com>
 Date:   Thu Oct 17 10:38:37 2019 +0800

     btrfs: tracepoints: Fix bad entry members of qgroup events

     commit 1b2442b4ae0f234daeadd90e153b466332c466d8 upstream.

     [BUG]
     For btrfs:qgroup_meta_reserve event, the trace event can output garbage:

       qgroup_meta_reserve: 9c7f6acc-b342-4037-bc47-7f6e4d2232d7: refroot=5(FS_TREE)
 type=DATA diff=2
       qgroup_meta_reserve: 9c7f6acc-b342-4037-bc47-7f6e4d2232d7: refroot=5(FS_TREE)
 type=0x258792 diff=2

     The @type can be completely garbage, as DATA type is not possible for
```

```
    trace_qgroup_meta_reserve() trace event.

    [CAUSE]
    Ther are several problems related to qgroup trace events:
    - Unassigned entry member
      Member entry::type of trace_qgroup_update_reserve() and
      trace_qgourp_meta_reserve() is not assigned

    - Redundant entry member
      Member entry::type is completely useless in
      trace_qgroup_meta_convert()

    Fixes: 4ee0d8832c2e ("btrfs: qgroup: Update trace events for metadata reservation")
    CC: stable@vger.kernel.org # 4.10+
    Reviewed-by: Nikolay Borisov <nborisov@suse.com>
    Signed-off-by: Qu Wenruo <wqu@suse.com>
    Reviewed-by: David Sterba <dsterba@suse.com>
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c5c92780a1904a670454e46cba3702c6ed84ece5
Author: Qu Wenruo <wqu@suse.com>
Date:   Thu Oct 17 10:38:36 2019 +0800

    btrfs: tracepoints: Fix wrong parameter order for qgroup events

    commit fd2b007eaec898564e269d1f478a2da0380ecf51 upstream.

    [BUG]
    For btrfs:qgroup_meta_reserve event, the trace event can output garbage:

      qgroup_meta_reserve: 9c7f6acc-b342-4037-bc47-7f6e4d2232d7: refroot=5(FS_TREE)
type=DATA diff=2

    The diff should always be alinged to sector size (4k), so there is
    definitely something wrong.

    [CAUSE]
    For the wrong @diff, it's caused by wrong parameter order.
    The correct parameters are:

      struct btrfs_root, s64 diff, int type.

    However the parameters used are:

      struct btrfs_root, int type, s64 diff.

    Fixes: 4ee0d8832c2e ("btrfs: qgroup: Update trace events for metadata reservation")
    CC: stable@vger.kernel.org # 4.19+
    Reviewed-by: Nikolay Borisov <nborisov@suse.com>
    Signed-off-by: Qu Wenruo <wqu@suse.com>
    Reviewed-by: David Sterba <dsterba@suse.com>
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 6b19f2cc2d590d1a99e649709ab38de406024c66
Author: Filipe Manana <fdmanana@suse.com>
Date:   Wed Oct 16 16:28:52 2019 +0100

    Btrfs: check for the full sync flag while holding the inode lock during fsync

    commit ba0b084ac309283db6e329785c1dc4f45fdbd379 upstream.

    We were checking for the full fsync flag in the inode before locking the
    inode, which is racy, since at that that time it might not be set but
```

```
    after we acquire the inode lock some other task set it. One case where
    this can happen is on a system low on memory and some concurrent task
    failed to allocate an extent map and therefore set the full sync flag on
    the inode, to force the next fsync to work in full mode.

    A consequence of missing the full fsync flag set is hitting the problems
    fixed by commit 0c713cbab620 ("Btrfs: fix race between ranged fsync and
    writeback of adjacent ranges"), BUG_ON() when dropping extents from a log
    tree, hitting assertion failures at tree-log.c:copy_items() or all sorts
    of weird inconsistencies after replaying a log due to file extents items
    representing ranges that overlap.

    So just move the check such that it's done after locking the inode and
    before starting writeback again.

    Fixes: 0c713cbab620 ("Btrfs: fix race between ranged fsync and writeback of adjacent
ranges")
    CC: stable@vger.kernel.org # 5.2+
    Signed-off-by: Filipe Manana <fdmanana@suse.com>
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 4b448150c1645e6406bef72f9dbe2b0552cd8552
Author: Filipe Manana <fdmanana@suse.com>
Date:   Tue Oct 15 10:54:39 2019 +0100

    Btrfs: fix qgroup double free after failure to reserve metadata for delalloc

    commit c7967fc1499beb9b70bb9d33525fb0b384af8883 upstream.

    If we fail to reserve metadata for delalloc operations we end up releasing
    the previously reserved qgroup amount twice, once explicitly under the
    'out_qgroup' label by calling btrfs_qgroup_free_meta_prealloc() and once
    again, under label 'out_fail', by calling btrfs_inode_rsv_release() with a
    value of 'true' for its 'qgroup_free' argument, which results in
    btrfs_qgroup_free_meta_prealloc() being called again, so we end up having
    a double free.

    Also if we fail to reserve the necessary qgroup amount, we jump to the
    label 'out_fail', which calls btrfs_inode_rsv_release() and that in turns
    calls btrfs_qgroup_free_meta_prealloc(), even though we weren't able to
    reserve any qgroup amount. So we freed some amount we never reserved.

    So fix this by removing the call to btrfs_inode_rsv_release() in the
    failure path, since it's not necessary at all as we haven't changed the
    inode's block reserve in any way at this point.

    Fixes: c8eaeac7b73434 ("btrfs: reserve delalloc metadata differently")
    CC: stable@vger.kernel.org # 5.2+
    Signed-off-by: Filipe Manana <fdmanana@suse.com>
    Reviewed-by: David Sterba <dsterba@suse.com>
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit fb5b7a63e55641adef3a2c84f31ab4956f510e43
Author: David Sterba <dsterba@suse.com>
Date:   Sat Oct 12 18:42:10 2019 +0200

    btrfs: don't needlessly create extent-refs kernel thread

    commit 80ed4548d0711d15ca51be5dee0ff813051cfc90 upstream.

    The patch 32b593bfcb58 ("Btrfs: remove no longer used function to run
    delayed refs asynchronously") removed the async delayed refs but the
    thread has been created, without any use. Remove it to avoid resource
```

```
    consumption.

    Fixes: 32b593bfcb58 ("Btrfs: remove no longer used function to run delayed refs
 asynchronously")
    CC: stable@vger.kernel.org # 5.2+
    Reviewed-by: Josef Bacik <josef@toxicpanda.com>
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 467e21ac196b2e0386db368891a801c73aafa572
Author: Filipe Manana <fdmanana@suse.com>
Date:   Wed Oct 9 17:43:45 2019 +0100

    Btrfs: add missing extents release on file extent cluster relocation error

    commit 44db1216efe37bf670f8d1019cdc41658d84baf5 upstream.

    If we error out when finding a page at relocate_file_extent_cluster(), we
    need to release the outstanding extents counter on the relocation inode,
    set by the previous call to btrfs_delalloc_reserve_metadata(), otherwise
    the inode's block reserve size can never decrease to zero and metadata
    space is leaked. Therefore add a call to btrfs_delalloc_release_extents()
    in case we can't find the target page.

    Fixes: 8b62f87bad9c ("Btrfs: rework outstanding_extents")
    CC: stable@vger.kernel.org # 4.19+
    Signed-off-by: Filipe Manana <fdmanana@suse.com>
    Reviewed-by: David Sterba <dsterba@suse.com>
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c4c091c87a7a804bb091a190463426fbd99dc49c
Author: Qu Wenruo <wqu@suse.com>
Date:   Thu Oct 10 10:39:26 2019 +0800

    btrfs: block-group: Fix a memory leak due to missing btrfs_put_block_group()

    commit 4b654acdae850f48b8250b9a578a4eaa518c7a6f upstream.

    In btrfs_read_block_groups(), if we have an invalid block group which
    has mixed type (DATA|METADATA) while the fs doesn't have MIXED_GROUPS
    feature, we error out without freeing the block group cache.

    This patch will add the missing btrfs_put_block_group() to prevent
    memory leak.

    Note for stable backports: the file to patch in versions <= 5.3 is
    fs/btrfs/extent-tree.c

    Fixes: 49303381f19a ("Btrfs: bail out if block group has different mixed flag")
    CC: stable@vger.kernel.org # 4.9+
    Reviewed-by: Anand Jain <anand.jain@oracle.com>
    Reviewed-by: Johannes Thumshirn <jthumshirn@suse.de>
    Signed-off-by: Qu Wenruo <wqu@suse.com>
    Reviewed-by: David Sterba <dsterba@suse.com>
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c5f0d7d4540e6530e19b792a58f6c44177f591b7
Author: Patrick Williams <alpawi@amazon.com>
Date:   Tue Oct 1 10:51:38 2019 -0500

    pinctrl: armada-37xx: swap polarity on LED group

    commit b835d6953009dc350d61402a854b5a7178d8c615 upstream.
```

```
    The configuration registers for the LED group have inverted
    polarity, which puts the GPIO into open-drain state when used in
    GPIO mode.  Switch to '0' for GPIO and '1' for LED modes.

    Fixes: 87466ccd9401 ("pinctrl: armada-37xx: Add pin controller support for Armada 37xx")
    Signed-off-by: Patrick Williams <alpawi@amazon.com>
    Cc: <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20191001155154.99710-1-alpawi@amazon.com
    Signed-off-by: Linus Walleij <linus.walleij@linaro.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 094c0d824044fa8d5516931d7462e9cce69e2402
Author: Patrick Williams <alpawi@amazon.com>
Date:   Tue Oct 1 10:46:31 2019 -0500

    pinctrl: armada-37xx: fix control of pins 32 and up

    commit 20504fa1d2ffd5d03cdd9dc9c9dd4ed4579b97ef upstream.

    The 37xx configuration registers are only 32 bits long, so
    pins 32-35 spill over into the next register.  The calculation
    for the register address was done, but the bitmask was not, so
    any configuration to pin 32 or above resulted in a bitmask that
    overflowed and performed no action.

    Fix the register / offset calculation to also adjust the offset.

    Fixes: 5715092a458c ("pinctrl: armada-37xx: Add gpio support")
    Signed-off-by: Patrick Williams <alpawi@amazon.com>
    Acked-by: Gregory CLEMENT <gregory.clement@bootlin.com>
    Cc: <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20191001154634.96165-1-alpawi@amazon.com
    Signed-off-by: Linus Walleij <linus.walleij@linaro.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 40a493c71e460f9dad2c9d718709f4cc33b81038
Author: Dmitry Torokhov <dmitry.torokhov@gmail.com>
Date:   Mon Sep 23 19:49:58 2019 -0700

    pinctrl: cherryview: restore Strago DMI workaround for all versions

    commit 260996c30f4f3a732f45045e3e0efe27017615e4 upstream.

    This is essentially a revert of:

    e3f72b749da2 pinctrl: cherryview: fix Strago DMI workaround
    86c5dd6860a6 pinctrl: cherryview: limit Strago DMI workarounds to version 1.0

    because even with 1.1 versions of BIOS there are some pins that are
    configured as interrupts but not claimed by any driver, and they
    sometimes fire up and result in interrupt storms that cause touchpad
    stop functioning and other issues.

    Given that we are unlikely to qualify another firmware version for a
    while it is better to keep the workaround active on all Strago boards.

    Reported-by: Alex Levin <levinale@chromium.org>
    Fixes: 86c5dd6860a6 ("pinctrl: cherryview: limit Strago DMI workarounds to version 1.0")
    Cc: stable@vger.kernel.org
    Signed-off-by: Dmitry Torokhov <dmitry.torokhov@gmail.com>
    Reviewed-by: Andy Shevchenko <andriy.shevchenko@linux.intel.com>
    Tested-by: Alex Levin <levinale@chromium.org>
    Signed-off-by: Mika Westerberg <mika.westerberg@linux.intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit 69167f7b43a693abdeb0d2b25c16dbec25f90da8
Author: Roman Kagan <rkagan@virtuozzo.com>
Date:   Thu Oct 10 12:33:05 2019 +0000

    x86/hyperv: Make vapic support x2apic mode

    commit e211288b72f15259da86eed6eca680758dbe9e74 upstream.

    Now that there's Hyper-V IOMMU driver, Linux can switch to x2apic mode
    when supported by the vcpus.

    However, the apic access functions for Hyper-V enlightened apic assume
    xapic mode only.

    As a result, Linux fails to bring up secondary cpus when run as a guest
    in QEMU/KVM with both hv_apic and x2apic enabled.

    According to Michael Kelley, when in x2apic mode, the Hyper-V synthetic
    apic MSRs behave exactly the same as the corresponding architectural
    x2apic MSRs, so there's no need to override the apic accessors.  The
    only exception is hv_apic_eoi_write, which benefits from lazy EOI when
    available; however, its implementation works for both xapic and x2apic
    modes.

    Fixes: 29217a474683 ("iommu/hyper-v: Add Hyper-V stub IOMMU driver")
    Fixes: 6b48cb5f8347 ("X86/Hyper-V: Enlighten APIC access")
    Suggested-by: Michael Kelley <mikelley@microsoft.com>
    Signed-off-by: Roman Kagan <rkagan@virtuozzo.com>
    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Reviewed-by: Vitaly Kuznetsov <vkuznets@redhat.com>
    Reviewed-by: Michael Kelley <mikelley@microsoft.com>
    Cc: stable@vger.kernel.org
    Link: https://lkml.kernel.org/r/20191010123258.16919-1-rkagan@virtuozzo.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit d59040b307141fc3839280f4e90772f31321f0bf
Author: Sean Christopherson <sean.j.christopherson@intel.com>
Date:   Tue Oct 1 13:50:19 2019 -0700

    x86/apic/x2apic: Fix a NULL pointer deref when handling a dying cpu

    commit 7a22e03b0c02988e91003c505b34d752a51de344 upstream.

    Check that the per-cpu cluster mask pointer has been set prior to
    clearing a dying cpu's bit.  The per-cpu pointer is not set until the
    target cpu reaches smp_callin() during CPUHP_BRINGUP_CPU, whereas the
    teardown function, x2apic_dead_cpu(), is associated with the earlier
    CPUHP_X2APIC_PREPARE.  If an error occurs before the cpu is awakened,
    e.g. if do_boot_cpu() itself fails, x2apic_dead_cpu() will dereference
    the NULL pointer and cause a panic.

      smpboot: do_boot_cpu failed(-22) to wakeup CPU#1
      BUG: kernel NULL pointer dereference, address: 0000000000000008
      RIP: 0010:x2apic_dead_cpu+0x1a/0x30
      Call Trace:
       cpuhp_invoke_callback+0x9a/0x580
       _cpu_up+0x10d/0x140
       do_cpu_up+0x69/0xb0
       smp_init+0x63/0xa9
       kernel_init_freeable+0xd7/0x229
       ? rest_init+0xa0/0xa0
       kernel_init+0xa/0x100
       ret_from_fork+0x35/0x40
```

```
    Fixes: 023a611748fd5 ("x86/apic/x2apic: Simplify cluster management")
    Signed-off-by: Sean Christopherson <sean.j.christopherson@intel.com>
    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Cc: stable@vger.kernel.org
    Link: https://lkml.kernel.org/r/20191001205019.5789-1-sean.j.christopherson@intel.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1776cd44e71fd35b5fa856d2eb7eddfeb9cc715b
Author: Steve Wahl <steve.wahl@hpe.com>
Date:   Tue Sep 24 16:03:55 2019 -0500

    x86/boot/64: Make level2_kernel_pgt pages invalid outside kernel area

    commit 2aa85f246c181b1fa89f27e8e20c5636426be624 upstream.

    Our hardware (UV aka Superdome Flex) has address ranges marked
    reserved by the BIOS. Access to these ranges is caught as an error,
    causing the BIOS to halt the system.

    Initial page tables mapped a large range of physical addresses that
    were not checked against the list of BIOS reserved addresses, and
    sometimes included reserved addresses in part of the mapped range.
    Including the reserved range in the map allowed processor speculative
    accesses to the reserved range, triggering a BIOS halt.

    Used early in booting, the page table level2_kernel_pgt addresses 1
    GiB divided into 2 MiB pages, and it was set up to linearly map a full
     1 GiB of physical addresses that included the physical address range
    of the kernel image, as chosen by KASLR.  But this also included a
    large range of unused addresses on either side of the kernel image.
    And unlike the kernel image's physical address range, this extra
    mapped space was not checked against the BIOS tables of usable RAM
    addresses.  So there were times when the addresses chosen by KASLR
    would result in processor accessible mappings of BIOS reserved
    physical addresses.

    The kernel code did not directly access any of this extra mapped
    space, but having it mapped allowed the processor to issue speculative
    accesses into reserved memory, causing system halts.

    This was encountered somewhat rarely on a normal system boot, and much
    more often when starting the crash kernel if "crashkernel=512M,high"
    was specified on the command line (this heavily restricts the physical
    address of the crash kernel, in our case usually within 1 GiB of
    reserved space).

    The solution is to invalidate the pages of this table outside the kernel
    image's space before the page table is activated. It fixes this problem
    on our hardware.

     [ bp: Touchups. ]

    Signed-off-by: Steve Wahl <steve.wahl@hpe.com>
    Signed-off-by: Borislav Petkov <bp@suse.de>
    Acked-by: Dave Hansen <dave.hansen@linux.intel.com>
    Acked-by: Kirill A. Shutemov <kirill.shutemov@linux.intel.com>
    Cc: Baoquan He <bhe@redhat.com>
    Cc: Brijesh Singh <brijesh.singh@amd.com>
    Cc: dimitri.sivanich@hpe.com
    Cc: Feng Tang <feng.tang@intel.com>
    Cc: "H. Peter Anvin" <hpa@zytor.com>
    Cc: Ingo Molnar <mingo@redhat.com>
    Cc: Jordan Borgner <mail@jordan-borgner.de>
    Cc: Juergen Gross <jgross@suse.com>
    Cc: mike.travis@hpe.com
```

```
    Cc: russ.anderson@hpe.com
    Cc: stable@vger.kernel.org
    Cc: Thomas Gleixner <tglx@linutronix.de>
    Cc: x86-ml <x86@kernel.org>
    Cc: Zhenzhong Duan <zhenzhong.duan@oracle.com>
    Link:
https://lkml.kernel.org/r/9c011ee51b081534a7a15065b1681d200298b530.1569358539.git.steve.wahl
@hpe.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit fd1c6478aae0e5365bffc3469a5136bf26899796
Author: Marc Zyngier <maz@kernel.org>
Date:   Sun Sep 15 15:17:45 2019 +0100

    irqchip/sifive-plic: Switch to fasteoi flow

    commit bb0fed1c60cccbe4063b455a7228818395dac86e upstream.

    The SiFive PLIC interrupt controller seems to have all the HW
    features to support the fasteoi flow, but the driver seems to be
    stuck in a distant past. Bring it into the 21st century.

    Signed-off-by: Marc Zyngier <maz@kernel.org>
    Tested-by: Palmer Dabbelt <palmer@sifive.com> (QEMU Boot)
    Tested-by: Darius Rad <darius@bluespec.com> (on 2 HW PLIC implementations)
    Tested-by: Paul Walmsley <paul.walmsley@sifive.com> (HiFive Unleashed)
    Reviewed-by: Palmer Dabbelt <palmer@sifive.com>
    Cc: stable@vger.kernel.org
    Link: https://lore.kernel.org/r/8636gxskmj.wl-maz@kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit bd249c9b3476674a2da1c262b6ca0d86cac83290
Author: Mikulas Patocka <mpatocka@redhat.com>
Date:   Wed Oct 16 09:21:50 2019 -0400

    dm cache: fix bugs when a GFP_NOWAIT allocation fails

    commit 13bd677a472d534bf100bab2713efc3f9e3f5978 upstream.

    GFP_NOWAIT allocation can fail anytime - it doesn't wait for memory being
    available and it fails if the mempool is exhausted and there is not enough
    memory.

    If we go down this path:
      map_bio -> mg_start -> alloc_migration -> mempool_alloc(GFP_NOWAIT)
    we can see that map_bio() doesn't check the return value of mg_start(),
    and the bio is leaked.

    If we go down this path:
      map_bio -> mg_start -> mg_lock_writes -> alloc_prison_cell ->
      dm_bio_prison_alloc_cell_v2 -> mempool_alloc(GFP_NOWAIT) ->
      mg_lock_writes -> mg_complete
    the bio is ended with an error - it is unacceptable because it could
    cause filesystem corruption if the machine ran out of memory
    temporarily.

    Change GFP_NOWAIT to GFP_NOIO, so that the mempool code will properly
    wait until memory becomes available. mempool_alloc with GFP_NOIO can't
    fail, so remove the code paths that deal with allocation failure.

    Cc: stable@vger.kernel.org
    Signed-off-by: Mikulas Patocka <mpatocka@redhat.com>
    Signed-off-by: Mike Snitzer <snitzer@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit a71d83cde91e0be74db3b1c3682ee2cab4a2f413
Author: Dan Williams <dan.j.williams@intel.com>
Date:   Mon Oct 21 09:29:20 2019 -0700

    fs/dax: Fix pmd vs pte conflict detection

    commit 6370740e5f8ef12de7f9a9bf48a0393d202cd827 upstream.

    Users reported a v5.3 performance regression and inability to establish
    huge page mappings. A revised version of the ndctl "dax.sh" huge page
    unit test identifies commit 23c84eb78375 "dax: Fix missed wakeup with
    PMD faults" as the source.

    Update get_unlocked_entry() to check for NULL entries before checking
    the entry order, otherwise NULL is misinterpreted as a present pte
    conflict. The 'order' check needs to happen before the locked check as
    an unlocked entry at the wrong order must fallback to lookup the correct
    order.

    Reported-by: Jeff Smits <jeff.smits@intel.com>
    Reported-by: Doug Nelson <doug.nelson@intel.com>
    Cc: <stable@vger.kernel.org>
    Fixes: 23c84eb78375 ("dax: Fix missed wakeup with PMD faults")
    Reviewed-by: Jan Kara <jack@suse.cz>
    Cc: Jeff Moyer <jmoyer@redhat.com>
    Cc: Matthew Wilcox (Oracle) <willy@infradead.org>
    Reviewed-by: Johannes Thumshirn <jthumshirn@suse.de>
    Link:
https://lore.kernel.org/r/157167532455.3945484.11971474077040503994.stgit@dwillia2-
desk3.amr.corp.intel.com
    Signed-off-by: Dan Williams <dan.j.williams@intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 2872b10f19556b3c568f3dcf44c595a028b07e16
Author: Prateek Sood <prsood@codeaurora.org>
Date:   Tue Oct 15 11:47:25 2019 +0530

    tracing: Fix race in perf_trace_buf initialization

    commit 6b1340cc00edeadd52ebd8a45171f38c8de2a387 upstream.

    A race condition exists while initialiazing perf_trace_buf from
    perf_trace_init() and perf_kprobe_init().

          CPU0                                          CPU1
    perf_trace_init()
      mutex_lock(&event_mutex)
        perf_trace_event_init()
          perf_trace_event_reg()
            total_ref_count == 0
            buf = alloc_percpu()
            perf_trace_buf[i] = buf
            tp_event->class->reg() //fails       perf_kprobe_init()
            goto fail                              perf_trace_event_init()
                                                     perf_trace_event_reg()
            fail:
              total_ref_count == 0

                                                       total_ref_count == 0
                                                       buf = alloc_percpu()
                                                       perf_trace_buf[i] = buf
                                                       tp_event->class->reg()
                                                       total_ref_count++

              free_percpu(perf_trace_buf[i])
```

```
                perf_trace_buf[i] = NULL

    Any subsequent call to perf_trace_event_reg() will observe total_ref_count > 0,
    causing the perf_trace_buf to be always NULL. This can result in perf_trace_buf
    getting accessed from perf_trace_buf_alloc() without being initialized. Acquiring
    event_mutex in perf_kprobe_init() before calling perf_trace_event_init() should
    fix this race.

    The race caused the following bug:

     Unable to handle kernel paging request at virtual address 0000003106f2003c
     Mem abort info:
       ESR = 0x96000045
       Exception class = DABT (current EL), IL = 32 bits
       SET = 0, FnV = 0
       EA = 0, S1PTW = 0
     Data abort info:
       ISV = 0, ISS = 0x00000045
       CM = 0, WnR = 1
     user pgtable: 4k pages, 39-bit VAs, pgdp = ffffffc034b9b000
     [0000003106f2003c] pgd=0000000000000000, pud=0000000000000000
     Internal error: Oops: 96000045 [#1] PREEMPT SMP
     Process syz-executor (pid: 18393, stack limit = 0xffffffc093190000)
     pstate: 80400005 (Nzcv daif +PAN -UAO)
     pc : __memset+0x20/0x1ac
     lr : memset+0x3c/0x50
     sp : ffffffc09319fc50

      __memset+0x20/0x1ac
      perf_trace_buf_alloc+0x140/0x1a0
      perf_trace_sys_enter+0x158/0x310
      syscall_trace_enter+0x348/0x7c0
      el0_svc_common+0x11c/0x368
      el0_svc_handler+0x12c/0x198
      el0_svc+0x8/0xc

    Ramdumps showed the following:
      total_ref_count = 3
      perf_trace_buf = (
          0x0 -> NULL,
          0x0 -> NULL,
          0x0 -> NULL,
          0x0 -> NULL)

    Link: http://lkml.kernel.org/r/1571120245-4186-1-git-send-email-prsood@codeaurora.org

    Cc: stable@vger.kernel.org
    Fixes: e12f03d7031a9 ("perf/core: Implement the 'perf_kprobe' PMU")
    Acked-by: Song Liu <songliubraving@fb.com>
    Signed-off-by: Prateek Sood <prsood@codeaurora.org>
    Signed-off-by: Steven Rostedt (VMware) <rostedt@goodmis.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b7fbb762b52547e797f737e46bc956b2232a03bb
Author: Alexander Shishkin <alexander.shishkin@linux.intel.com>
Date:   Tue Oct 22 10:39:40 2019 +0300

    perf/aux: Fix AUX output stopping

    commit f3a519e4add93b7b31a6616f0b09635ff2e6a159 upstream.

    Commit:

      8a58ddae2379 ("perf/core: Fix exclusive events' grouping")
```

```
    allows CAP_EXCLUSIVE events to be grouped with other events. Since all
    of those also happen to be AUX events (which is not the case the other
    way around, because arch/s390), this changes the rules for stopping the
    output: the AUX event may not be on its PMU's context any more, if it's
    grouped with a HW event, in which case it will be on that HW event's
    context instead. If that's the case, munmap() of the AUX buffer can't
    find and stop the AUX event, potentially leaving the last reference with
    the atomic context, which will then end up freeing the AUX buffer. This
    will then trip warnings:

    Fix this by using the context's PMU context when looking for events
    to stop, instead of the event's PMU context.

    Signed-off-by: Alexander Shishkin <alexander.shishkin@linux.intel.com>
    Cc: Arnaldo Carvalho de Melo <acme@redhat.com>
    Cc: Jiri Olsa <jolsa@redhat.com>
    Cc: Linus Torvalds <torvalds@linux-foundation.org>
    Cc: Peter Zijlstra <a.p.zijlstra@chello.nl>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Cc: Stephane Eranian <eranian@google.com>
    Cc: Thomas Gleixner <tglx@linutronix.de>
    Cc: Vince Weaver <vincent.weaver@maine.edu>
    Cc: stable@vger.kernel.org
    Link: https://lkml.kernel.org/r/20191022073940.61814-1-
alexander.shishkin@linux.intel.com
    Signed-off-by: Ingo Molnar <mingo@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1ab60236916b2112be0d9384d27760fd87aadcc8
Author: Pavel Shilovsky <pshilov@microsoft.com>
Date:   Wed Oct 23 15:37:19 2019 -0700

    CIFS: Fix use after free of file info structures

    commit 1a67c415965752879e2e9fad407bc44fc7f25f23 upstream.

    Currently the code assumes that if a file info entry belongs
    to lists of open file handles of an inode and a tcon then
    it has non-zero reference. The recent changes broke that
    assumption when putting the last reference of the file info.
    There may be a situation when a file is being deleted but
    nothing prevents another thread to reference it again
    and start using it. This happens because we do not hold
    the inode list lock while checking the number of references
    of the file info structure. Fix this by doing the proper
    locking when doing the check.

    Fixes: 487317c99477d ("cifs: add spinlock for the openFileList to cifsInodeInfo")
    Fixes: cb248819d209d ("cifs: use cifsInodeInfo->open_file_lock while iterating to avoid
a panic")
    Cc: Stable <stable@vger.kernel.org>
    Reviewed-by: Ronnie Sahlberg <lsahlber@redhat.com>
    Signed-off-by: Pavel Shilovsky <pshilov@microsoft.com>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 0715022a4f6dbd7c7b830007b2a00d5a9f06d66c
Author: Chuhong Yuan <hslester96@gmail.com>
Date:   Mon Oct 14 15:15:31 2019 +0800

    cifs: Fix missed free operations

    commit 783bf7b8b641167fb6f3f4f787f60ae62bad41b3 upstream.

    cifs_setattr_nounix has two paths which miss free operations
```

```
    for xid and fullpath.
    Use goto cifs_setattr_exit like other paths to fix them.

    CC: Stable <stable@vger.kernel.org>
    Fixes: aa081859b10c ("cifs: flush before set-info if we have writeable handles")
    Signed-off-by: Chuhong Yuan <hslester96@gmail.com>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Reviewed-by: Pavel Shilovsky <pshilov@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit dfebd925c3fccc95aacb9e108455701e7c187de8
Author: Roberto Bergantinos Corpas <rbergant@redhat.com>
Date:   Mon Oct 14 10:59:23 2019 +0200

    CIFS: avoid using MID 0xFFFF

    commit 03d9a9fe3f3aec508e485dd3dcfa1e99933b4bdb upstream.

    According to MS-CIFS specification MID 0xFFFF should not be used by the
    CIFS client, but we actually do. Besides, this has proven to cause races
    leading to oops between SendReceive2/cifs_demultiplex_thread. On SMB1,
    MID is a 2 byte value easy to reach in CurrentMid which may conflict with
    an oplock break notification request coming from server

    Signed-off-by: Roberto Bergantinos Corpas <rbergant@redhat.com>
    Reviewed-by: Ronnie Sahlberg <lsahlber@redhat.com>
    Reviewed-by: Aurelien Aptel <aaptel@suse.com>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    CC: Stable <stable@vger.kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 35ae16161f59861a10d35f9f91e6de10fca3d60d
Author: Marc Zyngier <marc.zyngier@arm.com>
Date:   Fri Sep 13 10:57:50 2019 +0100

    arm64: Allow CAVIUM_TX2_ERRATUM_219 to be selected

    commit 603afdc9438ac546181e843f807253d75d3dbc45 upstream.

    Allow the user to select the workaround for TX2-219, and update
    the silicon-errata.rst file to reflect this.

    Cc: <stable@vger.kernel.org>
    Signed-off-by: Marc Zyngier <marc.zyngier@arm.com>
    Signed-off-by: Will Deacon <will@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 542c4b6923c701f647785c8472c19b977a217cfd
Author: Marc Zyngier <marc.zyngier@arm.com>
Date:   Tue Apr 9 16:26:21 2019 +0100

    arm64: Enable workaround for Cavium TX2 erratum 219 when running SMT

    commit 93916beb70143c46bf1d2bacf814be3a124b253b upstream.

    It appears that the only case where we need to apply the TX2_219_TVM
    mitigation is when the core is in SMT mode. So let's condition the
    enabling on detecting a CPU whose MPIDR_EL1.Aff0 is non-zero.

    Cc: <stable@vger.kernel.org>
    Signed-off-by: Marc Zyngier <marc.zyngier@arm.com>
    Signed-off-by: Will Deacon <will@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 351a0572d40fec408cbae34f02246c68e206ce55
```

```
Author: Marc Zyngier <marc.zyngier@arm.com>
Date:   Tue Apr 9 16:22:24 2019 +0100

    arm64: Avoid Cavium TX2 erratum 219 when switching TTBR

    commit 9405447ef79bc93101373e130f72e9e6cbf17dbb upstream.

    As a PRFM instruction racing against a TTBR update can have undesirable
    effects on TX2, NOP-out such PRFM on cores that are affected by
    the TX2-219 erratum.

    Cc: <stable@vger.kernel.org>
    Signed-off-by: Marc Zyngier <marc.zyngier@arm.com>
    Signed-off-by: Will Deacon <will@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit adae972de033d8f93c75303c2df1e225375f7cfd
Author: Marc Zyngier <marc.zyngier@arm.com>
Date:   Thu Feb 7 16:01:21 2019 +0000

    arm64: KVM: Trap VM ops when ARM64_WORKAROUND_CAVIUM_TX2_219_TVM is set

    commit d3ec3a08fa700c8b46abb137dce4e2514a6f9668 upstream.

    In order to workaround the TX2-219 erratum, it is necessary to trap
    TTBRx_EL1 accesses to EL2. This is done by setting HCR_EL2.TVM on
    guest entry, which has the side effect of trapping all the other
    VM-related sysregs as well.

    To minimize the overhead, a fast path is used so that we don't
    have to go all the way back to the main sysreg handling code,
    unless the rest of the hypervisor expects to see these accesses.

    Cc: <stable@vger.kernel.org>
    Signed-off-by: Marc Zyngier <marc.zyngier@arm.com>
    Signed-off-by: Will Deacon <will@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 98582d2702eebb242013defa2c261f3f8a4336dc
Author: James Morse <james.morse@arm.com>
Date:   Mon Oct 14 18:19:18 2019 +0100

    EDAC/ghes: Fix Use after free in ghes_edac remove path

    commit 1e72e673b9d102ff2e8333e74b3308d012ddf75b upstream.

    ghes_edac models a single logical memory controller, and uses a global
    ghes_init variable to ensure only the first ghes_edac_register() will
    do anything.

    ghes_edac is registered the first time a GHES entry in the HEST is
    probed. There may be multiple entries, so subsequent attempts to
    register ghes_edac are silently ignored as the work has already been
    done.

    When a GHES entry is unregistered, it calls ghes_edac_unregister(),
    which free()s the memory behind the global variables in ghes_edac.

    But there may be multiple GHES entries, the next call to
    ghes_edac_unregister() will dereference the free()d memory, and attempt
    to free it a second time.

    This may also be triggered on a platform with one GHES entry, if the
    driver is unbound/re-bound and unbound. The re-bind step will do
    nothing because of ghes_init, the second unbind will then do the same
```

```
    work as the first.

    Doing the unregister work on the first call is unsafe, as another
    CPU may be processing a notification in ghes_edac_report_mem_error(),
    using the memory we are about to free.

    ghes_init is already half of the reference counting. We only need
    to do the register work for the first call, and the unregister work
    for the last. Add the unregister check.

    This means we no longer free ghes_edac's memory while there are
    GHES entries that may receive a notification.

    This was detected by KASAN and DEBUG_TEST_DRIVER_REMOVE.

     [ bp: merge into a single patch. ]

    Fixes: 0fe5f281f749 ("EDAC, ghes: Model a single, logical memory controller")
    Reported-by: John Garry <john.garry@huawei.com>
    Signed-off-by: James Morse <james.morse@arm.com>
    Signed-off-by: Borislav Petkov <bp@suse.de>
    Cc: linux-edac <linux-edac@vger.kernel.org>
    Cc: Mauro Carvalho Chehab <mchehab@kernel.org>
    Cc: Robert Richter <rrichter@marvell.com>
    Cc: Tony Luck <tony.luck@intel.com>
    Cc: <stable@vger.kernel.org>
    Link: https://lkml.kernel.org/r/20191014171919.85044-2-james.morse@arm.com
    Link: https://lkml.kernel.org/r/304df85b-8b56-b77e-1a11-aa23769f2e7c@huawei.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c2ccbde22ee28bb2a0ffaf50c1f51858de54f09b
Author: Helge Deller <deller@gmx.de>
Date:   Fri Oct 4 19:23:37 2019 +0200

    parisc: Fix vmap memory leak in ioremap()/iounmap()

    commit 513f7f747e1cba81f28a436911fba0b485878ebd upstream.

    Sven noticed that calling ioremap() and iounmap() multiple times leads
    to a vmap memory leak:
            vmap allocation for size 4198400 failed:
            use vmalloc=<size> to increase size

    It seems we missed calling vunmap() in iounmap().

    Signed-off-by: Helge Deller <deller@gmx.de>
    Noticed-by: Sven Schnelle <svens@stackframe.org>
    Cc: <stable@vger.kernel.org> # v3.16+
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ba149e6440840acb665147b4763274eacd6c1b6f
Author: Thomas Gleixner <tglx@linutronix.de>
Date:   Mon Oct 21 12:07:15 2019 +0200

    lib/vdso: Make clock_getres() POSIX compliant again

    commit 1638b8f096ca165965189b9626564c933c79fe63 upstream.

    A recent commit removed the NULL pointer check from the clock_getres()
    implementation causing a test case to fault.

    POSIX requires an explicit NULL pointer check for clock_getres() aside of
    the validity check of the clock_id argument for obscure reasons.

    Add it back for both 32bit and 64bit.
```

```
    Note, this is only a partial revert of the offending commit which does not
    bring back the broken fallback invocation in the the 32bit compat
    implementations of clock_getres() and clock_gettime().

    Fixes: a9446a906f52 ("lib/vdso/32: Remove inconsistent NULL pointer checks")
    Reported-by: Andreas Schwab <schwab@linux-m68k.org>
    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Tested-by: Christophe Leroy <christophe.leroy@c-s.fr>
    Cc: stable@vger.kernel.org
    Link:
https://lkml.kernel.org/r/alpine.DEB.2.21.1910211202260.1904@nanos.tec.linutronix.de
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit fa8ef4a0c1c9eea54113eb090c3a3dcda624039d
Author: Gerald Schaefer <gerald.schaefer@de.ibm.com>
Date:   Mon Oct 21 19:56:00 2019 +0200

    s390/kaslr: add support for R_390_GLOB_DAT relocation type

    commit ac49303d9ef0ad98b79867a380ef23480e48870b upstream.

    Commit "bpf: Process in-kernel BTF" in linux-next introduced an undefined
    __weak symbol, which results in an R_390_GLOB_DAT relocation type. That
    is not yet handled by the KASLR relocation code, and the kernel stops with
    the message "Unknown relocation type".

    Add code to detect and handle R_390_GLOB_DAT relocation types and undefined
    symbols.

    Fixes: 805bc0bc238f ("s390/kernel: build a relocatable kernel")
    Cc: <stable@vger.kernel.org> # v5.2+
    Acked-by: Heiko Carstens <heiko.carstens@de.ibm.com>
    Signed-off-by: Gerald Schaefer <gerald.schaefer@de.ibm.com>
    Signed-off-by: Vasily Gorbik <gor@linux.ibm.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 65b870d6eb0bf2084c5ce3a4638b44c6f9ccc4c9
Author: Johan Hovold <johan@kernel.org>
Date:   Thu Oct 10 15:13:33 2019 +0200

    s390/zcrypt: fix memleak at release

    commit 388bb19be8eab4674a660e0c97eaf60775362bc7 upstream.

    If a process is interrupted while accessing the crypto device and the
    global ap_perms_mutex is contented, release() could return early and
    fail to free related resources.

    Fixes: 00fab2350e6b ("s390/zcrypt: multiple zcrypt device nodes support")
    Cc: <stable@vger.kernel.org> # 4.19
    Cc: Harald Freudenberger <freude@linux.ibm.com>
    Signed-off-by: Johan Hovold <johan@kernel.org>
    Signed-off-by: Heiko Carstens <heiko.carstens@de.ibm.com>
    Signed-off-by: Vasily Gorbik <gor@linux.ibm.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a47e86abba87c5a625ba34e661a06bc798e40cf3
Author: Max Filippov <jcmvbkbc@gmail.com>
Date:   Tue Oct 15 21:51:43 2019 -0700

    xtensa: fix change_bit in exclusive access option

    commit 775fd6bfefc66a8c33e91dd9687ed530643b954d upstream.
```

```
    change_bit implementation for XCHAL_HAVE_EXCLUSIVE case changes all bits
    except the one required due to copy-paste error from clear_bit.

    Cc: stable@vger.kernel.org # v5.2+
    Fixes: f7c34874f04a ("xtensa: add exclusive atomics support")
    Signed-off-by: Max Filippov <jcmvbkbc@gmail.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 859628f7853eaa580123cfa5dd93906781481dc9
Author: Max Filippov <jcmvbkbc@gmail.com>
Date:   Mon Oct 14 15:48:19 2019 -0700

    xtensa: drop EXPORT_SYMBOL for outs*/ins*

    commit 8b39da985194aac2998dd9e3a22d00b596cebf1e upstream.

    Custom outs*/ins* implementations are long gone from the xtensa port,
    remove matching EXPORT_SYMBOLs.
    This fixes the following build warnings issued by modpost since commit
    15bfc2348d54 ("modpost: check for static EXPORT_SYMBOL* functions"):

      WARNING: "insb" [vmlinux] is a static EXPORT_SYMBOL
      WARNING: "insw" [vmlinux] is a static EXPORT_SYMBOL
      WARNING: "insl" [vmlinux] is a static EXPORT_SYMBOL
      WARNING: "outsb" [vmlinux] is a static EXPORT_SYMBOL
      WARNING: "outsw" [vmlinux] is a static EXPORT_SYMBOL
      WARNING: "outsl" [vmlinux] is a static EXPORT_SYMBOL

    Cc: stable@vger.kernel.org
    Fixes: d38efc1f150f ("xtensa: adopt generic io routines")
    Signed-off-by: Max Filippov <jcmvbkbc@gmail.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b0c233e5d62da8ab481939f733d87cbd48c76904
Author: Chenwandun <chenwandun@huawei.com>
Date:   Fri Oct 18 20:20:14 2019 -0700

    zram: fix race between backing_dev_show and backing_dev_store

    commit f7daefe4231e57381d92c2e2ad905a899c28e402 upstream.

    CPU0:                                   CPU1:
    backing_dev_show                        backing_dev_store
        ......                                  ......
        file = zram->backing_dev;
        down_read(&zram->init_lock);            down_read(&zram->init_init_lock)
        file_path(file, ...);                   zram->backing_dev = backing_dev;
        up_read(&zram->init_lock);              up_read(&zram->init_lock);

    gets the value of zram->backing_dev too early in backing_dev_show, which
    resultin the value being NULL at the beginning, and not NULL later.

    backtrace:
      d_path+0xcc/0x174
      file_path+0x10/0x18
      backing_dev_show+0x40/0xb4
      dev_attr_show+0x20/0x54
      sysfs_kf_seq_show+0x9c/0x10c
      kernfs_seq_show+0x28/0x30
      seq_read+0x184/0x488
      kernfs_fop_read+0x5c/0x1a4
      __vfs_read+0x44/0x128
      vfs_read+0xa0/0x138
      SyS_read+0x54/0xb4
```

```
    Link: http://lkml.kernel.org/r/1571046839-16814-1-git-send-email-chenwandun@huawei.com
    Signed-off-by: Chenwandun <chenwandun@huawei.com>
    Acked-by: Minchan Kim <minchan@kernel.org>
    Cc: Sergey Senozhatsky <sergey.senozhatsky.work@gmail.com>
    Cc: Jens Axboe <axboe@kernel.dk>
    Cc: <stable@vger.kernel.org>    [4.14+]
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 2277a48e5d88a2fc6e9cdb099472e7ce08b831de
Author: Jane Chu <jane.chu@oracle.com>
Date:   Mon Oct 14 14:12:29 2019 -0700

    mm/memory-failure: poison read receives SIGKILL instead of SIGBUS if mmaped more than
once

    commit 3d7fed4ad8ccb691d217efbb0f934e6a4df5ef91 upstream.

    Mmap /dev/dax more than once, then read the poison location using
    address from one of the mappings.  The other mappings due to not having
    the page mapped in will cause SIGKILLs delivered to the process.
    SIGKILL succeeds over SIGBUS, so user process loses the opportunity to
    handle the UE.

    Although one may add MAP_POPULATE to mmap(2) to work around the issue,
    MAP_POPULATE makes mapping 128GB of pmem several magnitudes slower, so
    isn't always an option.

    Details -

      ndctl inject-error --block=10 --count=1 namespace6.0

      ./read_poison -x dax6.0 -o 5120 -m 2
      mmaped address 0x7f5bb6600000
      mmaped address 0x7f3cf3600000
      doing local read at address 0x7f3cf3601400
      Killed

    Console messages in instrumented kernel -

      mce: Uncorrected hardware memory error in user-access at edbe201400
      Memory failure: tk->addr = 7f5bb6601000
      Memory failure: address edbe201: call dev_pagemap_mapping_shift
      dev_pagemap_mapping_shift: page edbe201: no PUD
      Memory failure: tk->size_shift == 0
      Memory failure: Unable to find user space address edbe201 in read_poison
      Memory failure: tk->addr = 7f3cf3601000
      Memory failure: address edbe201: call dev_pagemap_mapping_shift
      Memory failure: tk->size_shift = 21
      Memory failure: 0xedbe201: forcibly killing read_poison:22434 because of failure to
unmap corrupted page
        => to deliver SIGKILL
      Memory failure: 0xedbe201: Killing read_poison:22434 due to hardware memory corruption
        => to deliver SIGBUS

    Link: http://lkml.kernel.org/r/1565112345-28754-3-git-send-email-jane.chu@oracle.com
    Signed-off-by: Jane Chu <jane.chu@oracle.com>
    Suggested-by: Naoya Horiguchi <n-horiguchi@ah.jp.nec.com>
    Reviewed-by: Dan Williams <dan.j.williams@intel.com>
    Acked-by: Naoya Horiguchi <n-horiguchi@ah.jp.nec.com>
    Cc: Michal Hocko <mhocko@kernel.org>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
```

```
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a525aafa26860ea09bca05e47fe225cf69d5bf1e
Author: David Hildenbrand <david@redhat.com>
Date:   Fri Oct 18 20:20:05 2019 -0700

    hugetlbfs: don't access uninitialized memmaps in pfn_range_valid_gigantic()

    commit f231fe4235e22e18d847e05cbe705deaca56580a upstream.

    Uninitialized memmaps contain garbage and in the worst case trigger
    kernel BUGs, especially with CONFIG_PAGE_POISONING.  They should not get
    touched.

    Let's make sure that we only consider online memory (managed by the
    buddy) that has initialized memmaps.  ZONE_DEVICE is not applicable.

    page_zone() will call page_to_nid(), which will trigger
    VM_BUG_ON_PGFLAGS(PagePoisoned(page), page) with CONFIG_PAGE_POISONING
    and CONFIG_DEBUG_VM_PGFLAGS when called on uninitialized memmaps.  This
    can be the case when an offline memory block (e.g., never onlined) is
    spanned by a zone.

    Note: As explained by Michal in [1], alloc_contig_range() will verify
    the range.  So it boils down to the wrong access in this function.

    [1] http://lkml.kernel.org/r/20180423000943.GO17484@dhcp22.suse.cz

    Link: http://lkml.kernel.org/r/20191015120717.4858-1-david@redhat.com
    Fixes: f1dd2cd13c4b ("mm, memory_hotplug: do not associate hotadded memory to zones
until online")      [visible after d0dc12e86b319]
    Signed-off-by: David Hildenbrand <david@redhat.com>
    Reported-by: Michal Hocko <mhocko@kernel.org>
    Acked-by: Michal Hocko <mhocko@suse.com>
    Reviewed-by: Mike Kravetz <mike.kravetz@oracle.com>
    Cc: Anshuman Khandual <anshuman.khandual@arm.com>
    Cc: <stable@vger.kernel.org>    [4.13+]
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b8b72c42001709d79f3dd934b3a16af65fda346e
Author: Mike Rapoport <rppt@linux.ibm.com>
Date:   Fri Oct 18 20:20:01 2019 -0700

    mm: memblock: do not enforce current limit for memblock_phys* family

    commit f3057ad767542be7bbac44e548cb44017178a163 upstream.

    Until commit 92d12f9544b7 ("memblock: refactor internal allocation
    functions") the maximal address for memblock allocations was forced to
    memblock.current_limit only for the allocation functions returning
    virtual address.  The changes introduced by that commit moved the limit
    enforcement into the allocation core and as a result the allocation
    functions returning physical address also started to limit allocations
    to memblock.current_limit.

    This caused breakage of etnaviv GPU driver:

      etnaviv etnaviv: bound 130000.gpu (ops gpu_ops)
      etnaviv etnaviv: bound 134000.gpu (ops gpu_ops)
      etnaviv etnaviv: bound 2204000.gpu (ops gpu_ops)
      etnaviv-gpu 130000.gpu: model: GC2000, revision: 5108
      etnaviv-gpu 130000.gpu: command buffer outside valid memory window
      etnaviv-gpu 134000.gpu: model: GC320, revision: 5007
```

```
       etnaviv-gpu 134000.gpu: command buffer outside valid memory window
       etnaviv-gpu 2204000.gpu: model: GC355, revision: 1215
       etnaviv-gpu 2204000.gpu: Ignoring GPU with VG and FE2.0

    Restore the behaviour of memblock_phys* family so that these functions
    will not enforce memblock.current_limit.

    Link: http://lkml.kernel.org/r/1570915861-17633-1-git-send-email-rppt@kernel.org
    Fixes: 92d12f9544b7 ("memblock: refactor internal allocation functions")
    Signed-off-by: Mike Rapoport <rppt@linux.ibm.com>
    Reported-by: Adam Ford <aford173@gmail.com>
    Tested-by: Adam Ford <aford173@gmail.com>       [imx6q-logicpd]
    Cc: Catalin Marinas <catalin.marinas@arm.com>
    Cc: Christoph Hellwig <hch@lst.de>
    Cc: Fabio Estevam <festevam@gmail.com>
    Cc: Lucas Stach <l.stach@pengutronix.de>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 681470bd95e28de580c2d6f67dc11106f6987c82
Author: Honglei Wang <honglei.wang@oracle.com>
Date:   Fri Oct 18 20:19:58 2019 -0700

    mm: memcg: get number of pages on the LRU list in memcgroup base on lru_zone_size

    commit b11edebbc967ebf5c55b8f9e1d5bb6d68ec3a7fd upstream.

    Commit 1a61ab8038e72 ("mm: memcontrol: replace zone summing with
    lruvec_page_state()") has made lruvec_page_state to use per-cpu counters
    instead of calculating it directly from lru_zone_size with an idea that
    this would be more effective.

    Tim has reported that this is not really the case for their database
    benchmark which is showing an opposite results where lruvec_page_state
    is taking up a huge chunk of CPU cycles (about 25% of the system time
    which is roughly 7% of total cpu cycles) on 5.3 kernels.  The workload
    is running on a larger machine (96cpus), it has many cgroups (500) and
    it is heavily direct reclaim bound.

    Tim Chen said:

    : The problem can also be reproduced by running simple multi-threaded
    : pmbench benchmark with a fast Optane SSD swap (see profile below).
    :
    :
    : 6.15%     3.08%  pmbench          [kernel.vmlinux]            [k] lruvec_lru_size
    :            |
    :            |--3.07%--lruvec_lru_size
    :            |          |
    :            |          |--2.11%--cpumask_next
    :            |          |          |
    :            |          |           --1.66%--find_next_bit
    :            |          |
    :            |           --0.57%--call_function_interrupt
    :            |                     |
    :            |                      --0.55%--smp_call_function_interrupt
    :            |
    :            |--1.59%--0x441f0fc3d009
    :            |          _ops_rdtsc_init_base_freq
    :            |          access_histogram
    :            |          page_fault
    :            |          __do_page_fault
    :            |          handle_mm_fault
```

```
:               |           __handle_mm_fault
:               |           |
:               |            --1.54%--do_swap_page
:               |                      swapin_readahead
:               |                      swap_cluster_readahead
:               |                      |
:               |                       --1.53%--read_swap_cache_async
:               |                                 __read_swap_cache_async
:               |                                 alloc_pages_vma
:               |                                 __alloc_pages_nodemask
:               |                                 __alloc_pages_slowpath
:               |                                 try_to_free_pages
:               |                                 do_try_to_free_pages
:               |                                 shrink_node
:               |                                 shrink_node_memcg
:               |                                 |
:               |                                 |--0.77%--lruvec_lru_size
:               |                                 |
:               |                                  --0.76%--inactive_list_is_low
:               |                                            |
:               |                                             --0.76%--lruvec_lru_size
:               |
:                --1.50%--measure_read
:                          page_fault
:                          __do_page_fault
:                          handle_mm_fault
:                          __handle_mm_fault
:                          do_swap_page
:                          swapin_readahead
:                          swap_cluster_readahead
:                          |
:                           --1.48%--read_swap_cache_async
:                                     __read_swap_cache_async
:                                     alloc_pages_vma
:                                     __alloc_pages_nodemask
:                                     __alloc_pages_slowpath
:                                     try_to_free_pages
:                                     do_try_to_free_pages
:                                     shrink_node
:                                     shrink_node_memcg
:                                     |
:                                     |--0.75%--inactive_list_is_low
:                                     |          |
:                                     |           --0.75%--lruvec_lru_size
:                                     |
:                                      --0.73%--lruvec_lru_size
```

The likely culprit is the cache traffic the lruvec_page_state_local
generates. Dave Hansen says:

```
: I was thinking purely of the cache footprint.  If it's reading
: pn->lruvec_stat_local->count[idx] is three separate cachelines, so 192
: bytes of cache *96 CPUs = 18k of data, mostly read-only.  1 cgroup would
: be 18k of data for the whole system and the caching would be pretty
: efficient and all 18k would probably survive a tight page fault loop in
: the L1.  500 cgroups would be ~90k of data per CPU thread which doesn't
: fit in the L1 and probably wouldn't survive a tight page fault loop if
: both logical threads were banging on different cgroups.
:
: It's just a theory, but it's why I noted the number of cgroups when I
: initially saw this show up in profiles
```

Fix the regression by partially reverting the said commit and calculate
the lru size explicitly.

```
    Link: http://lkml.kernel.org/r/20190905071034.16822-1-honglei.wang@oracle.com
    Fixes: 1a61ab8038e72 ("mm: memcontrol: replace zone summing with lruvec_page_state()")
    Signed-off-by: Honglei Wang <honglei.wang@oracle.com>
    Reported-by: Tim Chen <tim.c.chen@linux.intel.com>
    Acked-by: Tim Chen <tim.c.chen@linux.intel.com>
    Tested-by: Tim Chen <tim.c.chen@linux.intel.com>
    Acked-by: Michal Hocko <mhocko@suse.com>
    Cc: Vladimir Davydov <vdavydov.dev@gmail.com>
    Cc: Johannes Weiner <hannes@cmpxchg.org>
    Cc: Roman Gushchin <guro@fb.com>
    Cc: Tejun Heo <tj@kernel.org>
    Cc: Dave Hansen <dave.hansen@intel.com>
    Cc: <stable@vger.kernel.org>    [5.2+]
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 48b8ba3a73535618d1b844a594be3033548e8656
Author: Vlastimil Babka <vbabka@suse.cz>
Date:   Mon Oct 14 14:12:07 2019 -0700

    mm, compaction: fix wrong pfn handling in __reset_isolation_pfn()

    commit a2e9a5afce080226edbf1882d63d99bf32070e9e upstream.

    Florian and Dave reported [1] a NULL pointer dereference in
    __reset_isolation_pfn().  While the exact cause is unclear, staring at
    the code revealed two bugs, which might be related.

    One bug is that if zone starts in the middle of pageblock, block_page
    might correspond to different pfn than block_pfn, and then the
    pfn_valid_within() checks will check different pfn's than those accessed
    via struct page.  This might result in acessing an unitialized page in
    CONFIG_HOLES_IN_ZONE configs.

    The other bug is that end_page refers to the first page of next
    pageblock and not last page of current pageblock.  The online and valid
    check is then wrong and with sections, the while (page < end_page) loop
    might wander off actual struct page arrays.

    [1] https://lore.kernel.org/linux-xfs/87o8z1fvqu.fsf@mid.deneb.enyo.de/

    Link: http://lkml.kernel.org/r/20191008152915.24704-1-vbabka@suse.cz
    Fixes: 6b0868c820ff ("mm/compaction.c: correct zone boundary handling when resetting
pageblock skip hints")
    Signed-off-by: Vlastimil Babka <vbabka@suse.cz>
    Reported-by: Florian Weimer <fw@deneb.enyo.de>
    Reported-by: Dave Chinner <david@fromorbit.com>
    Acked-by: Mel Gorman <mgorman@techsingularity.net>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 6815408c1a20e3e68cbe5564fc6573bd7ff24582
Author: Roman Gushchin <guro@fb.com>
Date:   Fri Oct 18 20:19:44 2019 -0700

    mm: memcg/slab: fix panic in __free_slab() caused by premature memcg pointer release

    commit b749ecfaf6c53ce79d6ab66afd2fc34189a073b1 upstream.

    Karsten reported the following panic in __free_slab() happening on a s390x
    machine:
```

```
       Unable to handle kernel pointer dereference in virtual kernel address space
       Failing address: 0000000000000000 TEID: 0000000000000483
       Fault in home space mode while using kernel ASCE.
       AS:00000000017d4007 R3:000000007fbd0007 S:000000007fbff000 P:000000000000003d
       Oops: 0004 ilc:3 Ý#1¨ PREEMPT SMP
       Modules linked in: tcp_diag inet_diag xt_tcpudp ip6t_rpfilter ip6t_REJECT
 nf_reject_ipv6 ipt_REJECT nf_reject_ipv4 xt_conntrack ip6table_nat ip6table_mangle
 ip6table_raw ip6table_security iptable_at nf_nat
       CPU: 0 PID: 0 Comm: swapper/0 Not tainted 5.3.0-05872-g6133e3e4bada-dirty #14
       Hardware name: IBM 2964 NC9 702 (z/VM 6.4.0)
       Krnl PSW : 0704d00180000000 00000000003cadb6 (__free_slab+0x686/0x6b0)
                  R:0 T:1 IO:1 EX:1 Key:0 M:1 W:0 P:0 AS:3 CC:1 PM:0 RI:0 EA:3
       Krnl GPRS: 00000000f3a32928 0000000000000000 000000007fbf5d00 000000000117c4b8
                  0000000000000000 000000009e3291c1 0000000000000000 0000000000000000
                  0000000000000003 0000000000000008 000000002b478b00 000003d080a97600
                  0000000000000003 0000000000000008 000000002b478b00 000003d080a97600
                  000000000117ba00 000003e000057db0 00000000003cabcc 000003e000057c78
       Krnl Code: 00000000003cada6: e310a1400004        lg      %r1,320(%r10)
                  00000000003cadac: c0e50046c286        brasl   %r14,ca32b8
                 #00000000003cadb2: a7f4fe36            brc     15,3caa1e
                 >00000000003cadb6: e32060800024        stg     %r2,128(%r6)
                  00000000003cadbc: a7f4fd9e            brc     15,3ca8f8
                  00000000003cadc0: c0e50046790c        brasl   %r14,c99fd8
                  00000000003cadc6: a7f4fe2c            brc     15,3caa
                  00000000003cadc6: a7f4fe2c            brc     15,3caa1e
                  00000000003cadca: ecb1ffff00d9        aghik   %r11,%r1,-1
       Call Trace:
       (<00000000003cabcc> __free_slab+0x49c/0x6b0)
        <00000000001f5886> rcu_core+0x5a6/0x7e0
        <0000000000ca2dea> __do_softirq+0xf2/0x5c0
        <0000000000152644> irq_exit+0x104/0x130
        <000000000010d222> do_IRQ+0x9a/0xf0
        <0000000000ca2344> ext_int_handler+0x130/0x134
        <0000000000103648> enabled_wait+0x58/0x128
       (<0000000000103634> enabled_wait+0x44/0x128)
        <0000000000103b00> arch_cpu_idle+0x40/0x58
        <0000000000ca0544> default_idle_call+0x3c/0x68
        <000000000018eaa4> do_idle+0xec/0x1c0
        <000000000018ee0e> cpu_startup_entry+0x36/0x40
        <000000000122df34> arch_call_rest_init+0x5c/0x88
        <0000000000000000> 0x0
       INFO: lockdep is turned off.
       Last Breaking-Event-Address:
        <00000000003ca8f4> __free_slab+0x1c4/0x6b0
       Kernel panic - not syncing: Fatal exception in interrupt
```

The kernel panics on an attempt to dereference the NULL memcg pointer.
When shutdown_cache() is called from the kmem_cache_destroy() context, a
memcg kmem_cache might have empty slab pages in a partial list, which are
still charged to the memory cgroup.

These pages are released by free_partial() at the beginning of
shutdown_cache(): either directly or by scheduling a RCU-delayed work
(if the kmem_cache has the SLAB_TYPESAFE_BY_RCU flag). The latter case
is when the reported panic can happen: memcg_unlink_cache() is called
immediately after shrinking partial lists, without waiting for scheduled
RCU works. It sets the kmem_cache->memcg_params.memcg pointer to NULL,
and the following attempt to dereference it by __free_slab() from the
RCU work context causes the panic.

To fix the issue, let's postpone the release of the memcg pointer to
destroy_memcg_params(). It's called from a separate work context by
slab_caches_to_rcu_destroy_workfn(), which contains a full RCU barrier.
This guarantees that all scheduled page release RCU works will complete
before the memcg pointer will be zeroed.

```
    Big thanks for Karsten for the perfect report containing all necessary
    information, his help with the analysis of the problem and testing of the
    fix.

    Link: http://lkml.kernel.org/r/20191010160549.1584316-1-guro@fb.com
    Fixes: fb2f2b0adb98 ("mm: memcg/slab: reparent memcg kmem_caches on cgroup removal")
    Signed-off-by: Roman Gushchin <guro@fb.com>
    Reported-by: Karsten Graul <kgraul@linux.ibm.com>
    Tested-by: Karsten Graul <kgraul@linux.ibm.com>
    Acked-by: Vlastimil Babka <vbabka@suse.cz>
    Reviewed-by: Shakeel Butt <shakeelb@google.com>
    Cc: Karsten Graul <kgraul@linux.ibm.com>
    Cc: Vladimir Davydov <vdavydov.dev@gmail.com>
    Cc: David Rientjes <rientjes@google.com>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 6e5d341bd2456821aec805056ead7d18a2726a08
Author: Aneesh Kumar K.V <aneesh.kumar@linux.ibm.com>
Date:   Fri Oct 18 20:19:39 2019 -0700

    mm/memunmap: don't access uninitialized memmap in memunmap_pages()

    commit 77e080e7680e1e615587352f70c87b9e98126d03 upstream.

    Patch series "mm/memory_hotplug: Shrink zones before removing memory",
    v6.

    This series fixes the access of uninitialized memmaps when shrinking
    zones/nodes and when removing memory.  Also, it contains all fixes for
    crashes that can be triggered when removing certain namespace using
    memunmap_pages() - ZONE_DEVICE, reported by Aneesh.

    We stop trying to shrink ZONE_DEVICE, as it's buggy, fixing it would be
    more involved (we don't have SECTION_IS_ONLINE as an indicator), and
    shrinking is only of limited use (set_zone_contiguous() cannot detect
    the ZONE_DEVICE as contiguous).

    We continue shrinking !ZONE_DEVICE zones, however, I reduced the amount
    of code to a minimum.  Shrinking is especially necessary to keep
    zone->contiguous set where possible, especially, on memory unplug of
    DIMMs at zone boundaries.

    --------------------------------------------------------------------------

    Zones are now properly shrunk when offlining memory blocks or when
    onlining failed.  This allows to properly shrink zones on memory unplug
    even if the separate memory blocks of a DIMM were onlined to different
    zones or re-onlined to a different zone after offlining.

    Example:

      :/# cat /proc/zoneinfo
      Node 1, zone  Movable
              spanned  0
              present  0
              managed  0
      :/# echo "online_movable" > /sys/devices/system/memory/memory41/state
      :/# echo "online_movable" > /sys/devices/system/memory/memory43/state
      :/# cat /proc/zoneinfo
      Node 1, zone  Movable
              spanned  98304
```

```
            present  65536
            managed  65536
      :/# echo 0 > /sys/devices/system/memory/memory43/online
      :/# cat /proc/zoneinfo
      Node 1, zone  Movable
            spanned  32768
            present  32768
            managed  32768
      :/# echo 0 > /sys/devices/system/memory/memory41/online
      :/# cat /proc/zoneinfo
      Node 1, zone  Movable
            spanned  0
            present  0
            managed  0
```

This patch (of 10):

With an altmap, the memmap falling into the reserved altmap space are not initialized and, therefore, contain a garbage NID and a garbage zone. Make sure to read the NID/zone from a memmap that was initialized.

This fixes a kernel crash that is observed when destroying a namespace:

```
  kernel BUG at include/linux/mm.h:1107!
  cpu 0x1: Vector: 700 (Program Check) at [c000000274087890]
      pc: c0000000004b9728: memunmap_pages+0x238/0x340
      lr: c0000000004b9724: memunmap_pages+0x234/0x340
  ...
      pid   = 3669, comm = ndctl
  kernel BUG at include/linux/mm.h:1107!
    devm_action_release+0x30/0x50
    release_nodes+0x268/0x2d0
    device_release_driver_internal+0x174/0x240
    unbind_store+0x13c/0x190
    drv_attr_store+0x44/0x60
    sysfs_kf_write+0x70/0xa0
    kernfs_fop_write+0x1ac/0x290
    __vfs_write+0x3c/0x70
    vfs_write+0xe4/0x200
    ksys_write+0x7c/0x140
    system_call+0x5c/0x68
```

The "page_zone(pfn_to_page(pfn)" was introduced by 69324b8f4833 ("mm, devm_memremap_pages: add MEMORY_DEVICE_PRIVATE support"), however, I think we will never have driver reserved memory with MEMORY_DEVICE_PRIVATE (no altmap AFAIKS).

[david@redhat.com: minimze code changes, rephrase description]
Link: http://lkml.kernel.org/r/20191006085646.5768-2-david@redhat.com
Fixes: 2c2a5af6fed2 ("mm, memory_hotplug: add nid parameter to arch_remove_memory")
Signed-off-by: Aneesh Kumar K.V <aneesh.kumar@linux.ibm.com>
Signed-off-by: David Hildenbrand <david@redhat.com>
Cc: Dan Williams <dan.j.williams@intel.com>
Cc: Jason Gunthorpe <jgg@ziepe.ca>
Cc: Logan Gunthorpe <logang@deltatee.com>
Cc: Ira Weiny <ira.weiny@intel.com>
Cc: Damian Tometzki <damian.tometzki@gmail.com>
Cc: Alexander Duyck <alexander.h.duyck@linux.intel.com>
Cc: Alexander Potapenko <glider@google.com>
Cc: Andy Lutomirski <luto@kernel.org>
Cc: Anshuman Khandual <anshuman.khandual@arm.com>
Cc: Benjamin Herrenschmidt <benh@kernel.crashing.org>
Cc: Borislav Petkov <bp@alien8.de>
Cc: Catalin Marinas <catalin.marinas@arm.com>
Cc: Christian Borntraeger <borntraeger@de.ibm.com>

```
    Cc: Christophe Leroy <christophe.leroy@c-s.fr>
    Cc: Dave Hansen <dave.hansen@linux.intel.com>
    Cc: Fenghua Yu <fenghua.yu@intel.com>
    Cc: Gerald Schaefer <gerald.schaefer@de.ibm.com>
    Cc: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Cc: Halil Pasic <pasic@linux.ibm.com>
    Cc: Heiko Carstens <heiko.carstens@de.ibm.com>
    Cc: "H. Peter Anvin" <hpa@zytor.com>
    Cc: Ingo Molnar <mingo@redhat.com>
    Cc: Jun Yao <yaojun8558363@gmail.com>
    Cc: Mark Rutland <mark.rutland@arm.com>
    Cc: Masahiro Yamada <yamada.masahiro@socionext.com>
    Cc: "Matthew Wilcox (Oracle)" <willy@infradead.org>
    Cc: Mel Gorman <mgorman@techsingularity.net>
    Cc: Michael Ellerman <mpe@ellerman.id.au>
    Cc: Michal Hocko <mhocko@suse.com>
    Cc: Mike Rapoport <rppt@linux.ibm.com>
    Cc: Oscar Salvador <osalvador@suse.de>
    Cc: Pankaj Gupta <pagupta@redhat.com>
    Cc: Paul Mackerras <paulus@samba.org>
    Cc: Pavel Tatashin <pasha.tatashin@soleen.com>
    Cc: Pavel Tatashin <pavel.tatashin@microsoft.com>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Cc: Qian Cai <cai@lca.pw>
    Cc: Rich Felker <dalias@libc.org>
    Cc: Robin Murphy <robin.murphy@arm.com>
    Cc: Steve Capper <steve.capper@arm.com>
    Cc: Thomas Gleixner <tglx@linutronix.de>
    Cc: Tom Lendacky <thomas.lendacky@amd.com>
    Cc: Tony Luck <tony.luck@intel.com>
    Cc: Vasily Gorbik <gor@linux.ibm.com>
    Cc: Vlastimil Babka <vbabka@suse.cz>
    Cc: Wei Yang <richard.weiyang@gmail.com>
    Cc: Wei Yang <richardw.yang@linux.intel.com>
    Cc: Will Deacon <will@kernel.org>
    Cc: Yoshinori Sato <ysato@users.sourceforge.jp>
    Cc: Yu Zhao <yuzhao@google.com>
    Cc: <stable@vger.kernel.org>    [5.0+]
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit aea168ab2f1fad042e0a0f7cf3b9cab988d9dd5a
Author: Qian Cai <cai@lca.pw>
Date:   Fri Oct 18 20:19:29 2019 -0700

    mm/page_owner: don't access uninitialized memmaps when reading /proc/pagetypeinfo

    commit a26ee565b6cd8dc2bf15ff6aa70bbb28f928b773 upstream.

    Uninitialized memmaps contain garbage and in the worst case trigger
    kernel BUGs, especially with CONFIG_PAGE_POISONING.  They should not get
    touched.

    For example, when not onlining a memory block that is spanned by a zone
    and reading /proc/pagetypeinfo with CONFIG_DEBUG_VM_PGFLAGS and
    CONFIG_PAGE_POISONING, we can trigger a kernel BUG:

      :/# echo 1 > /sys/devices/system/memory/memory40/online
      :/# echo 1 > /sys/devices/system/memory/memory42/online
      :/# cat /proc/pagetypeinfo > test.file
       page:fffff2c585200000 is uninitialized and poisoned
       raw: ffffffffffffffff ffffffffffffffff ffffffffffffffff ffffffffffffffff
       raw: ffffffffffffffff ffffffffffffffff ffffffffffffffff ffffffffffffffff
       page dumped because: VM_BUG_ON_PAGE(PagePoisoned(p))
```

```
        There is not page extension available.
        ------------[ cut here ]------------
        kernel BUG at include/linux/mm.h:1107!
        invalid opcode: 0000 [#1] SMP NOPTI
```

    Please note that this change does not affect ZONE_DEVICE, because
    pagetypeinfo_showmixedcount_print() is called from
    mm/vmstat.c:pagetypeinfo_showmixedcount() only for populated zones, and
    ZONE_DEVICE is never populated (zone->present_pages always 0).

    [david@redhat.com: move check to outer loop, add comment, rephrase description]
    Link: http://lkml.kernel.org/r/20191011140638.8160-1-david@redhat.com
    Fixes: f1dd2cd13c4b ("mm, memory_hotplug: do not associate hotadded memory to zones
until online") # visible after d0dc12e86b319
    Signed-off-by: Qian Cai <cai@lca.pw>
    Signed-off-by: David Hildenbrand <david@redhat.com>
    Acked-by: Michal Hocko <mhocko@suse.com>
    Acked-by: Vlastimil Babka <vbabka@suse.cz>
    Cc: Thomas Gleixner <tglx@linutronix.de>
    Cc: "Peter Zijlstra (Intel)" <peterz@infradead.org>
    Cc: Miles Chen <miles.chen@mediatek.com>
    Cc: Mike Rapoport <rppt@linux.vnet.ibm.com>
    Cc: Qian Cai <cai@lca.pw>
    Cc: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Cc: <stable@vger.kernel.org>   [4.13+]
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 11842ccb1d862c134538f4e3d6b063215cd8f853
Author: Qian Cai <cai@lca.pw>
Date: Mon Oct 14 14:11:51 2019 -0700

    mm/slub: fix a deadlock in show_slab_objects()

    commit e4f8e513c3d353c134ad4eef9fd0bba12406c7c8 upstream.

    A long time ago we fixed a similar deadlock in show_slab_objects() [1].
    However, it is apparently due to the commits like 01fb58bcba63 ("slab:
    remove synchronous synchronize_sched() from memcg cache deactivation
    path") and 03afc0e25f7f ("slab: get_online_mems for
    kmem_cache_{create,destroy,shrink}"), this kind of deadlock is back by
    just reading files in /sys/kernel/slab which will generate a lockdep
    splat below.

    Since the "mem_hotplug_lock" here is only to obtain a stable online node
    mask while racing with NUMA node hotplug, in the worst case, the results
    may me miscalculated while doing NUMA node hotplug, but they shall be
    corrected by later reads of the same files.

```
      WARNING: possible circular locking dependency detected
      ------------------------------------------------------
      cat/5224 is trying to acquire lock:
      ffff900012ac3120 (mem_hotplug_lock.rw_sem){++++}, at:
      show_slab_objects+0x94/0x3a8

      but task is already holding lock:
      b8ff009693eee398 (kn->count#45){++++}, at: kernfs_seq_start+0x44/0xf0

      which lock already depends on the new lock.

      the existing dependency chain (in reverse order) is:

      -> #2 (kn->count#45){++++}:
             lock_acquire+0x31c/0x360
```

```
       __kernfs_remove+0x290/0x490
       kernfs_remove+0x30/0x44
       sysfs_remove_dir+0x70/0x88
       kobject_del+0x50/0xb0
       sysfs_slab_unlink+0x2c/0x38
       shutdown_cache+0xa0/0xf0
       kmemcg_cache_shutdown_fn+0x1c/0x34
       kmemcg_workfn+0x44/0x64
       process_one_work+0x4f4/0x950
       worker_thread+0x390/0x4bc
       kthread+0x1cc/0x1e8
       ret_from_fork+0x10/0x18

-> #1 (slab_mutex){+.+.}:
       lock_acquire+0x31c/0x360
       __mutex_lock_common+0x16c/0xf78
       mutex_lock_nested+0x40/0x50
       memcg_create_kmem_cache+0x38/0x16c
       memcg_kmem_cache_create_func+0x3c/0x70
       process_one_work+0x4f4/0x950
       worker_thread+0x390/0x4bc
       kthread+0x1cc/0x1e8
       ret_from_fork+0x10/0x18

-> #0 (mem_hotplug_lock.rw_sem){++++}:
       validate_chain+0xd10/0x2bcc
       __lock_acquire+0x7f4/0xb8c
       lock_acquire+0x31c/0x360
       get_online_mems+0x54/0x150
       show_slab_objects+0x94/0x3a8
       total_objects_show+0x28/0x34
       slab_attr_show+0x38/0x54
       sysfs_kf_seq_show+0x198/0x2d4
       kernfs_seq_show+0xa4/0xcc
       seq_read+0x30c/0x8a8
       kernfs_fop_read+0xa8/0x314
       __vfs_read+0x88/0x20c
       vfs_read+0xd8/0x10c
       ksys_read+0xb0/0x120
       __arm64_sys_read+0x54/0x88
       el0_svc_handler+0x170/0x240
       el0_svc+0x8/0xc

other info that might help us debug this:

Chain exists of:
  mem_hotplug_lock.rw_sem --> slab_mutex --> kn->count#45

 Possible unsafe locking scenario:

       CPU0                    CPU1
       ----                    ----
  lock(kn->count#45);
                               lock(slab_mutex);
                               lock(kn->count#45);
  lock(mem_hotplug_lock.rw_sem);

 *** DEADLOCK ***

3 locks held by cat/5224:
 #0: 9eff00095b14b2a0 (&p->lock){+.+.}, at: seq_read+0x4c/0x8a8
 #1: 0eff008997041480 (&of->mutex){+.+.}, at: kernfs_seq_start+0x34/0xf0
 #2: b8ff009693eee398 (kn->count#45){++++}, at:
kernfs_seq_start+0x44/0xf0
```

```
       stack backtrace:
       Call trace:
        dump_backtrace+0x0/0x248
        show_stack+0x20/0x2c
        dump_stack+0xd0/0x140
        print_circular_bug+0x368/0x380
        check_noncircular+0x248/0x250
        validate_chain+0xd10/0x2bcc
        __lock_acquire+0x7f4/0xb8c
        lock_acquire+0x31c/0x360
        get_online_mems+0x54/0x150
        show_slab_objects+0x94/0x3a8
        total_objects_show+0x28/0x34
        slab_attr_show+0x38/0x54
        sysfs_kf_seq_show+0x198/0x2d4
        kernfs_seq_show+0xa4/0xcc
        seq_read+0x30c/0x8a8
        kernfs_fop_read+0xa8/0x314
        __vfs_read+0x88/0x20c
        vfs_read+0xd8/0x10c
        ksys_read+0xb0/0x120
        __arm64_sys_read+0x54/0x88
        el0_svc_handler+0x170/0x240
        el0_svc+0x8/0xc

    I think it is important to mention that this doesn't expose the
    show_slab_objects to use-after-free.  There is only a single path that
    might really race here and that is the slab hotplug notifier callback
    __kmem_cache_shrink (via slab_mem_going_offline_callback) but that path
    doesn't really destroy kmem_cache_node data structures.

    [1] http://lkml.iu.edu/hypermail/linux/kernel/1101.0/02850.html

    [akpm@linux-foundation.org: add comment explaining why we don't need mem_hotplug_lock]
    Link: http://lkml.kernel.org/r/1570192309-10132-1-git-send-email-cai@lca.pw
    Fixes: 01fb58bcba63 ("slab: remove synchronous synchronize_sched() from memcg cache
deactivation path")
    Fixes: 03afc0e25f7f ("slab: get_online_mems for kmem_cache_{create,destroy,shrink}")
    Signed-off-by: Qian Cai <cai@lca.pw>
    Acked-by: Michal Hocko <mhocko@suse.com>
    Cc: Christoph Lameter <cl@linux.com>
    Cc: Pekka Enberg <penberg@kernel.org>
    Cc: David Rientjes <rientjes@google.com>
    Cc: Joonsoo Kim <iamjoonsoo.kim@lge.com>
    Cc: Tejun Heo <tj@kernel.org>
    Cc: Vladimir Davydov <vdavydov.dev@gmail.com>
    Cc: Roman Gushchin <guro@fb.com>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 0675a83aa8400854664aa11aa211c8b8b862c84f
Author: David Hildenbrand <david@redhat.com>
Date:   Fri Oct 18 20:19:23 2019 -0700

    mm/memory-failure.c: don't access uninitialized memmaps in memory_failure()

    commit 96c804a6ae8c59a9092b3d5dd581198472063184 upstream.

    We should check for pfn_to_online_page() to not access uninitialized
    memmaps.  Reshuffle the code so we don't have to duplicate the error
    message.

    Link: http://lkml.kernel.org/r/20191009142435.3975-3-david@redhat.com
```

```
    Signed-off-by: David Hildenbrand <david@redhat.com>
    Fixes: f1dd2cd13c4b ("mm, memory_hotplug: do not associate hotadded memory to zones
until online")      [visible after d0dc12e86b319]
    Acked-by: Naoya Horiguchi <n-horiguchi@ah.jp.nec.com>
    Cc: Michal Hocko <mhocko@kernel.org>
    Cc: <stable@vger.kernel.org>    [4.13+]
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 034b66868627d808f842de0b10aa55e211a400f9
Author: Faiz Abbas <faiz_abbas@ti.com>
Date:   Thu Oct 10 16:22:30 2019 +0530

    mmc: sdhci-omap: Fix Tuning procedure for temperatures < -20C

    commit feb40824d78eac5e48f56498dca941754dff33d7 upstream.

    According to the App note[1] detailing the tuning algorithm, for
    temperatures < -20C, the initial tuning value should be min(largest value
    in LPW - 24, ceil(13/16 ratio of LPW)). The largest value in LPW is
    (max_window + 4 * (max_len - 1)) and not (max_window + 4 * max_len) itself.
    Fix this implementation.

    [1] http://www.ti.com/lit/an/spraca9b/spraca9b.pdf

    Fixes: 961de0a856e3 ("mmc: sdhci-omap: Workaround errata regarding SDR104/HS200 tuning
failures (i929)")
    Cc: stable@vger.kernel.org
    Signed-off-by: Faiz Abbas <faiz_abbas@ti.com>
    Signed-off-by: Ulf Hansson <ulf.hansson@linaro.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 9857a43849dbd0ff0586fdabce75ac71e22d0d49
Author: Faiz Abbas <faiz_abbas@ti.com>
Date:   Tue Oct 15 00:08:49 2019 +0530

    mmc: cqhci: Commit descriptors before setting the doorbell

    commit c07d0073b9ec80a139d07ebf78e9c30d2a28279e upstream.

    Add a write memory barrier to make sure that descriptors are actually
    written to memory, before ringing the doorbell.

    Signed-off-by: Faiz Abbas <faiz_abbas@ti.com>
    Acked-by: Adrian Hunter <adrian.hunter@intel.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Ulf Hansson <ulf.hansson@linaro.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1e495b4e410e40945cbcf4ff5db6da0b7d602bbc
Author: Sascha Hauer <s.hauer@pengutronix.de>
Date:   Fri Oct 18 11:39:34 2019 +0200

    mmc: mxs: fix flags passed to dmaengine_prep_slave_sg

    commit 2bb9f7566ba7ab3c2154964461e37b52cdc6b91b upstream.

    Since ceeeb99cd821 we no longer abuse the DMA_CTRL_ACK flag for custom
    driver use and introduced the MXS_DMA_CTRL_WAIT4END instead. We have not
    changed all users to this flag though. This patch fixes it for the
    mxs-mmc driver.

    Fixes: ceeeb99cd821 ("dmaengine: mxs: rename custom flag")
    Signed-off-by: Sascha Hauer <s.hauer@pengutronix.de>
```

```
    Tested-by: Fabio Estevam <festevam@gmail.com>
    Reported-by: Bruno Thomsen <bruno.thomsen@gmail.com>
    Tested-by: Bruno Thomsen <bruno.thomsen@gmail.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Ulf Hansson <ulf.hansson@linaro.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit d366d732edf343f6010366c55cd26e37c4b91f0a
Author: Jens Axboe <axboe@kernel.dk>
Date:   Fri Oct 25 10:04:25 2019 -0600

    io_uring: used cached copies of sq->dropped and cq->overflow

    [ Upstream commit 498ccd9eda49117c34e0041563d0da6ac40e52b8 ]

    We currently use the ring values directly, but that can lead to issues
    if the application is malicious and changes these values on our behalf.
    Created in-kernel cached versions of them, and just overwrite the user
    side when we update them. This is similar to how we treat the sq/cq
    ring tail/head updates.

    Reported-by: Pavel Begunkov <asml.silence@gmail.com>
    Reviewed-by: Pavel Begunkov <asml.silence@gmail.com>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2da5af411f84fe26fd787202d3fe6e0423fc4216
Author: Pavel Begunkov <asml.silence@gmail.com>
Date:   Fri Oct 25 12:31:31 2019 +0300

    io_uring: Fix race for sqes with userspace

    [ Upstream commit 935d1e45908afb8853c497f2c2bbbb685dec51dc ]

    io_ring_submit() finalises with
    1. io_commit_sqring(), which releases sqes to the userspace
    2. Then calls to io_queue_link_head(), accessing released head's sqe

    Reorder them.

    Signed-off-by: Pavel Begunkov <asml.silence@gmail.com>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2df35e87786952031ffdbd5e22828c7790102c7b
Author: Pavel Begunkov <asml.silence@gmail.com>
Date:   Fri Oct 25 12:31:30 2019 +0300

    io_uring: Fix broken links with offloading

    [ Upstream commit fb5ccc98782f654778cb8d96ba8a998304f9a51f ]

    io_sq_thread() processes sqes by 8 without considering links. As a
    result, links will be randomely subdivided.

    The easiest way to fix it is to call io_get_sqring() inside
    io_submit_sqes() as do io_ring_submit().

    Downsides:
    1. This removes optimisation of not grabbing mm_struct for fixed files
    2. It submitting all sqes in one go, without finer-grained sheduling
    with cq processing.

    Signed-off-by: Pavel Begunkov <asml.silence@gmail.com>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
```

```
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 02a908123161ce81fa95620ed405d3f99374f7b0
Author: David Hildenbrand <david@redhat.com>
Date:   Fri Oct 18 20:19:20 2019 -0700

    fs/proc/page.c: don't access uninitialized memmaps in fs/proc/page.c

    commit aad5f69bc161af489dbb5934868bd347282f0764 upstream.

    There are three places where we access uninitialized memmaps, namely:
    - /proc/kpagecount
    - /proc/kpageflags
    - /proc/kpagecgroup

    We have initialized memmaps either when the section is online or when the
    page was initialized to the ZONE_DEVICE.  Uninitialized memmaps contain
    garbage and in the worst case trigger kernel BUGs, especially with
    CONFIG_PAGE_POISONING.

    For example, not onlining a DIMM during boot and calling /proc/kpagecount
    with CONFIG_PAGE_POISONING:

      :/# cat /proc/kpagecount > tmp.test
      BUG: unable to handle page fault for address: fffffffffffffffe
      #PF: supervisor read access in kernel mode
      #PF: error_code(0x0000) - not-present page
      PGD 114616067 P4D 114616067 PUD 114618067 PMD 0
      Oops: 0000 [#1] SMP NOPTI
      CPU: 0 PID: 469 Comm: cat Not tainted 5.4.0-rc1-next-20191004+ #11
      Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS rel-1.12.1-0-
ga5cab58e9a3f-prebuilt.qemu.4
      RIP: 0010:kpagecount_read+0xce/0x1e0
      Code: e8 09 83 e0 3f 48 0f a3 02 73 2d 4c 89 e7 48 c1 e7 06 48 03 3d ab 51 01 01 74 1d
48 8b 57 08 480
      RSP: 0018:ffffa14e409b7e78 EFLAGS: 00010202
      RAX: fffffffffffffffe RBX: 0000000000020000 RCX: 0000000000000000
      RDX: 0000000000000001 RSI: 00007f76b5595000 RDI: fffff35645000000
      RBP: 00007f76b5595000 R08: 0000000000000001 R09: 0000000000000000
      R10: 0000000000000000 R11: 0000000000000000 R12: 0000000000140000
      R13: 0000000000020000 R14: 00007f76b5595000 R15: ffffa14e409b7f08
      FS:  00007f76b577d580(0000) GS:ffff8f41bd400000(0000) knlGS:0000000000000000
      CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
      CR2: fffffffffffffffe CR3: 0000000078960000 CR4: 00000000000006f0
      Call Trace:
       proc_reg_read+0x3c/0x60
       vfs_read+0xc5/0x180
       ksys_read+0x68/0xe0
       do_syscall_64+0x5c/0xa0
       entry_SYSCALL_64_after_hwframe+0x49/0xbe

    For now, let's drop support for ZONE_DEVICE from the three pseudo files
    in order to fix this.  To distinguish offline memory (with garbage
    memmap) from ZONE_DEVICE memory with properly initialized memmaps, we
    would have to check get_dev_pagemap() and pfn_zone_device_reserved()
    right now.  The usage of both (especially, special casing devmem) is
    frowned upon and needs to be reworked.

    The fundamental issue we have is:

            if (pfn_to_online_page(pfn)) {
                    /* memmap initialized */
            } else if (pfn_valid(pfn)) {
                    /*
                     * ???
```

```
                        * a) offline memory. memmap garbage.
                        * b) devmem: memmap initialized to ZONE_DEVICE.
                        * c) devmem: reserved for driver. memmap garbage.
                        * (d) devmem: memmap currently initializing - garbage)
                        */
                }
```

```
    We'll leave the pfn_zone_device_reserved() check in stable_page_flags()
    in place as that function is also used from memory failure.  We now no
    longer dump information about pages that are not in use anymore -
    offline.

    Link: http://lkml.kernel.org/r/20191009142435.3975-2-david@redhat.com
    Fixes: f1dd2cd13c4b ("mm, memory_hotplug: do not associate hotadded memory to zones
until online")      [visible after d0dc12e86b319]
    Signed-off-by: David Hildenbrand <david@redhat.com>
    Reported-by: Qian Cai <cai@lca.pw>
    Acked-by: Michal Hocko <mhocko@suse.com>
    Cc: Dan Williams <dan.j.williams@intel.com>
    Cc: Alexey Dobriyan <adobriyan@gmail.com>
    Cc: Stephen Rothwell <sfr@canb.auug.org.au>
    Cc: Toshiki Fukasawa <t-fukasawa@vx.jp.nec.com>
    Cc: Pankaj gupta <pagupta@redhat.com>
    Cc: Mike Rapoport <rppt@linux.vnet.ibm.com>
    Cc: Anthony Yznaga <anthony.yznaga@oracle.com>
    Cc: "Aneesh Kumar K.V" <aneesh.kumar@linux.ibm.com>
    Cc: <stable@vger.kernel.org>    [4.13+]
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e21366484e5fee1d8018b797c0fa7ff5cfb6db18
Author: David Hildenbrand <david@redhat.com>
Date:   Fri Oct 18 20:19:16 2019 -0700

    drivers/base/memory.c: don't access uninitialized memmaps in soft_offline_page_store()

    commit 641fe2e9387a36f9ee01d7c69382d1fe147a5e98 upstream.

    Uninitialized memmaps contain garbage and in the worst case trigger kernel
    BUGs, especially with CONFIG_PAGE_POISONING.  They should not get touched.

    Right now, when trying to soft-offline a PFN that resides on a memory
    block that was never onlined, one gets a misleading error with
    CONFIG_PAGE_POISONING:

      :/# echo 5637144576 > /sys/devices/system/memory/soft_offline_page
      [   23.097167] soft offline: 0x150000 page already poisoned

    But the actual result depends on the garbage in the memmap.

    soft_offline_page() can only work with online pages, it returns -EIO in
    case of ZONE_DEVICE.  Make sure to only forward pages that are online
    (iow, managed by the buddy) and, therefore, have an initialized memmap.

    Add a check against pfn_to_online_page() and similarly return -EIO.

    Link: http://lkml.kernel.org/r/20191010141200.8985-1-david@redhat.com
    Fixes: f1dd2cd13c4b ("mm, memory_hotplug: do not associate hotadded memory to zones
until online")      [visible after d0dc12e86b319]
    Signed-off-by: David Hildenbrand <david@redhat.com>
    Acked-by: Naoya Horiguchi <n-horiguchi@ah.jp.nec.com>
    Acked-by: Michal Hocko <mhocko@suse.com>
    Cc: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Cc: "Rafael J. Wysocki" <rafael@kernel.org>
```

```
    Cc: <stable@vger.kernel.org>    [4.13+]
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3e4b0b29cbdae9706eacccbf1be2964770b3022f
Author: Philip Yang <Philip.Yang@amd.com>
Date:   Thu Oct 3 14:18:25 2019 -0400

    drm/amdgpu: user pages array memory leak fix

    commit 209620b422945ee03cebb03f726e706d537b692d upstream.

    user_pages array should always be freed after validation regardless if
    user pages are changed after bo is created because with HMM change parse
    bo always allocate user pages array to get user pages for userptr bo.

    v2: remove unused local variable and amend commit

    v3: add back get user pages in gem_userptr_ioctl, to detect application
    bug where an userptr VMA is not ananymous memory and reject it.

    Bugzilla: https://bugs.launchpad.net/ubuntu/+source/linux/+bug/1844962

    Signed-off-by: Philip Yang <Philip.Yang@amd.com>
    Tested-by: Joe Barnett <thejoe@gmail.com>
    Reviewed-by: Christian König <christian.koenig@amd.com>
    Reviewed-by: Felix Kuehling <Felix.Kuehling@amd.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Cc: stable@vger.kernel.org # 5.3
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ca8e0e7fdb88d806d2f4b4876769be43bbbdf75f
Author: Alex Deucher <alexander.deucher@amd.com>
Date:   Tue Oct 15 18:08:59 2019 -0400

    drm/amdgpu/uvd7: fix allocation size in enc ring test (v2)

    commit 5d230bc91f6c15e5d281f2851502918d98b9e770 upstream.

    We need to allocate a large enough buffer for the
    session info, otherwise the IB test can overwrite
    other memory.

    v2: - session info is 128K according to mesa
        - use the same session info for create and destroy

    Bug: https://bugzilla.kernel.org/show_bug.cgi?id=204241
    Acked-by: Christian König <christian.koenig@amd.com>
    Reviewed-by: James Zhu <James.Zhu@amd.com>
    Tested-by: James Zhu <James.Zhu@amd.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b2a79fb4e09026c52035bbed0b3e404e5a0abbf1
Author: Alex Deucher <alexander.deucher@amd.com>
Date:   Tue Oct 15 18:07:19 2019 -0400

    drm/amdgpu/uvd6: fix allocation size in enc ring test (v2)

    commit ce584a8e2885c7b59dfacba42db39761243cacb2 upstream.

    We need to allocate a large enough buffer for the
    session info, otherwise the IB test can overwrite
```

```
    other memory.

    v2: - session info is 128K according to mesa
        - use the same session info for create and destroy

    Bug: https://bugzilla.kernel.org/show_bug.cgi?id=204241
    Acked-by: Christian König <christian.koenig@amd.com>
    Reviewed-by: James Zhu <James.Zhu@amd.com>
    Tested-by: James Zhu <James.Zhu@amd.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ea387394bcf8cb79ea4313b652bd4d27de17f888
Author: Alex Deucher <alexander.deucher@amd.com>
Date:   Tue Oct 15 18:09:41 2019 -0400

    drm/amdgpu/vcn: fix allocation size in enc ring test

    commit c81fffc2c9450750dd7a54a36a788a860ab0425d upstream.

    We need to allocate a large enough buffer for the
    session info, otherwise the IB test can overwrite
    other memory.

    - Session info is 128K according to mesa
    - Use the same session info for create and destroy

    Bug: https://bugzilla.kernel.org/show_bug.cgi?id=204241
    Acked-by: Christian König <christian.koenig@amd.com>
    Reviewed-by: James Zhu <James.Zhu@amd.com>
    Tested-by: James Zhu <James.Zhu@amd.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 6fb13498c9d971dbcf733d69ea92756e2c7f4764
Author: Alex Deucher <alexander.deucher@amd.com>
Date:   Thu Oct 17 11:36:47 2019 -0400

    drm/amdgpu/vce: fix allocation size in enc ring test

    commit ee027828c40faa92a7ef4c2b0641bbb3f4be95d3 upstream.

    We need to allocate a large enough buffer for the
    feedback buffer, otherwise the IB test can overwrite
    other memory.

    Reviewed-by: James Zhu <James.Zhu@amd.com>
    Acked-by: Christian König <christian.koenig@amd.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 5e0b7f4811d296eea55626f349d1eceeb738cec5
Author: Ville Syrjälä <ville.syrjala@linux.intel.com>
Date:   Fri Oct 11 23:20:30 2019 +0300

    drm/i915: Favor last VBT child device with conflicting AUX ch/DDC pin

    commit 0336ab580878f4c5663dfa2b66095821fdc3e588 upstream.

    The first come first served apporoach to handling the VBT
    child device AUX ch conflicts has backfired. We have machines
    in the wild where the VBT specifies both port A eDP and
```

```
    port E DP (in that order) with port E being the real one.

    So let's try to flip the preference around and let the last
    child device win once again.

    Cc: stable@vger.kernel.org
    Cc: Jani Nikula <jani.nikula@intel.com>
    Tested-by: Masami Ichikawa <masami256@gmail.com>
    Tested-by: Torsten <freedesktop201910@liggy.de>
    Bugzilla: https://bugs.freedesktop.org/show_bug.cgi?id=111966
    Fixes: 36a0f92020dc ("drm/i915/bios: make child device order the priority order")
    Signed-off-by: Ville Syrjälä <ville.syrjala@linux.intel.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20191011202030.8829-1-
ville.syrjala@linux.intel.com
    Acked-by: Jani Nikula <jani.nikula@intel.com>
    (cherry picked from commit 41e35ffb380bde1379e4030bb5b2ac824d5139cf)
    Signed-off-by: Rodrigo Vivi <rodrigo.vivi@intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 7332164441a786daba25818c527df6b27324e276
Author: Chris Wilson <chris@chris-wilson.co.uk>
Date:   Sat Sep 28 09:25:46 2019 +0100

    drm/i915/userptr: Never allow userptr into the mappable GGTT

    commit 4f2a572eda67aecb1e7e4fc26cc985fb8158f6e8 upstream.

    Daniel Vetter uncovered a nasty cycle in using the mmu-notifiers to
    invalidate userptr objects which also happen to be pulled into GGTT
    mmaps. That is when we unbind the userptr object (on mmu invalidation),
    we revoke all CPU mmaps, which may then recurse into mmu invalidation.

    We looked for ways of breaking the cycle, but the revocation on
    invalidation is required and cannot be avoided. The only solution we
    could see was to not allow such GGTT bindings of userptr objects in the
    first place. In practice, no one really wants to use a GGTT mmapping of
    a CPU pointer...

    Just before Daniel's explosive lockdep patches land in v5.4-rc1, we got
    a genuine blip from CI:

    <4>[  246.793958] ======================================================
    <4>[  246.793972] WARNING: possible circular locking dependency detected
    <4>[  246.793989] 5.3.0-gbd6c56f50d15-drmtip_372+ #1 Tainted: G     U
    <4>[  246.794003] ------------------------------------------------------
    <4>[  246.794017] kswapd0/145 is trying to acquire lock:
    <4>[  246.794030] 000000003f565be6 (&dev->struct_mutex/1){+.+.}, at:
userptr_mn_invalidate_range_start+0x18f/0x220 [i915]
    <4>[  246.794250]
                      but task is already holding lock:
    <4>[  246.794263] 000000001799cef9 (&anon_vma->rwsem){++++}, at:
page_lock_anon_vma_read+0xe6/0x2a0
    <4>[  246.794291]
                      which lock already depends on the new lock.

    <4>[  246.794307]
                      the existing dependency chain (in reverse order) is:
    <4>[  246.794322]
                      -> #3 (&anon_vma->rwsem){++++}:
    <4>[  246.794344]        down_write+0x33/0x70
    <4>[  246.794357]        __vma_adjust+0x3d9/0x7b0
    <4>[  246.794370]        __split_vma+0x16a/0x180
    <4>[  246.794385]        mprotect_fixup+0x2a5/0x320
    <4>[  246.794399]        do_mprotect_pkey+0x208/0x2e0
    <4>[  246.794413]        __x64_sys_mprotect+0x16/0x20
```

```
<4>[  246.794429]        do_syscall_64+0x55/0x1c0
<4>[  246.794443]        entry_SYSCALL_64_after_hwframe+0x49/0xbe
<4>[  246.794456]
                    -> #2 (&mapping->i_mmap_rwsem){++++}:
<4>[  246.794478]        down_write+0x33/0x70
<4>[  246.794493]        unmap_mapping_pages+0x48/0x130
<4>[  246.794519]        i915_vma_revoke_mmap+0x81/0x1b0 [i915]
<4>[  246.794519]        i915_vma_unbind+0x11d/0x4a0 [i915]
<4>[  246.794519]        i915_vma_destroy+0x31/0x300 [i915]
<4>[  246.794519]        __i915_gem_free_objects+0xb8/0x4b0 [i915]
<4>[  246.794519]        drm_file_free.part.0+0x1e6/0x290
<4>[  246.794519]        drm_release+0xa6/0xe0
<4>[  246.794519]        __fput+0xc2/0x250
<4>[  246.794519]        task_work_run+0x82/0xb0
<4>[  246.794519]        do_exit+0x35b/0xdb0
<4>[  246.794519]        do_group_exit+0x34/0xb0
<4>[  246.794519]        __x64_sys_exit_group+0xf/0x10
<4>[  246.794519]        do_syscall_64+0x55/0x1c0
<4>[  246.794519]        entry_SYSCALL_64_after_hwframe+0x49/0xbe
<4>[  246.794519]
                    -> #1 (&vm->mutex){+.+.}:
<4>[  246.794519]        i915_gem_shrinker_taints_mutex+0x6d/0xe0 [i915]
<4>[  246.794519]        i915_address_space_init+0x9f/0x160 [i915]
<4>[  246.794519]        i915_ggtt_init_hw+0x55/0x170 [i915]
<4>[  246.794519]        i915_driver_probe+0xc9f/0x1620 [i915]
<4>[  246.794519]        i915_pci_probe+0x43/0x1b0 [i915]
<4>[  246.794519]        pci_device_probe+0x9e/0x120
<4>[  246.794519]        really_probe+0xea/0x3d0
<4>[  246.794519]        driver_probe_device+0x10b/0x120
<4>[  246.794519]        device_driver_attach+0x4a/0x50
<4>[  246.794519]        __driver_attach+0x97/0x130
<4>[  246.794519]        bus_for_each_dev+0x74/0xc0
<4>[  246.794519]        bus_add_driver+0x13f/0x210
<4>[  246.794519]        driver_register+0x56/0xe0
<4>[  246.794519]        do_one_initcall+0x58/0x300
<4>[  246.794519]        do_init_module+0x56/0x1f6
<4>[  246.794519]        load_module+0x25bd/0x2a40
<4>[  246.794519]        __se_sys_finit_module+0xd3/0xf0
<4>[  246.794519]        do_syscall_64+0x55/0x1c0
<4>[  246.794519]        entry_SYSCALL_64_after_hwframe+0x49/0xbe
<4>[  246.794519]
                    -> #0 (&dev->struct_mutex/1){+.+.}:
<4>[  246.794519]        __lock_acquire+0x15d8/0x1e90
<4>[  246.794519]        lock_acquire+0xa6/0x1c0
<4>[  246.794519]        __mutex_lock+0x9d/0x9b0
<4>[  246.794519]        userptr_mn_invalidate_range_start+0x18f/0x220 [i915]
<4>[  246.794519]        __mmu_notifier_invalidate_range_start+0x85/0x110
<4>[  246.794519]        try_to_unmap_one+0x76b/0x860
<4>[  246.794519]        rmap_walk_anon+0x104/0x280
<4>[  246.794519]        try_to_unmap+0xc0/0xf0
<4>[  246.794519]        shrink_page_list+0x561/0xc10
<4>[  246.794519]        shrink_inactive_list+0x220/0x440
<4>[  246.794519]        shrink_node_memcg+0x36e/0x740
<4>[  246.794519]        shrink_node+0xcb/0x490
<4>[  246.794519]        balance_pgdat+0x241/0x580
<4>[  246.794519]        kswapd+0x16c/0x530
<4>[  246.794519]        kthread+0x119/0x130
<4>[  246.794519]        ret_from_fork+0x24/0x50
<4>[  246.794519]
                    other info that might help us debug this:

<4>[  246.794519] Chain exists of:
                      &dev->struct_mutex/1 --> &mapping->i_mmap_rwsem --> &anon_vma->rwsem

<4>[  246.794519]  Possible unsafe locking scenario:
```

```
     <4>[  246.794519]        CPU0                    CPU1
     <4>[  246.794519]        ----                    ----
     <4>[  246.794519]   lock(&anon_vma->rwsem);
     <4>[  246.794519]                                lock(&mapping->i_mmap_rwsem);
     <4>[  246.794519]                                lock(&anon_vma->rwsem);
     <4>[  246.794519]   lock(&dev->struct_mutex/1);
     <4>[  246.794519]
                         *** DEADLOCK ***
```

```
     v2: Say no to mmap_ioctl

     Bugzilla: https://bugs.freedesktop.org/show_bug.cgi?id=111744
     Bugzilla: https://bugs.freedesktop.org/show_bug.cgi?id=111870
     Signed-off-by: Chris Wilson <chris@chris-wilson.co.uk>
     Cc: Tvrtko Ursulin <tvrtko.ursulin@intel.com>
     Cc: Daniel Vetter <daniel.vetter@ffwll.ch>
     Cc: stable@vger.kernel.org
     Reviewed-by: Tvrtko Ursulin <tvrtko.ursulin@intel.com>
     Link: https://patchwork.freedesktop.org/patch/msgid/20190928082546.3473-1-chris@chris-
  wilson.co.uk
     (cherry picked from commit a4311745bba9763e3c965643d4531bd5765b0513)
     Signed-off-by: Rodrigo Vivi <rodrigo.vivi@intel.com>
     Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

  commit ace574d8c1d7be0ee03f9fd13ff65b4d36157e9f
  Author: Xiaojie Yuan <xiaojie.yuan@amd.com>
  Date:   Thu Oct 10 01:01:23 2019 +0800

     drm/amdgpu/sdma5: fix mask value of POLL_REGMEM packet for pipe sync

     commit d12c50857c6edc1d18aa7a60c5a4d6d943137bc0 upstream.

     sdma will hang once sequence number to be polled reaches 0x1000_0000

     Reviewed-by: Christian König <christian.koenig@amd.com>
     Signed-off-by: Xiaojie Yuan <xiaojie.yuan@amd.com>
     Reviewed-by: Alex Deucher <alexander.deucher@amd.com>
     Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
     Cc: stable@vger.kernel.org
     Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

  commit c4e8f715f835390b383397716234deea98841f0e
  Author: Hans de Goede <hdegoede@redhat.com>
  Date:   Thu Oct 10 18:28:17 2019 +0200

     drm/amdgpu: Bail earlier when amdgpu.cik_/si_support is not set to 1

     commit 984d7a929ad68b7be9990fc9c5cfa5d5c9fc7942 upstream.

     Bail from the pci_driver probe function instead of from the drm_driver
     load function.

     This avoid /dev/dri/card0 temporarily getting registered and then
     unregistered again, sending unwanted add / remove udev events to
     userspace.

     Specifically this avoids triggering the (userspace) bug fixed by this
     plymouth merge-request:
     https://gitlab.freedesktop.org/plymouth/plymouth/merge_requests/59

     Note that despite that being a userspace bug, not sending unnecessary
     udev events is a good idea in general.

     BugLink: https://bugzilla.redhat.com/show_bug.cgi?id=1490490
```

```
    Reviewed-by: Daniel Vetter <daniel.vetter@ffwll.ch>
    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1e1c0d54c9ee561ec42f79e7c6e695be6fe23c60
Author: Steven Price <steven.price@arm.com>
Date:   Wed Oct 9 10:44:55 2019 +0100

    drm/panfrost: Handle resetting on timeout better

    commit 5b3ec8134f5f9fa1ed0a538441a495521078bbee upstream.

    Panfrost uses multiple schedulers (one for each slot, so 2 in reality),
    and on a timeout has to stop all the schedulers to safely perform a
    reset. However more than one scheduler can trigger a timeout at the same
    time. This race condition results in jobs being freed while they are
    still in use.

    When stopping other slots use cancel_delayed_work_sync() to ensure that
    any timeout started for that slot has completed. Also use
    mutex_trylock() to obtain reset_lock. This means that only one thread
    attempts the reset, the other threads will simply complete without doing
    anything (the first thread will wait for this in the call to
    cancel_delayed_work_sync()).

    While we're here and since the function is already dependent on
    sched_job not being NULL, let's remove the unnecessary checks.

    Fixes: aa20236784ab ("drm/panfrost: Prevent concurrent resets")
    Tested-by: Neil Armstrong <narmstrong@baylibre.com>
    Signed-off-by: Steven Price <steven.price@arm.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Rob Herring <robh@kernel.org>
    Link: https://patchwork.freedesktop.org/patch/msgid/20191009094456.9704-1-
steven.price@arm.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e89a4018db45e7643908f70d7d6030c6fe2818bc
Author: Thomas Hellstrom <thellstrom@vmware.com>
Date:   Thu Sep 12 20:38:54 2019 +0200

    drm/ttm: Restore ttm prefaulting

    commit 941f2f72dbbe0cf8c2d6e0b180a8021a0ec477fa upstream.

    Commit 4daa4fba3a38 ("gpu: drm: ttm: Adding new return type vm_fault_t")
    broke TTM prefaulting. Since vmf_insert_mixed() typically always returns
    VM_FAULT_NOPAGE, prefaulting stops after the second PTE.

    Restore (almost) the original behaviour. Unfortunately we can no longer
    with the new vm_fault_t return type determine whether a prefaulting
    PTE insertion hit an already populated PTE, and terminate the insertion
    loop. Instead we continue with the pre-determined number of prefaults.

    Fixes: 4daa4fba3a38 ("gpu: drm: ttm: Adding new return type vm_fault_t")
    Cc: Souptick Joarder <jrdr.linux@gmail.com>
    Cc: Christian König <christian.koenig@amd.com>
    Signed-off-by: Thomas Hellstrom <thellstrom@vmware.com>
    Reviewed-by: Christian König <christian.koenig@amd.com>
    Cc: stable@vger.kernel.org # v4.19+
    Signed-off-by: Christian König <christian.koenig@amd.com>
    Link: https://patchwork.freedesktop.org/patch/330387/
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit 2807c17a517059d1661df3b8af73fd2e65d7f386
Author: Kai-Heng Feng <kai.heng.feng@canonical.com>
Date:   Tue Apr 2 11:30:37 2019 +0800

    drm/edid: Add 6 bpc quirk for SDC panel in Lenovo G50

    commit 11bcf5f78905b90baae8fb01e16650664ed0cb00 upstream.

    Another panel that needs 6BPC quirk.

    BugLink: https://bugs.launchpad.net/bugs/1819968
    Cc: <stable@vger.kernel.org> # v4.8+
    Reviewed-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Kai-Heng Feng <kai.heng.feng@canonical.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20190402033037.21877-1-
kai.heng.feng@canonical.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 0062897bfeedbbc8164b73c0943931b50ee885aa
Author: Will Deacon <will@kernel.org>
Date:   Fri Oct 4 10:51:31 2019 +0100

    mac80211: Reject malformed SSID elements

    commit 4152561f5da3fca92af7179dd538ea89e248f9d0 upstream.

    Although this shouldn't occur in practice, it's a good idea to bounds
    check the length field of the SSID element prior to using it for things
    like allocations or memcpy operations.

    Cc: <stable@vger.kernel.org>
    Cc: Kees Cook <keescook@chromium.org>
    Reported-by: Nicolas Waisman <nico@semmle.com>
    Signed-off-by: Will Deacon <will@kernel.org>
    Link: https://lore.kernel.org/r/20191004095132.15777-1-will@kernel.org
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 9f053a9643bdc36645165b9979e849810d4063eb
Author: Will Deacon <will@kernel.org>
Date:   Fri Oct 4 10:51:32 2019 +0100

    cfg80211: wext: avoid copying malformed SSIDs

    commit 4ac2813cc867ae563a1ba5a9414bfb554e5796fa upstream.

    Ensure the SSID element is bounds-checked prior to invoking memcpy()
    with its length field, when copying to userspace.

    Cc: <stable@vger.kernel.org>
    Cc: Kees Cook <keescook@chromium.org>
    Reported-by: Nicolas Waisman <nico@semmle.com>
    Signed-off-by: Will Deacon <will@kernel.org>
    Link: https://lore.kernel.org/r/20191004095132.15777-2-will@kernel.org
    [adjust commit log a bit]
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 693675bacf8ab7841ff55360319b062a847851f7
Author: Luca Coelho <luciano.coelho@intel.com>
Date:   Tue Oct 8 13:21:02 2019 +0300

    iwlwifi: pcie: change qu with jf devices to use qu configuration
```

```
    commit aa0cc7dde17bb6b8cc533bbcfe3f53d70e0dd269 upstream.

    There were a bunch of devices with qu and jf that were loading the
    configuration with pu and jf, which is wrong.  Fix them all
    accordingly.  Additionally, remove 0x1010 and 0x1210 subsytem IDs from
    the list, since they are obviously wrong, and 0x0044 and 0x0244, which
    were duplicate.

    Cc: stable@vger.kernel.org # 5.1+
    Signed-off-by: Luca Coelho <luciano.coelho@intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 202118f99951dcb091bf1f9b2a612b94f069de56
Author: Dan Carpenter <dan.carpenter@oracle.com>
Date:   Fri Oct 18 15:35:34 2019 +0300

    ACPI: NFIT: Fix unlock on error in scrub_show()

    commit edffc70f505abdab885f4b4212438b4298dec78f upstream.

    We change the locking in this function and forgot to update this error
    path so we are accidentally still holding the "dev->lockdep_mutex".

    Fixes: 87a30e1f05d7 ("driver-core, libnvdimm: Let device subsystems add local lockdep
coverage")
    Signed-off-by: Dan Carpenter <dan.carpenter@oracle.com>
    Reviewed-by: Ira Weiny <ira.weiny@intel.com>
    Acked-by: Dan Williams <dan.j.williams@intel.com>
    Cc: 5.3+ <stable@vger.kernel.org> # 5.3+
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f1053652fbfe761697f6e20ade6b44255cfbcce7
Author: John Garry <john.garry@huawei.com>
Date:   Tue Oct 15 22:07:31 2019 +0800

    ACPI: CPPC: Set pcc_data[pcc_ss_id] to NULL in acpi_cppc_processor_exit()

    commit 56a0b978d42f58c7e3ba715cf65af487d427524d upstream.

    When enabling KASAN and DEBUG_TEST_DRIVER_REMOVE, I find this KASAN
    warning:

    [   20.872057] BUG: KASAN: use-after-free in pcc_data_alloc+0x40/0xb8
    [   20.878226] Read of size 4 at addr ffff00236cdeb684 by task swapper/0/1
    [   20.884826]
    [   20.886309] CPU: 19 PID: 1 Comm: swapper/0 Not tainted 5.4.0-rc1-00009-ge7f7df3db5bf-
dirty #289
    [   20.894994] Hardware name: Huawei D06 /D06, BIOS Hisilicon D06 UEFI RC0 - V1.16.01
03/15/2019
    [   20.903505] Call trace:
    [   20.905942]  dump_backtrace+0x0/0x200
    [   20.909593]  show_stack+0x14/0x20
    [   20.912899]  dump_stack+0xd4/0x130
    [   20.916291]  print_address_description.isra.9+0x6c/0x3b8
    [   20.921592]  __kasan_report+0x12c/0x23c
    [   20.925417]  kasan_report+0xc/0x18
    [   20.928808]  __asan_load4+0x94/0xb8
    [   20.932286]  pcc_data_alloc+0x40/0xb8
    [   20.935938]  acpi_cppc_processor_probe+0x4e8/0xb08
    [   20.940717]  __acpi_processor_start+0x48/0xb0
    [   20.945062]  acpi_processor_start+0x40/0x60
    [   20.949235]  really_probe+0x118/0x548
    [   20.952887]  driver_probe_device+0x7c/0x148
```

```
[   20.957059]  device_driver_attach+0x94/0xa0
[   20.961231]  __driver_attach+0xa4/0x110
[   20.965055]  bus_for_each_dev+0xe8/0x158
[   20.968966]  driver_attach+0x30/0x40
[   20.972531]  bus_add_driver+0x234/0x2f0
[   20.976356]  driver_register+0xbc/0x1d0
[   20.980182]  acpi_processor_driver_init+0x40/0xe4
[   20.984875]  do_one_initcall+0xb4/0x254
[   20.988700]  kernel_init_freeable+0x24c/0x2f8
[   20.993047]  kernel_init+0x10/0x118
[   20.996524]  ret_from_fork+0x10/0x18
[   21.000087]
[   21.001567] Allocated by task 1:
[   21.004785]  save_stack+0x28/0xc8
[   21.008089]  __kasan_kmalloc.isra.9+0xbc/0xd8
[   21.012435]  kasan_kmalloc+0xc/0x18
[   21.015913]  pcc_data_alloc+0x94/0xb8
[   21.019564]  acpi_cppc_processor_probe+0x4e8/0xb08
[   21.024343]  __acpi_processor_start+0x48/0xb0
[   21.028689]  acpi_processor_start+0x40/0x60
[   21.032860]  really_probe+0x118/0x548
[   21.036512]  driver_probe_device+0x7c/0x148
[   21.040684]  device_driver_attach+0x94/0xa0
[   21.044855]  __driver_attach+0xa4/0x110
[   21.048680]  bus_for_each_dev+0xe8/0x158
[   21.052591]  driver_attach+0x30/0x40
[   21.056155]  bus_add_driver+0x234/0x2f0
[   21.059980]  driver_register+0xbc/0x1d0
[   21.063805]  acpi_processor_driver_init+0x40/0xe4
[   21.068497]  do_one_initcall+0xb4/0x254
[   21.072322]  kernel_init_freeable+0x24c/0x2f8
[   21.076667]  kernel_init+0x10/0x118
[   21.080144]  ret_from_fork+0x10/0x18
[   21.083707]
[   21.085186] Freed by task 1:
[   21.088056]  save_stack+0x28/0xc8
[   21.091360]  __kasan_slab_free+0x118/0x180
[   21.095445]  kasan_slab_free+0x10/0x18
[   21.099183]  kfree+0x80/0x268
[   21.102139]  acpi_cppc_processor_exit+0x1a8/0x1b8
[   21.106832]  acpi_processor_stop+0x70/0x80
[   21.110917]  really_probe+0x174/0x548
[   21.114568]  driver_probe_device+0x7c/0x148
[   21.118740]  device_driver_attach+0x94/0xa0
[   21.122912]  __driver_attach+0xa4/0x110
[   21.126736]  bus_for_each_dev+0xe8/0x158
[   21.130648]  driver_attach+0x30/0x40
[   21.134212]  bus_add_driver+0x234/0x2f0
[   21.0x10/0x18
[   21.161764]
[   21.163244] The buggy address belongs to the object at ffff00236cdeb600
[   21.163244]  which belongs to the cache kmalloc-256 of size 256
[   21.175750] The buggy address is located 132 bytes inside of
[   21.175750]  256-byte region [ffff00236cdeb600, ffff00236cdeb700)
[   21.187473] The buggy address belongs to the page:
[   21.192254] page:fffffe008d937a00 refcount:1 mapcount:0 mapping:ffff002370c0fa00
index:0x0 compound_mapcount: 0
[   21.202331] flags: 0x1ffff00000010200(slab|head)
[   21.206940] raw: 1ffff00000010200 dead000000000100 dead000000000122 ffff002370c0fa00
[   21.214671] raw: 0000000000000000 00000000802a002a 00000001ffffffff 0000000000000000
[   21.222400] page dumped because: kasan: bad access detected
[   21.227959]
[   21.229438] Memory state around the buggy address:
[   21.234218]  ffff00236cdeb580: fc fc fc fc fc fc fc fc fc fc fc fc fc fc fc fc
[   21.241427]  ffff00236cdeb600: fb fb fb fb fb fb fb fb fb fb fb fb fb fb fb fb
```

```
[   21.248637] >ffff00236cdeb680: fb fb fb fb fb fb fb fb fb fb fb fb fb fb fb fb
[   21.255845]                    ^
[   21.259062]  ffff00236cdeb700: fc fc fc fc fc fc fc fc fc fc fc fc fc fc fc fc
[   21.266272]  ffff00236cdeb780: fb fb fb fb fb fb fb fb fb fb fb fb fb fb fb fb
[   21.273480] ============================================================================
```

It seems that global pcc_data[pcc_ss_id] can be freed in
acpi_cppc_processor_exit(), but we may later reference this value, so
NULLify it when freed.

Also remove the useless setting of data "pcc_channel_acquired", which
we're about to free.

Fixes: 85b1407bf6d2 ("ACPI / CPPC: Make CPPC ACPI driver aware of PCC subspace IDs")
Signed-off-by: John Garry <john.garry@huawei.com>
Cc: 4.15+ <stable@vger.kernel.org> # 4.15+
Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 4799fff7bc08251aa6f2aab4fe4f8c2e26471869
Author: Junya Monden <jmonden@jp.adit-jv.com>
Date: Wed Oct 16 14:42:55 2019 +0200

ASoC: rsnd: Reinitialize bit clock inversion flag for every format setting

commit 22e58665a01006d05f0239621f7d41cacca96cc4 upstream.

Unlike other format-related DAI parameters, rdai->bit_clk_inv flag
is not properly re-initialized when setting format for new stream
processing. The inversion, if requested, is then applied not to default,
but to a previous value, which leads to SCKP bit in SSICR register being
set incorrectly.
Fix this by re-setting the flag to its initial value, determined by format.

Fixes: 1a7889ca8aba3 ("ASoC: rsnd: fixup SND_SOC_DAIFMT_xB_xF behavior")
Cc: Andrew Gabbasov <andrew_gabbasov@mentor.com>
Cc: Jiada Wang <jiada_wang@mentor.com>
Cc: Timo Wischer <twischer@de.adit-jv.com>
Cc: stable@vger.kernel.org # v3.17+
Signed-off-by: Junya Monden <jmonden@jp.adit-jv.com>
Signed-off-by: Eugeniu Rosca <erosca@de.adit-jv.com>
Acked-by: Kuninori Morimoto <kuninori.morimoto.gx@renesas.com>
Link: https://lore.kernel.org/r/20191016124255.7442-1-erosca@de.adit-jv.com
Signed-off-by: Mark Brown <broonie@kernel.org>
Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b8eda9dc727799e92c2c96c46550d1e619c9f276
Author: Dixit Parmar <dixitparmar19@gmail.com>
Date: Mon Oct 21 09:32:47 2019 -0700

Input: st1232 - fix reporting multitouch coordinates

commit b1a402e75a5f5127ff1ffff0615249f98df8b7b3 upstream.

For Sitronix st1633 multi-touch controller driver the coordinates reported
for multiple fingers were wrong, as it was always taking LSB of coordinates
from the first contact data.

Signed-off-by: Dixit Parmar <dixitparmar19@gmail.com>
Reviewed-by: Martin Kepplinger <martink@posteo.de>
Cc: stable@vger.kernel.org
Fixes: 351e0592bfea ("Input: st1232 - add support for st1633")
Bugzilla: https://bugzilla.kernel.org/show_bug.cgi?id=204561
Link: https://lore.kernel.org/r/1566209314-21767-1-git-send-email-
dixitparmar19@gmail.com

```
    Signed-off-by: Dmitry Torokhov <dmitry.torokhov@gmail.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 12d5e63257396c94c852c32c8a721c5dd971b65d
Author: Evan Green <evgreen@chromium.org>
Date:   Fri Oct 11 17:22:09 2019 -0700

    Input: synaptics-rmi4 - avoid processing unknown IRQs

    commit 363c53875aef8fce69d4a2d0873919ccc7d9e2ad upstream.

    rmi_process_interrupt_requests() calls handle_nested_irq() for
    each interrupt status bit it finds. If the irq domain mapping for
    this bit had not yet been set up, then it ends up calling
    handle_nested_irq(0), which causes a NULL pointer dereference.

    There's already code that masks the irq_status bits coming out of the
    hardware with current_irq_mask, presumably to avoid this situation.
    However current_irq_mask seems to more reflect the actual mask set
    in the hardware rather than the IRQs software has set up and registered
    for. For example, in rmi_driver_reset_handler(), the current_irq_mask
    is initialized based on what is read from the hardware. If the reset
    value of this mask enables IRQs that Linux has not set up yet, then
    we end up in this situation.

    There appears to be a third unused bitmask that used to serve this
    purpose, fn_irq_bits. Use that bitmask instead of current_irq_mask
    to avoid calling handle_nested_irq() on IRQs that have not yet been
    set up.

    Signed-off-by: Evan Green <evgreen@chromium.org>
    Reviewed-by: Andrew Duggan <aduggan@synaptics.com>
    Link: https://lore.kernel.org/r/20191008223657.163366-1-evgreen@chromium.org
    Cc: stable@vger.kernel.org
    Signed-off-by: Dmitry Torokhov <dmitry.torokhov@gmail.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ff05c6759645e18c227904abb36f53f9f44c42d6
Author: Marco Felsch <m.felsch@pengutronix.de>
Date:   Mon Sep 16 12:45:48 2019 -0700

    Input: da9063 - fix capability and drop KEY_SLEEP

    commit afce285b859cea91c182015fc9858ea58c26cd0e upstream.

    Since commit f889beaaab1c ("Input: da9063 - report KEY_POWER instead of
    KEY_SLEEP during power key-press") KEY_SLEEP isn't supported anymore. This
    caused input device to not generate any events if "dlg,disable-key-power"
    is set.

    Fix this by unconditionally setting KEY_POWER capability, and not
    declaring KEY_SLEEP.

    Fixes: f889beaaab1c ("Input: da9063 - report KEY_POWER instead of KEY_SLEEP during power
key-press")
    Signed-off-by: Marco Felsch <m.felsch@pengutronix.de>
    Cc: stable@vger.kernel.org
    Signed-off-by: Dmitry Torokhov <dmitry.torokhov@gmail.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 0776a251f8773341a1f9a21a04646a69a60f4a92
Author: Kai-Heng Feng <kai.heng.feng@canonical.com>
Date:   Tue Oct 15 17:37:37 2019 -0700

    Revert "Input: elantech - enable SMBus on new (2018+) systems"
```

```
    commit c324345ce89c3cc50226372960619c7ee940f616 upstream.

    This reverts commit 883a2a80f79ca5c0c105605fafabd1f3df99b34c.

    Apparently use dmi_get_bios_year() as manufacturing date isn't accurate
    and this breaks older laptops with new BIOS update.

    So let's revert this patch.

    There are still new HP laptops still need to use SMBus to support all
    features, but it'll be enabled via a whitelist.

    Signed-off-by: Kai-Heng Feng <kai.heng.feng@canonical.com>
    Acked-by: Benjamin Tissoires <benjamin.tissoires@redhat.com>
    Cc: stable@vger.kernel.org
    Link: https://lore.kernel.org/r/20191001070845.9720-1-kai.heng.feng@canonical.com
    Signed-off-by: Dmitry Torokhov <dmitry.torokhov@gmail.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 34a6ba0a0225efa21de60070a86e5aa03c8e477d
Author: Bart Van Assche <bvanassche@acm.org>
Date:   Wed Oct 9 10:35:36 2019 -0700

    scsi: ch: Make it possible to open a ch device multiple times again

    commit 6a0990eaa768dfb7064f06777743acc6d392084b upstream.

    Clearing ch->device in ch_release() is wrong because that pointer must
    remain valid until ch_remove() is called. This patch fixes the following
    crash the second time a ch device is opened:

    BUG: kernel NULL pointer dereference, address: 0000000000000790
    RIP: 0010:scsi_device_get+0x5/0x60
    Call Trace:
     ch_open+0x4c/0xa0 [ch]
     chrdev_open+0xa2/0x1c0
     do_dentry_open+0x13a/0x380
     path_openat+0x591/0x1470
     do_filp_open+0x91/0x100
     do_sys_open+0x184/0x220
     do_syscall_64+0x5f/0x1a0
     entry_SYSCALL_64_after_hwframe+0x44/0xa9

    Fixes: 085e56766f74 ("scsi: ch: add refcounting")
    Cc: Hannes Reinecke <hare@suse.de>
    Cc: <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20191009173536.247889-1-bvanassche@acm.org
    Reported-by: Rob Turk <robtu@rtist.nl>
    Suggested-by: Rob Turk <robtu@rtist.nl>
    Signed-off-by: Bart Van Assche <bvanassche@acm.org>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 8c94bafc9667c5d40e0f3b4d009a01a2b130f66c
Author: Yufen Yu <yuyufen@huawei.com>
Date:   Tue Oct 15 21:05:56 2019 +0800

    scsi: core: try to get module before removing device

    commit 77c301287ebae86cc71d03eb3806f271cb14da79 upstream.

    We have a test case like block/001 in blktests, which will create a scsi
    device by loading scsi_debug module and then try to delete the device by
    sysfs interface. At the same time, it may remove the scsi_debug module.
```

```
    And getting a invalid paging request BUG_ON as following:

    [   34.625854] BUG: unable to handle page fault for address: ffffffffa0016bb8
    [   34.629189] Oops: 0000 [#1] SMP PTI
    [   34.629618] CPU: 1 PID: 450 Comm: bash Tainted: G        W         5.4.0-rc3+ #473
    [   34.632524] RIP: 0010:scsi_proc_hostdir_rm+0x5/0xa0
    [   34.643555] CR2: ffffffffa0016bb8 CR3: 000000012cd88000 CR4: 00000000000006e0
    [   34.644545] Call Trace:
    [   34.644907]  scsi_host_dev_release+0x6b/0x1f0
    [   34.645511]  device_release+0x74/0x110
    [   34.646046]  kobject_put+0x116/0x390
    [   34.646559]  put_device+0x17/0x30
    [   34.647041]  scsi_target_dev_release+0x2b/0x40
    [   34.647652]  device_release+0x74/0x110
    [   34.648186]  kobject_put+0x116/0x390
    [   34.648691]  put_device+0x17/0x30
    [   34.649157]  scsi_device_dev_release_usercontext+0x2e8/0x360
    [   34.649953]  execute_in_process_context+0x29/0x80
    [   34.650603]  scsi_device_dev_release+0x20/0x30
    [   34.651221]  device_release+0x74/0x110
    [   34.651732]  kobject_put+0x116/0x390
    [   34.652230]  sysfs_unbreak_active_protection+0x3f/0x50
    [   34.652935]  sdev_store_delete.cold.4+0x71/0x8f
    [   34.653579]  dev_attr_store+0x1b/0x40
    [   34.654103]  sysfs_kf_write+0x3d/0x60
    [   34.654603]  kernfs_fop_write+0x174/0x250
    [   34.655165]  __vfs_write+0x1f/0x60
    [   34.655639]  vfs_write+0xc7/0x280
    [   34.656117]  ksys_write+0x6d/0x140
    [   34.656591]  __x64_sys_write+0x1e/0x30
    [   34.657114]  do_syscall_64+0xb1/0x400
    [   34.657627]  entry_SYSCALL_64_after_hwframe+0x44/0xa9
    [   34.658335] RIP: 0033:0x7f156f337130

    During deleting scsi target, the scsi_debug module have been removed. Then,
    sdebug_driver_template belonged to the module cannot be accessd, resulting
    in scsi_proc_hostdir_rm() BUG_ON.

    To fix the bug, we add scsi_device_get() in sdev_store_delete() to try to
    increase refcount of module, avoiding the module been removed.

    Cc: stable@vger.kernel.org
    Link: https://lore.kernel.org/r/20191015130556.18061-1-yuyufen@huawei.com
    Signed-off-by: Yufen Yu <yuyufen@huawei.com>
    Reviewed-by: Bart Van Assche <bvanassche@acm.org>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 0b3ebcf54f832b9497afdb086893853007efa86b
Author: Damien Le Moal <damien.lemoal@wdc.com>
Date:   Tue Oct 1 16:48:39 2019 +0900

    scsi: core: save/restore command resid for error handling

    commit 8f8fed0cdbbd6cdbf28d9ebe662f45765d2f7d39 upstream.

    When a non-passthrough command is terminated with CHECK CONDITION, request
    sense is executed by hijacking the command descriptor. Since
    scsi_eh_prep_cmnd() and scsi_eh_restore_cmnd() do not save/restore the
    original command resid, the value returned on failure of the original
    command is lost and replaced with the value set by the execution of the
    request sense command. This value may in many instances be unaligned to the
    device sector size, causing sd_done() to print a warning message about the
    incorrect unaligned resid before the command is retried.
```

```
    Fix this problem by saving the original command residual in struct
    scsi_eh_save using scsi_eh_prep_cmnd() and restoring it in
    scsi_eh_restore_cmnd(). In addition, to make sure that the request sense
    command is executed with a correctly initialized command structure, also
    reset the residual to 0 in scsi_eh_prep_cmnd() after saving the original
    command value in struct scsi_eh_save.

    Cc: stable@vger.kernel.org
    Link: https://lore.kernel.org/r/20191001074839.1994-1-damien.lemoal@wdc.com
    Signed-off-by: Damien Le Moal <damien.lemoal@wdc.com>
    Reviewed-by: Bart Van Assche <bvanassche@acm.org>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 03f1f054a03372a27c3db810656e5efe49f55e30
Author: Oliver Neukum <oneukum@suse.com>
Date:   Tue Sep 3 12:18:39 2019 +0200

    scsi: sd: Ignore a failure to sync cache due to lack of authorization

    commit 21e3d6c81179bbdfa279efc8de456c34b814cfd2 upstream.

    I've got a report about a UAS drive enclosure reporting back Sense: Logical
    unit access not authorized if the drive it holds is password protected.
    While the drive is obviously unusable in that state as a mass storage
    device, it still exists as a sd device and when the system is asked to
    perform a suspend of the drive, it will be sent a SYNCHRONIZE CACHE. If
    that fails due to password protection, the error must be ignored.

    Cc: <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20190903101840.16483-1-oneukum@suse.com
    Signed-off-by: Oliver Neukum <oneukum@suse.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e6e9380e65b8dc304466da3f3b144dfa80e2105d
Author: Steffen Maier <maier@linux.ibm.com>
Date:   Tue Oct 1 12:49:49 2019 +0200

    scsi: zfcp: fix reaction on bit error threshold notification

    commit 2190168aaea42c31bff7b9a967e7b045f07df095 upstream.

    On excessive bit errors for the FCP channel ingress fibre path, the channel
    notifies us.  Previously, we only emitted a kernel message and a trace
    record.  Since performance can become suboptimal with I/O timeouts due to
    bit errors, we now stop using an FCP device by default on channel
    notification so multipath on top can timely failover to other paths.  A new
    module parameter zfcp.ber_stop can be used to get zfcp old behavior.

    User explanation of new kernel message:

     * Description:
     * The FCP channel reported that its bit error threshold has been exceeded.
     * These errors might result from a problem with the physical components
     * of the local fibre link into the FCP channel.
     * The problem might be damage or malfunction of the cable or
     * cable connection between the FCP channel and
     * the adjacent fabric switch port or the point-to-point peer.
     * Find details about the errors in the HBA trace for the FCP device.
     * The zfcp device driver closed down the FCP device
     * to limit the performance impact from possible I/O command timeouts.
     * User action:
     * Check for problems on the local fibre link, ensure that fibre optics are
```

```
      * clean and functional, and all cables are properly plugged.
      * After the repair action, you can manually recover the FCP device by
      * writing "0" into its "failed" sysfs attribute.
      * If recovery through sysfs is not possible, set the CHPID of the device
      * offline and back online on the service element.

    Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
    Cc: <stable@vger.kernel.org> #2.6.30+
    Link: https://lore.kernel.org/r/20191001104949.42810-1-maier@linux.ibm.com
    Reviewed-by: Jens Remus <jremus@linux.ibm.com>
    Reviewed-by: Benjamin Block <bblock@linux.ibm.com>
    Signed-off-by: Steffen Maier <maier@linux.ibm.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3ca11291533abe62fc88d384b9bfd68bcec5c478
Author: Colin Ian King <colin.king@canonical.com>
Date:   Mon Oct 14 12:02:01 2019 +0100

    staging: wlan-ng: fix exit return when sme->key_idx >= NUM_WEPKEYS

    commit 153c5d8191c26165dbbd2646448ca7207f7796d0 upstream.

    Currently the exit return path when sme->key_idx >= NUM_WEPKEYS is via
    label 'exit' and this checks if result is non-zero, however result has
    not been initialized and contains garbage.  Fix this by replacing the
    goto with a return with the error code.

    Addresses-Coverity: ("Uninitialized scalar variable")
    Fixes: 0ca6d8e74489 ("Staging: wlan-ng: replace switch-case statements with macro")
    Signed-off-by: Colin Ian King <colin.king@canonical.com>
    Cc: stable <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20191014110201.9874-1-colin.king@canonical.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ab0d2face1c0d467aab851127aeb52278a39d23b
Author: Paul Burton <paulburton@kernel.org>
Date:   Fri Oct 18 15:38:48 2019 -0700

    MIPS: tlbex: Fix build_restore_pagemask KScratch restore

    commit b42aa3fd5957e4daf4b69129e5ce752a2a53e7d6 upstream.

    build_restore_pagemask() will restore the value of register $1/$at when
    its restore_scratch argument is non-zero, and aims to do so by filling a
    branch delay slot. Commit 0b24cae4d535 ("MIPS: Add missing EHB in mtc0
    -> mfc0 sequence.") added an EHB instruction (Execution Hazard Barrier)
    prior to restoring $1 from a KScratch register, in order to resolve a
    hazard that can result in stale values of the KScratch register being
    observed. In particular, P-class CPUs from MIPS with out of order
    execution pipelines such as the P5600 & P6600 are affected.

    Unfortunately this EHB instruction was inserted in the branch delay slot
    causing the MFC0 instruction which performs the restoration to no longer
    execute along with the branch. The result is that the $1 register isn't
    actually restored, ie. the TLB refill exception handler clobbers it -
    which is exactly the problem the EHB is meant to avoid for the P-class
    CPUs.

    Similarly build_get_pgd_vmalloc() will restore the value of $1/$at when
    its mode argument equals refill_scratch, and suffers from the same
    problem.

    Fix this by in both cases moving the EHB earlier in the emitted code.
    There's no reason it needs to immediately precede the MFC0 - it simply
```

```
    needs to be between the MTC0 & MFC0.

    This bug only affects Cavium Octeon systems which use
    build_fast_tlb_refill_handler().

    Signed-off-by: Paul Burton <paulburton@kernel.org>
    Fixes: 0b24cae4d535 ("MIPS: Add missing EHB in mtc0 -> mfc0 sequence.")
    Cc: Dmitry Korotin <dkorotin@wavecomp.com>
    Cc: stable@vger.kernel.org # v3.15+
    Cc: linux-mips@vger.kernel.org
    Cc: linux-kernel@vger.kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 7dc081b3021e3cd544ebdfa5f7be650dc28bbe85
Author: Jann Horn <jannh@google.com>
Date:   Wed Oct 16 17:01:18 2019 +0200

    binder: Don't modify VMA bounds in ->mmap handler

    commit 45d02f79b539073b76077836871de6b674e36eb4 upstream.

    binder_mmap() tries to prevent the creation of overly big binder mappings
    by silently truncating the size of the VMA to 4MiB. However, this violates
    the API contract of mmap(). If userspace attempts to create a large binder
    VMA, and later attempts to unmap that VMA, it will call munmap() on a range
    beyond the end of the VMA, which may have been allocated to another VMA in
    the meantime. This can lead to userspace memory corruption.

    The following sequence of calls leads to a segfault without this commit:

    int main(void) {
      int binder_fd = open("/dev/binder", O_RDWR);
      if (binder_fd == -1) err(1, "open binder");
      void *binder_mapping = mmap(NULL, 0x800000UL, PROT_READ, MAP_SHARED,
                                  binder_fd, 0);
      if (binder_mapping == MAP_FAILED) err(1, "mmap binder");
      void *data_mapping = mmap(NULL, 0x400000UL, PROT_READ|PROT_WRITE,
                                MAP_PRIVATE|MAP_ANONYMOUS, -1, 0);
      if (data_mapping == MAP_FAILED) err(1, "mmap data");
      munmap(binder_mapping, 0x800000UL);
      *(char*)data_mapping = 1;
      return 0;
    }

    Cc: stable@vger.kernel.org
    Signed-off-by: Jann Horn <jannh@google.com>
    Acked-by: Todd Kjos <tkjos@google.com>
    Acked-by: Christian Brauner <christian.brauner@ubuntu.com>
    Link: https://lore.kernel.org/r/20191016150119.154756-1-jannh@google.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1339e279a497881f10055bff7e286d5dd4fe77de
Author: Johan Hovold <johan@kernel.org>
Date:   Fri Oct 18 17:19:54 2019 +0200

    USB: ldusb: fix read info leaks

    commit 7a6f22d7479b7a0b68eadd308a997dd64dda7dae upstream.

    Fix broken read implementation, which could be used to trigger slab info
    leaks.

    The driver failed to check if the custom ring buffer was still empty
    when waking up after having waited for more data. This would happen on
    every interrupt-in completion, even if no data had been added to the
```

```
    ring buffer (e.g. on disconnect events).

    Due to missing sanity checks and uninitialised (kmalloced) ring-buffer
    entries, this meant that huge slab info leaks could easily be triggered.

    Note that the empty-buffer check after wakeup is enough to fix the info
    leak on disconnect, but let's clear the buffer on allocation and add a
    sanity check to read() to prevent further leaks.

    Fixes: 2824bd250f0b ("[PATCH] USB: add ldusb driver")
    Cc: stable <stable@vger.kernel.org>     # 2.6.13
    Reported-by: syzbot+6fe95b826644f7f12b0b@syzkaller.appspotmail.com
    Signed-off-by: Johan Hovold <johan@kernel.org>
    Link: https://lore.kernel.org/r/20191018151955.25135-2-johan@kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 99c6e67ef3343f4035af06264956c57477e8ee14
Author: Johan Hovold <johan@kernel.org>
Date:   Tue Oct 15 19:55:22 2019 +0200

    USB: usblp: fix use-after-free on disconnect

    commit 7a759197974894213621aa65f0571b51904733d6 upstream.

    A recent commit addressing a runtime PM use-count regression, introduced
    a use-after-free by not making sure we held a reference to the struct
    usb_interface for the lifetime of the driver data.

    Fixes: 9a31535859bf ("USB: usblp: fix runtime PM after driver unbind")
    Cc: stable <stable@vger.kernel.org>
    Reported-by: syzbot+cd24df4d075c319ebfc5@syzkaller.appspotmail.com
    Signed-off-by: Johan Hovold <johan@kernel.org>
    Link: https://lore.kernel.org/r/20191015175522.18490-1-johan@kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 25660fdaf1de024988316b6b9a0d3d6394bcde6f
Author: Johan Hovold <johan@kernel.org>
Date:   Thu Oct 10 14:58:34 2019 +0200

    USB: ldusb: fix memleak on disconnect

    commit b14a39048c1156cfee76228bf449852da2f14df8 upstream.

    If disconnect() races with release() after a process has been
    interrupted, release() could end up returning early and the driver would
    fail to free its driver data.

    Fixes: 2824bd250f0b ("[PATCH] USB: add ldusb driver")
    Cc: stable <stable@vger.kernel.org>     # 2.6.13
    Signed-off-by: Johan Hovold <johan@kernel.org>
    Link: https://lore.kernel.org/r/20191010125835.27031-2-johan@kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit de83289d42bb1c4a6c1ce2386241977a76653007
Author: Johan Hovold <johan@kernel.org>
Date:   Fri Oct 11 11:57:35 2019 +0200

    USB: serial: ti_usb_3410_5052: fix port-close races

    commit 6f1d1dc8d540a9aa6e39b9cb86d3a67bbc1c8d8d upstream.

    Fix races between closing a port and opening or closing another port on
    the same device which could lead to a failure to start or stop the
    shared interrupt URB. The latter could potentially cause a
    use-after-free or worse in the completion handler on driver unbind.
```

```
    Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
    Cc: stable <stable@vger.kernel.org>
    Signed-off-by: Johan Hovold <johan@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 5ca15922a65f69d66cb03311c8a6920b92ac33e9
Author: Gustavo A. R. Silva <gustavo@embeddedor.com>
Date:   Mon Oct 14 14:18:30 2019 -0500

    usb: udc: lpc32xx: fix bad bit shift operation

    commit b987b66ac3a2bc2f7b03a0ba48a07dc553100c07 upstream.

    It seems that the right variable to use in this case is *i*, instead of
    *n*, otherwise there is an undefined behavior when right shifiting by more
    than 31 bits when multiplying n by 8; notice that *n* can take values
    equal or greater than 4 (4, 8, 16, ...).

    Also, notice that under the current conditions (bl = 3), we are skiping
    the handling of bytes 3, 7, 31... So, fix this by updating this logic
    and limit *bl* up to 4 instead of up to 3.

    This fix is based on function udc_stuff_fifo().

    Addresses-Coverity-ID: 1454834 ("Bad bit shift operation")
    Fixes: 24a28e428351 ("USB: gadget driver for LPC32xx")
    Cc: stable@vger.kernel.org
    Signed-off-by: Gustavo A. R. Silva <gustavo@embeddedor.com>
    Link: https://lore.kernel.org/r/20191014191830.GA10721@embeddedor
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3f5fa0ba267074fe39d4cd56f34d873064350911
Author: Lukas Wunner <lukas@wunner.de>
Date:   Thu Oct 17 17:04:11 2019 +0200

    ALSA: hda - Force runtime PM on Nvidia HDMI codecs

    commit 94989e318b2f11e217e86bee058088064fa9a2e9 upstream.

    Przemysław Kopa reports that since commit b516ea586d71 ("PCI: Enable
    NVIDIA HDA controllers"), the discrete GPU Nvidia GeForce GT 540M on his
    2011 Samsung laptop refuses to runtime suspend, resulting in a power
    regression and excessive heat.

    Rivera Valdez witnesses the same issue with a GeForce GT 525M (GF108M)
    of the same era, as does another Arch Linux user named "R0AR" with a
    more recent GeForce GTX 1050 Ti (GP107M).

    The commit exposes the discrete GPU's HDA controller and all four codecs
    on the controller do not set the CLKSTOP and EPSS bits in the Supported
    Power States Response.  They also do not set the PS-ClkStopOk bit in the
    Get Power State Response.  hda_codec_runtime_suspend() therefore does
    not call snd_hdac_codec_link_down(), which prevents each codec and the
    PCI device from runtime suspending.

    The same issue is present on some AMD discrete GPUs and we addressed it
    by forcing runtime PM despite the bits not being set, see commit
    57cb54e53bdd ("ALSA: hda - Force to link down at runtime suspend on
    ATI/AMD HDMI").

    Do the same for Nvidia HDMI codecs.

    Fixes: b516ea586d71 ("PCI: Enable NVIDIA HDA controllers")
    Link: https://bbs.archlinux.org/viewtopic.php?pid=1865512
```

```
    Link: https://bugs.freedesktop.org/show_bug.cgi?id=75985#c81
    Reported-by: Przemysław Kopa <prymoo@gmail.com>
    Reported-by: Rivera Valdez <riveravaldez@ysinembargo.com>
    Signed-off-by: Lukas Wunner <lukas@wunner.de>
    Cc: Daniel Drake <dan@reactivated.net>
    Cc: stable@vger.kernel.org # v5.3+
    Link:
https://lore.kernel.org/r/3086bc75135c1e3567c5bc4f3cc4ff5cbf7a56c2.1571324194.git.lukas@wunn
er.de
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 7b5061f59f68bb19d7937f5bac9bb6708b5f3a35
Author: Szabolcs Szőke <szszoke.code@gmail.com>
Date:   Fri Oct 11 19:19:36 2019 +0200

    ALSA: usb-audio: Disable quirks for BOSS Katana amplifiers

    commit 7571b6a17fcc5e4f6903f065a82d0e38011346ed upstream.

    BOSS Katana amplifiers cannot be used for recording or playback if quirks
    are applied

    BugLink: https://bugzilla.kernel.org/show_bug.cgi?id=195223
    Signed-off-by: Szabolcs Szőke <szszoke.code@gmail.com>
    Cc: <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20191011171937.8013-1-szszoke.code@gmail.com
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 40c35b6ce147efd11f6b026db7c1bccbf01d9cec
Author: Daniel Drake <drake@endlessm.com>
Date:   Thu Oct 17 16:15:01 2019 +0800

    ALSA: hda/realtek - Enable headset mic on Asus MJ401TA

    commit 8c8967a7dc01a25f57a0757fdca10987773cd1f2 upstream.

    On Asus MJ401TA (with Realtek ALC256), the headset mic is connected to
    pin 0x19, with default configuration value 0x411111f0 (indicating no
    physical connection).

    Enable this by quirking the pin. Mic jack detection was also tested and
    found to be working.

    This enables use of the headset mic on this product.

    Signed-off-by: Daniel Drake <drake@endlessm.com>
    Cc: <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20191017081501.17135-1-drake@endlessm.com
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 35f83d26c374134409a91f5fa404256a0c7a6a3f
Author: Kailang Yang <kailang@realtek.com>
Date:   Thu May 2 16:03:26 2019 +0800

    ALSA: hda/realtek - Add support for ALC711

    commit 83629532ce45ef9df1f297b419b9ea112045685d upstream.

    Support new codec ALC711.

    Signed-off-by: Kailang Yang <kailang@realtek.com>
    Cc: <stable@vger.kernel.org>
```

```
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 9f902499690bcd1523394b5242292702dbc09a1b
Author: Johan Hovold <johan@kernel.org>
Date:   Thu Oct 10 14:58:35 2019 +0200

    USB: legousbtower: fix memleak on disconnect

    commit b6c03e5f7b463efcafd1ce141bd5a8fc4e583ae2 upstream.

    If disconnect() races with release() after a process has been
    interrupted, release() could end up returning early and the driver would
    fail to free its driver data.

    Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
    Cc: stable <stable@vger.kernel.org>
    Signed-off-by: Johan Hovold <johan@kernel.org>
    Link: https://lore.kernel.org/r/20191010125835.27031-3-johan@kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b7897a36d0f96b944bf14888679eca8eda0164ea
Author: Pavel Begunkov <asml.silence@gmail.com>
Date:   Fri Oct 25 12:31:29 2019 +0300

    io_uring: Fix corrupted user_data

    commit 84d55dc5b9e57b513a702fbc358e1b5489651590 upstream.

    There is a bug, where failed linked requests are returned not with
    specified @user_data, but with garbage from a kernel stack.

    The reason is that io_fail_links() uses req->user_data, which is
    uninitialised when called from io_queue_sqe() on fail path.

    Signed-off-by: Pavel Begunkov <asml.silence@gmail.com>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ab03d9b7899ee979ee99b66e9cb42c065da08769
Author: Jens Axboe <axboe@kernel.dk>
Date:   Fri Oct 25 10:06:15 2019 -0600

    io_uring: fix bad inflight accounting for SETUP_IOPOLL|SETUP_SQTHREAD

    commit 2b2ed9750fc9d040b9f6d076afcef6f00b6f1f7c upstream.

    We currently assume that submissions from the sqthread are successful,
    and if IO polling is enabled, we use that value for knowing how many
    completions to look for. But if we overflowed the CQ ring or some
    requests simply got errored and already completed, they won't be
    available for polling.

    For the case of IO polling and SQTHREAD usage, look at the pending
    poll list. If it ever hits empty then we know that we don't have
    anymore pollable requests inflight. For that case, simply reset
    the inflight count to zero.

    Reported-by: Pavel Begunkov <asml.silence@gmail.com>
    Reviewed-by: Pavel Begunkov <asml.silence@gmail.com>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 35b0eff1997f6783085e7d43e199ebafcda63650
Author: Eric Dumazet <edumazet@google.com>
```

```
Date:   Mon Oct 14 06:04:38 2019 -0700

    rxrpc: use rcu protection while reading sk->sk_user_data

    [ Upstream commit 2ca4f6ca4562594ef161e4140c2a5e0e5282967b ]

    We need to extend the rcu_read_lock() section in rxrpc_error_report()
    and use rcu_dereference_sk_user_data() instead of plain access
    to sk->sk_user_data to make sure all rules are respected.

    The compiler wont reload sk->sk_user_data at will, and RCU rules
    prevent memory beeing freed too soon.

    Fixes: f0308fb07080 ("rxrpc: Fix possible NULL pointer access in ICMP handling")
    Fixes: 17926a79320a ("[AF_RXRPC]: Provide secure RxRPC sockets for use by userspace and
kernel both")
    Signed-off-by: Eric Dumazet <edumazet@google.com>
    Cc: David Howells <dhowells@redhat.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8cd0912adf398c7b2ee1a154368f56a43de12494
Author: Micah Morton <mortonm@chromium.org>
Date:   Tue Sep 17 11:27:05 2019 -0700

    LSM: SafeSetID: Stop releasing uninitialized ruleset

    [ Upstream commit 21ab8580b383f27b7f59b84ac1699cb26d6c3d69 ]

    The first time a rule set is configured for SafeSetID, we shouldn't be
    trying to release the previously configured ruleset, since there isn't
    one. Currently, the pointer that would point to a previously configured
    ruleset is uninitialized on first rule set configuration, leading to a
    crash when we try to call release_ruleset with that pointer.

    Acked-by: Jann Horn <jannh@google.com>
    Signed-off-by: Micah Morton <mortonm@chromium.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7a5c15a0a905897636933357cb5752af9b13a5a5
Author: Yonglong Liu <liuyonglong@huawei.com>
Date:   Wed Oct 16 10:30:39 2019 +0800

    net: phy: Fix "link partner" information disappear issue

    [ Upstream commit 3de5ae54712c75cf3c517a288e0a704784ec6cf5 ]

    Some drivers just call phy_ethtool_ksettings_set() to set the
    links, for those phy drivers that use genphy_read_status(), if
    autoneg is on, and the link is up, than execute "ethtool -s
    ethx autoneg on" will cause "link partner" information disappear.

    The call trace is phy_ethtool_ksettings_set()->phy_start_aneg()
    ->linkmode_zero(phydev->lp_advertising)->genphy_read_status(),
    the link didn't change, so genphy_read_status() just return, and
    phydev->lp_advertising is zero now.

    This patch moves the clear operation of lp_advertising from
    phy_start_aneg() to genphy_read_lpa()/genphy_c45_read_lpa(), and
    if autoneg on and autoneg not complete, just clear what the
    generic functions care about.

    Fixes: 88d6272acaaa ("net: phy: avoid unneeded MDIO reads in genphy_read_status")
    Signed-off-by: Yonglong Liu <liuyonglong@huawei.com>
    Reviewed-by: Heiner Kallweit <hkallweit1@gmail.com>
```

```
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 50efedbf77433a9dc9cbc9a1c5f941499874c11d
Author: Randy Dunlap <rdunlap@infradead.org>
Date:   Fri Oct 11 21:03:33 2019 -0700

    net: ethernet: broadcom: have drivers select DIMLIB as needed

    [ Upstream commit ddc790e92b3afa4e366ffb41818cfcd19015031e ]

    NET_VENDOR_BROADCOM is intended to control a kconfig menu only.
    It should not have anything to do with code generation.
    As such, it should not select DIMLIB for all drivers under
    NET_VENDOR_BROADCOM.  Instead each driver that needs DIMLIB should
    select it (being the symbols SYSTEMPORT, BNXT, and BCMGENET).

    Link: https://lkml.kernel.org/r/alpine.DEB.2.21.1907021810220.13058@ramsan.of.borg/

    Fixes: 4f75da3666c0 ("linux/dim: Move implementation to .c files")
    Reported-by: Geert Uytterhoeven <geert@linux-m68k.org>
    Signed-off-by: Randy Dunlap <rdunlap@infradead.org>
    Cc: Uwe Kleine-König <uwe@kleine-koenig.org>
    Cc: Tal Gilboa <talgi@mellanox.com>
    Cc: Saeed Mahameed <saeedm@mellanox.com>
    Cc: netdev@vger.kernel.org
    Cc: linux-rdma@vger.kernel.org
    Cc: "David S. Miller" <davem@davemloft.net>
    Cc: Jakub Kicinski <jakub.kicinski@netronome.com>
    Cc: Doug Ledford <dledford@redhat.com>
    Cc: Jason Gunthorpe <jgg@mellanox.com>
    Cc: Leon Romanovsky <leonro@mellanox.com>
    Cc: Or Gerlitz <ogerlitz@mellanox.com>
    Cc: Sagi Grimberg <sagi@grimberg.me>
    Acked-by: Florian Fainelli <f.fainelli@gmail.com>
    Reviewed-by: Leon Romanovsky <leonro@mellanox.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit d3d24a38c6708043f5580b92f24734960a1e8414
Author: YueHaibing <yuehaibing@huawei.com>
Date:   Fri Oct 11 17:46:53 2019 +0800

    netdevsim: Fix error handling in nsim_fib_init and nsim_fib_exit

    [ Upstream commit 33902b4a4227877896dd9368ac10f4ca0d100de5 ]

    In nsim_fib_init(), if register_fib_notifier failed, nsim_fib_net_ops
    should be unregistered before return.

    In nsim_fib_exit(), unregister_fib_notifier should be called before
    nsim_fib_net_ops be unregistered, otherwise may cause use-after-free:

    BUG: KASAN: use-after-free in nsim_fib_event_nb+0x342/0x570 [netdevsim]
    Read of size 8 at addr ffff8881daaf4388 by task kworker/0:3/3499

    CPU: 0 PID: 3499 Comm: kworker/0:3 Not tainted 5.3.0-rc7+ #30
    Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS 1.10.2-1ubuntu1 04/01/2014
    Workqueue: ipv6_addrconf addrconf_dad_work [ipv6]
    Call Trace:
     __dump_stack lib/dump_stack.c:77 [inline]
     dump_stack+0xa9/0x10e lib/dump_stack.c:113
     print_address_description+0x65/0x380 mm/kasan/report.c:351
     __kasan_report+0x149/0x18d mm/kasan/report.c:482
     kasan_report+0xe/0x20 mm/kasan/common.c:618
```

```
       nsim_fib_event_nb+0x342/0x570 [netdevsim]
       notifier_call_chain+0x52/0xf0 kernel/notifier.c:95
       __atomic_notifier_call_chain+0x78/0x140 kernel/notifier.c:185
       call_fib_notifiers+0x30/0x60 net/core/fib_notifier.c:30
       call_fib6_entry_notifiers+0xc1/0x100 [ipv6]
       fib6_add+0x92e/0x1b10 [ipv6]
       __ip6_ins_rt+0x40/0x60 [ipv6]
       ip6_ins_rt+0x84/0xb0 [ipv6]
       __ipv6_ifa_notify+0x4b6/0x550 [ipv6]
       ipv6_ifa_notify+0xa5/0x180 [ipv6]
       addrconf_dad_completed+0xca/0x640 [ipv6]
       addrconf_dad_work+0x296/0x960 [ipv6]
       process_one_work+0x5c0/0xc00 kernel/workqueue.c:2269
       worker_thread+0x5c/0x670 kernel/workqueue.c:2415
       kthread+0x1d7/0x200 kernel/kthread.c:255
       ret_from_fork+0x3a/0x50 arch/x86/entry/entry_64.S:352

      Allocated by task 3388:
       save_stack+0x19/0x80 mm/kasan/common.c:69
       set_track mm/kasan/common.c:77 [inline]
       __kasan_kmalloc.constprop.3+0xa0/0xd0 mm/kasan/common.c:493
       kmalloc include/linux/slab.h:557 [inline]
       kzalloc include/linux/slab.h:748 [inline]
       ops_init+0xa9/0x220 net/core/net_namespace.c:127
       __register_pernet_operations net/core/net_namespace.c:1135 [inline]
       register_pernet_operations+0x1d4/0x420 net/core/net_namespace.c:1212
       register_pernet_subsys+0x24/0x40 net/core/net_namespace.c:1253
       nsim_fib_init+0x12/0x70 [netdevsim]
       veth_get_link_ksettings+0x2b/0x50 [veth]
       do_one_initcall+0xd4/0x454 init/main.c:939
       do_init_module+0xe0/0x330 kernel/module.c:3490
       load_module+0x3c2f/0x4620 kernel/module.c:3841
       __do_sys_finit_module+0x163/0x190 kernel/module.c:3931
       do_syscall_64+0x72/0x2e0 arch/x86/entry/common.c:296
       entry_SYSCALL_64_after_hwframe+0x49/0xbe

      Freed by task 3534:
       save_stack+0x19/0x80 mm/kasan/common.c:69
       set_track mm/kasan/common.c:77 [inline]
       __kasan_slab_free+0x130/0x180 mm/kasan/common.c:455
       slab_free_hook mm/slub.c:1423 [inline]
       slab_free_freelist_hook mm/slub.c:1474 [inline]
       slab_free mm/slub.c:3016 [inline]
       kfree+0xe9/0x2d0 mm/slub.c:3957
       ops_free net/core/net_namespace.c:151 [inline]
       ops_free_list.part.7+0x156/0x220 net/core/net_namespace.c:184
       ops_free_list net/core/net_namespace.c:182 [inline]
       __unregister_pernet_operations net/core/net_namespace.c:1165 [inline]
       unregister_pernet_operations+0x221/0x2a0 net/core/net_namespace.c:1224
       unregister_pernet_subsys+0x1d/0x30 net/core/net_namespace.c:1271
       nsim_fib_exit+0x11/0x20 [netdevsim]
       nsim_module_exit+0x16/0x21 [netdevsim]
       __do_sys_delete_module kernel/module.c:1015 [inline]
       __se_sys_delete_module kernel/module.c:958 [inline]
       __x64_sys_delete_module+0x244/0x330 kernel/module.c:958
       do_syscall_64+0x72/0x2e0 arch/x86/entry/common.c:296
       entry_SYSCALL_64_after_hwframe+0x49/0xbe

      Reported-by: Hulk Robot <hulkci@huawei.com>
      Fixes: 59c84b9fcf42 ("netdevsim: Restore per-network namespace accounting for fib
  entries")
      Signed-off-by: YueHaibing <yuehaibing@huawei.com>
      Acked-by: Jakub Kicinski <jakub.kicinski@netronome.com>
      Signed-off-by: David S. Miller <davem@davemloft.net>
      Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit 749b915a0e7d40946efc81ff961afc714a9956fb
Author: Davide Caratti <dcaratti@redhat.com>
Date:   Sat Oct 12 13:55:07 2019 +0200

    net/sched: fix corrupted L2 header with MPLS 'push' and 'pop' actions

    [ Upstream commit fa4e0f8855fcba600e0be2575ee29c69166f74bd ]

    the following script:

     # tc qdisc add dev eth0 clsact
     # tc filter add dev eth0 egress protocol ip matchall \
     > action mpls push protocol mpls_uc label 0x355aa bos 1

    causes corruption of all IP packets transmitted by eth0. On TC egress, we
    can't rely on the value of skb->mac_len, because it's 0 and a MPLS 'push'
    operation will result in an overwrite of the first 4 octets in the packet
    L2 header (e.g. the Destination Address if eth0 is an Ethernet); the same
    error pattern is present also in the MPLS 'pop' operation. Fix this error
    in act_mpls data plane, computing 'mac_len' as the difference between the
    network header and the mac header (when not at TC ingress), and use it in
    MPLS 'push'/'pop' core functions.

    v2: unbreak 'make htmldocs' because of missing documentation of 'mac_len'
        in skb_mpls_pop(), reported by kbuild test robot

    CC: Lorenzo Bianconi <lorenzo@kernel.org>
    Fixes: 2a2ea50870ba ("net: sched: add mpls manipulation actions to TC")
    Reviewed-by: Simon Horman <simon.horman@netronome.com>
    Acked-by: John Hurley <john.hurley@netronome.com>
    Signed-off-by: Davide Caratti <dcaratti@redhat.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 7e8f23c0c20cca0a46b896a8d6586dd825236489
Author: Davide Caratti <dcaratti@redhat.com>
Date:   Sat Oct 12 13:55:06 2019 +0200

    net: avoid errors when trying to pop MLPS header on non-MPLS packets

    [ Upstream commit dedc5a08da07874c6e0d411e7f39c5c2cf137014 ]

    the following script:

     # tc qdisc add dev eth0 clsact
     # tc filter add dev eth0 egress matchall action mpls pop

    implicitly makes the kernel drop all packets transmitted by eth0, if they
    don't have a MPLS header. This behavior is uncommon: other encapsulations
    (like VLAN) just let the packet pass unmodified. Since the result of MPLS
    'pop' operation would be the same regardless of the presence / absence of
    MPLS header(s) in the original packet, we can let skb_mpls_pop() return 0
    when dealing with non-MPLS packets.

    For the OVS use-case, this is acceptable because __ovs_nla_copy_actions()
    already ensures that MPLS 'pop' operation only occurs with packets having
    an MPLS Ethernet type (and there are no other callers in current code, so
    the semantic change should be ok).

    v2: better documentation of use-cases for skb_mpls_pop(), thanks to Simon
        Horman

    Fixes: 2a2ea50870ba ("net: sched: add mpls manipulation actions to TC")
    Reviewed-by: Simon Horman <simon.horman@netronome.com>
```

```
    Acked-by: John Hurley <john.hurley@netronome.com>
    Signed-off-by: Davide Caratti <dcaratti@redhat.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 472606ec96fcf8411121a293a075fcd6197a1b99
Author: Marek Vasut <marex@denx.de>
Date:   Wed Oct 16 15:35:07 2019 +0200

    net: phy: micrel: Update KSZ87xx PHY name

    [ Upstream commit 1d951ba3da67bbc7a9b0e05987e09552c2060e18 ]

    The KSZ8795 PHY ID is in fact used by KSZ8794/KSZ8795/KSZ8765 switches.
    Update the PHY ID and name to reflect that, as this family of switches
    is commonly refered to as KSZ87xx

    Signed-off-by: Marek Vasut <marex@denx.de>
    Cc: Andrew Lunn <andrew@lunn.ch>
    Cc: David S. Miller <davem@davemloft.net>
    Cc: Florian Fainelli <f.fainelli@gmail.com>
    Cc: George McCollister <george.mccollister@gmail.com>
    Cc: Heiner Kallweit <hkallweit1@gmail.com>
    Cc: Sean Nyekjaer <sean.nyekjaer@prevas.dk>
    Cc: Tristram Ha <Tristram.Ha@microchip.com>
    Cc: Woojung Huh <woojung.huh@microchip.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 6fc762b2d4b32c47fdb7003f36914ddbd86ba80e
Author: Marek Vasut <marex@denx.de>
Date:   Wed Oct 16 15:35:06 2019 +0200

    net: phy: micrel: Discern KSZ8051 and KSZ8795 PHYs

    [ Upstream commit 8b95599c55ed24b36cf44a4720067cfe67edbcb4 ]

    The KSZ8051 PHY and the KSZ8794/KSZ8795/KSZ8765 switch share exactly the
    same PHY ID. Since KSZ8051 is higher in the ksphy_driver[] list of PHYs
    in the micrel PHY driver, it is used even with the KSZ87xx switch. This
    is wrong, since the KSZ8051 configures registers of the PHY which are
    not present on the simplified KSZ87xx switch PHYs and misconfigures
    other registers of the KSZ87xx switch PHYs.

    Fortunatelly, it is possible to tell apart the KSZ8051 PHY from the
    KSZ87xx switch by checking the Basic Status register Bit 0, which is
    read-only and indicates presence of the Extended Capability Registers.
    The KSZ8051 PHY has those registers while the KSZ87xx switch does not.

    This patch implements simple check for the presence of this bit for
    both the KSZ8051 PHY and KSZ87xx switch, to let both use the correct
    PHY driver instance.

    Fixes: 9d162ed69f51 ("net: phy: micrel: add support for KSZ8795")
    Signed-off-by: Marek Vasut <marex@denx.de>
    Cc: Andrew Lunn <andrew@lunn.ch>
    Cc: David S. Miller <davem@davemloft.net>
    Cc: Florian Fainelli <f.fainelli@gmail.com>
    Cc: George McCollister <george.mccollister@gmail.com>
    Cc: Heiner Kallweit <hkallweit1@gmail.com>
    Cc: Sean Nyekjaer <sean.nyekjaer@prevas.dk>
    Cc: Tristram Ha <Tristram.Ha@microchip.com>
    Cc: Woojung Huh <woojung.huh@microchip.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit ab235356a824cbe57825a5f5c998284af8d03607
Author: Dmitry Bogdanov <dmitry.bogdanov@aquantia.com>
Date:   Fri Oct 11 13:45:23 2019 +0000

    net: aquantia: correctly handle macvlan and multicast coexistence

    [ Upstream commit 9f051db566da1e8110659ab4ab188af1c2510bb4 ]

    macvlan and multicast handling is now mixed up.
    The explicit issue is that macvlan interface gets broken (no traffic)
    after clearing MULTICAST flag on the real interface.

    We now do separate logic and consider both ALLMULTI and MULTICAST
    flags on the device.

    Fixes: 11ba961c9161 ("net: aquantia: Fix IFF_ALLMULTI flag functionality")
    Signed-off-by: Dmitry Bogdanov <dmitry.bogdanov@aquantia.com>
    Signed-off-by: Igor Russkikh <igor.russkikh@aquantia.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 031b228b134e606c3ce16ad779d4963409052f27
Author: Dmitry Bogdanov <dmitry.bogdanov@aquantia.com>
Date:   Fri Oct 11 13:45:22 2019 +0000

    net: aquantia: do not pass lro session with invalid tcp checksum

    [ Upstream commit d08b9a0a3ebdf71b0aabe576c7dd48e57e80e0f0 ]

    Individual descriptors on LRO TCP session should be checked
    for CRC errors. It was discovered that HW recalculates
    L4 checksums on LRO session and does not break it up on bad L4
    csum.

    Thus, driver should aggregate HW LRO L4 statuses from all individual
    buffers of LRO session and drop packet if one of the buffers has bad
    L4 checksum.

    Fixes: f38f1ee8aeb2 ("net: aquantia: check rx csum for all packets in LRO session")
    Signed-off-by: Dmitry Bogdanov <dmitry.bogdanov@aquantia.com>
    Signed-off-by: Igor Russkikh <igor.russkikh@aquantia.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 6e6738a74b6befec2d514d62f45720ba8f639677
Author: Igor Russkikh <Igor.Russkikh@aquantia.com>
Date:   Fri Oct 11 13:45:20 2019 +0000

    net: aquantia: when cleaning hw cache it should be toggled

    [ Upstream commit ed4d81c4b3f28ccf624f11fd66f67aec5b58859c ]

    >From HW specification to correctly reset HW caches (this is a required
    workaround when stopping the device), register bit should actually
    be toggled.

    It was previosly always just set. Due to the way driver stops HW this
    never actually caused any issues, but it still may, so cleaning this up.

    Fixes: 7a1bb49461b1 ("net: aquantia: fix potential IOMMU fault after driver unbind")
    Signed-off-by: Igor Russkikh <igor.russkikh@aquantia.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit 60cbcb2d493ffecafc4f6c8b1dd528bb965ca631
Author: Igor Russkikh <Igor.Russkikh@aquantia.com>
Date:   Fri Oct 11 13:45:19 2019 +0000

    net: aquantia: temperature retrieval fix

    [ Upstream commit 06b0d7fe7e5ff3ba4c7e265ef41135e8bcc232bb ]

    Chip temperature is a two byte word, colocated internally with cable
    length data. We do all readouts from HW memory by dwords, thus
    we should clear extra high bytes, otherwise temperature output
    gets weird as soon as we attach a cable to the NIC.

    Fixes: 8f8940118654 ("net: aquantia: add infrastructure to readout chip temperature")
    Tested-by: Holger Hoffstätte <holger@applied-asynchrony.com>
    Signed-off-by: Igor Russkikh <igor.russkikh@aquantia.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 291593baabf76e04f9943044792a2d9d0bfe0a66
Author: Xin Long <lucien.xin@gmail.com>
Date:   Tue Oct 15 15:24:38 2019 +0800

    sctp: change sctp_prot .no_autobind with true

    [ Upstream commit 63dfb7938b13fa2c2fbcb45f34d065769eb09414 ]

    syzbot reported a memory leak:

      BUG: memory leak, unreferenced object 0xffff888120b3d380 (size 64):
      backtrace:

        [...] slab_alloc mm/slab.c:3319 [inline]
        [...] kmem_cache_alloc+0x13f/0x2c0 mm/slab.c:3483
        [...] sctp_bucket_create net/sctp/socket.c:8523 [inline]
        [...] sctp_get_port_local+0x189/0x5a0 net/sctp/socket.c:8270
        [...] sctp_do_bind+0xcc/0x200 net/sctp/socket.c:402
        [...] sctp_bindx_add+0x4b/0xd0 net/sctp/socket.c:497
        [...] sctp_setsockopt_bindx+0x156/0x1b0 net/sctp/socket.c:1022
        [...] sctp_setsockopt net/sctp/socket.c:4641 [inline]
        [...] sctp_setsockopt+0xaea/0x2dc0 net/sctp/socket.c:4611
        [...] sock_common_setsockopt+0x38/0x50 net/core/sock.c:3147
        [...] __sys_setsockopt+0x10f/0x220 net/socket.c:2084
        [...] __do_sys_setsockopt net/socket.c:2100 [inline]

    It was caused by when sending msgs without binding a port, in the path:
    inet_sendmsg() -> inet_send_prepare() -> inet_autobind() ->
    .get_port/sctp_get_port(), sp->bind_hash will be set while bp->port is
    not. Later when binding another port by sctp_setsockopt_bindx(), a new
    bucket will be created as bp->port is not set.

    sctp's autobind is supposed to call sctp_autobind() where it does all
    things including setting bp->port. Since sctp_autobind() is called in
    sctp_sendmsg() if the sk is not yet bound, it should have skipped the
    auto bind.

    THis patch is to avoid calling inet_autobind() in inet_send_prepare()
    by changing sctp_prot .no_autobind with true, also remove the unused
    .get_port.

    Reported-by: syzbot+d44f7bbebdea49dbc84a@syzkaller.appspotmail.com
    Signed-off-by: Xin Long <lucien.xin@gmail.com>
    Acked-by: Marcelo Ricardo Leitner <marcelo.leitner@gmail.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit 47f268b5d9ee0b88eae2a86826111555c196768a
Author: Vinicius Costa Gomes <vinicius.gomes@intel.com>
Date:   Mon Oct 14 13:38:22 2019 -0700

    sched: etf: Fix ordering of packets with same txtime

    [ Upstream commit 28aa7c86c2b49f659c8460a89e53b506c45979bb ]

    When a application sends many packets with the same txtime, they may
    be transmitted out of order (different from the order in which they
    were enqueued).

    This happens because when inserting elements into the tree, when the
    txtime of two packets are the same, the new packet is inserted at the
    left side of the tree, causing the reordering. The only effect of this
    change should be that packets with the same txtime will be transmitted
    in the order they are enqueued.

    The application in question (the AVTP GStreamer plugin, still in
    development) is sending video traffic, in which each video frame have
    a single presentation time, the problem is that when packetizing,
    multiple packets end up with the same txtime.

    The receiving side was rejecting packets because they were being
    received out of order.

    Fixes: 25db26a91364 ("net/sched: Introduce the ETF Qdisc")
    Reported-by: Ederson de Souza <ederson.desouza@intel.com>
    Signed-off-by: Vinicius Costa Gomes <vinicius.gomes@intel.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit bc1e8b3451476cecadfcf38ab0ec4dda9a231822
Author: David Howells <dhowells@redhat.com>
Date:   Thu Oct 10 15:52:34 2019 +0100

    rxrpc: Fix possible NULL pointer access in ICMP handling

    [ Upstream commit f0308fb0708078d6c1d8a4d533941a7a191af634 ]

    If an ICMP packet comes in on the UDP socket backing an AF_RXRPC socket as
    the UDP socket is being shut down, rxrpc_error_report() may get called to
    deal with it after sk_user_data on the UDP socket has been cleared, leading
    to a NULL pointer access when this local endpoint record gets accessed.

    Fix this by just returning immediately if sk_user_data was NULL.

    The oops looks like the following:

    #PF: supervisor read access in kernel mode
    #PF: error_code(0x0000) - not-present page
    ...
    RIP: 0010:rxrpc_error_report+0x1bd/0x6a9
    ...
    Call Trace:
     ? sock_queue_err_skb+0xbd/0xde
     ? __udp4_lib_err+0x313/0x34d
     __udp4_lib_err+0x313/0x34d
     icmp_unreach+0x1ee/0x207
     icmp_rcv+0x25b/0x28f
     ip_protocol_deliver_rcu+0x95/0x10e
     ip_local_deliver+0xe9/0x148
     __netif_receive_skb_one_core+0x52/0x6e
     process_backlog+0xdc/0x177
```

```
      net_rx_action+0xf9/0x270
      __do_softirq+0x1b6/0x39a
      ? smpboot_register_percpu_thread+0xce/0xce
      run_ksoftirqd+0x1d/0x42
      smpboot_thread_fn+0x19e/0x1b3
      kthread+0xf1/0xf6
      ? kthread_delayed_work_timer_fn+0x83/0x83
      ret_from_fork+0x24/0x30

    Fixes: 17926a79320a ("[AF_RXRPC]: Provide secure RxRPC sockets for use by userspace and
kernel both")
    Reported-by: syzbot+611164843bd48cc2190c@syzkaller.appspotmail.com
    Signed-off-by: David Howells <dhowells@redhat.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a7fbbbbd17c1eb283d304a3fb5b6335780d7af17
Author: Biao Huang <biao.huang@mediatek.com>
Date:   Tue Oct 15 11:24:44 2019 +0800

    net: stmmac: disable/enable ptp_ref_clk in suspend/resume flow

    [ Upstream commit e497c20e203680aba9ccf7bb475959595908ca7e ]

    disable ptp_ref_clk in suspend flow, and enable it in resume flow.

    Fixes: f573c0b9c4e0 ("stmmac: move stmmac_clk, pclk, clk_ptp_ref and stmmac_rst to
platform structure")
    Signed-off-by: Biao Huang <biao.huang@mediatek.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit af239a783632f69b88573de506ca13d126554d6c
Author: Xin Long <lucien.xin@gmail.com>
Date:   Fri Aug 23 19:33:03 2019 +0800

    net: ipv6: fix listify ip6_rcv_finish in case of forwarding

    [ Upstream commit c7a42eb49212f93a800560662d17d5293960d3c3 ]

    We need a similar fix for ipv6 as Commit 0761680d5215 ("net: ipv4: fix
    listify ip_rcv_finish in case of forwarding") does for ipv4.

    This issue can be reprocuded by syzbot since Commit 323ebb61e32b ("net:
    use listified RX for handling GRO_NORMAL skbs") on net-next. The call
    trace was:

      kernel BUG at include/linux/skbuff.h:2225!
      RIP: 0010:__skb_pull include/linux/skbuff.h:2225 [inline]
      RIP: 0010:skb_pull+0xea/0x110 net/core/skbuff.c:1902
      Call Trace:
        sctp_inq_pop+0x2f1/0xd80 net/sctp/inqueue.c:202
        sctp_endpoint_bh_rcv+0x184/0x8d0 net/sctp/endpointola.c:385
        sctp_inq_push+0x1e4/0x280 net/sctp/inqueue.c:80
        sctp_rcv+0x2807/0x3590 net/sctp/input.c:256
        sctp6_rcv+0x17/0x30 net/sctp/ipv6.c:1049
        ip6_protocol_deliver_rcu+0x2fe/0x1660 net/ipv6/ip6_input.c:397
        ip6_input_finish+0x84/0x170 net/ipv6/ip6_input.c:438
        NF_HOOK include/linux/netfilter.h:305 [inline]
        NF_HOOK include/linux/netfilter.h:299 [inline]
        ip6_input+0xe4/0x3f0 net/ipv6/ip6_input.c:447
        dst_input include/net/dst.h:442 [inline]
        ip6_sublist_rcv_finish+0x98/0x1e0 net/ipv6/ip6_input.c:84
        ip6_list_rcv_finish net/ipv6/ip6_input.c:118 [inline]
        ip6_sublist_rcv+0x80c/0xcf0 net/ipv6/ip6_input.c:282
```

```
          ipv6_list_rcv+0x373/0x4b0 net/ipv6/ip6_input.c:316
          __netif_receive_skb_list_ptype net/core/dev.c:5049 [inline]
          __netif_receive_skb_list_core+0x5fc/0x9d0 net/core/dev.c:5097
          __netif_receive_skb_list net/core/dev.c:5149 [inline]
          netif_receive_skb_list_internal+0x7eb/0xe60 net/core/dev.c:5244
          gro_normal_list.part.0+0x1e/0xb0 net/core/dev.c:5757
          gro_normal_list net/core/dev.c:5755 [inline]
          gro_normal_one net/core/dev.c:5769 [inline]
          napi_frags_finish net/core/dev.c:5782 [inline]
          napi_gro_frags+0xa6a/0xea0 net/core/dev.c:5855
          tun_get_user+0x2e98/0x3fa0 drivers/net/tun.c:1974
          tun_chr_write_iter+0xbd/0x156 drivers/net/tun.c:2020

      Fixes: d8269e2cbf90 ("net: ipv6: listify ipv6_rcv() and ip6_rcv_finish()")
      Fixes: 323ebb61e32b ("net: use listified RX for handling GRO_NORMAL skbs")
      Reported-by: syzbot+eb349eeee854e389c36d@syzkaller.appspotmail.com
      Reported-by: syzbot+4a0643a653ac375612d1@syzkaller.appspotmail.com
      Signed-off-by: Xin Long <lucien.xin@gmail.com>
      Acked-by: Edward Cree <ecree@solarflare.com>
      Signed-off-by: David S. Miller <davem@davemloft.net>
      Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

  commit 8690a9a2ad177e8e9b08d1a19f5032c28c282183
  Author: Cédric Le Goater <clg@kaod.org>
  Date:   Fri Oct 11 07:52:54 2019 +0200

      net/ibmvnic: Fix EOI when running in XIVE mode.

      [ Upstream commit 11d49ce9f7946dfed4dcf5dbde865c78058b50ab ]

      pSeries machines on POWER9 processors can run with the XICS (legacy)
      interrupt mode or with the XIVE exploitation interrupt mode. These
      interrupt contollers have different interfaces for interrupt
      management : XICS uses hcalls and XIVE loads and stores on a page.
      H_EOI being a XICS interface the enable_scrq_irq() routine can fail
      when the machine runs in XIVE mode.

      Fix that by calling the EOI handler of the interrupt chip.

      Fixes: f23e0643cd0b ("ibmvnic: Clear pending interrupt after device reset")
      Signed-off-by: Cédric Le Goater <clg@kaod.org>
      Signed-off-by: David S. Miller <davem@davemloft.net>
      Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

  commit 891cb53a600a398c95cc4826ece19a5f476f17dd
  Author: Thomas Bogendoerfer <tbogendoerfer@suse.de>
  Date:   Tue Oct 15 16:42:45 2019 +0200

      net: i82596: fix dma_alloc_attr for sni_82596

      [ Upstream commit 61c1d33daf7b5146f44d4363b3322f8cda6a6c43 ]

      Commit 7f683b920479 ("i825xx: switch to switch to dma_alloc_attrs")
      switched dma allocation over to dma_alloc_attr, but didn't convert
      the SNI part to request consistent DMA memory. This broke sni_82596
      since driver doesn't do dma_cache_sync for performance reasons.
      Fix this by using different DMA_ATTRs for lasi_82596 and sni_82596.

      Fixes: 7f683b920479 ("i825xx: switch to switch to dma_alloc_attrs")
      Signed-off-by: Thomas Bogendoerfer <tbogendoerfer@suse.de>
      Signed-off-by: David S. Miller <davem@davemloft.net>
      Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

  commit e230bab860f3afb961a20e84f4b89cd45161cd7d
  Author: Florian Fainelli <f.fainelli@gmail.com>
```

```
Date:   Fri Oct 11 12:53:49 2019 -0700

    net: bcmgenet: Set phydev->dev_flags only for internal PHYs

    [ Upstream commit 92696286f3bb37ba50e4bd8d1beb24afb759a799 ]

    phydev->dev_flags is entirely dependent on the PHY device driver which
    is going to be used, setting the internal GENET PHY revision in those
    bits only makes sense when drivers/net/phy/bcm7xxx.c is the PHY driver
    being used.

    Fixes: 487320c54143 ("net: bcmgenet: communicate integrated PHY revision to PHY driver")
    Signed-off-by: Florian Fainelli <f.fainelli@gmail.com>
    Acked-by: Doug Berger <opendmb@gmail.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 16bc04ac20e0e4dc9faff2d9e5f2b47a1d347163
Author: Florian Fainelli <f.fainelli@gmail.com>
Date:   Tue Oct 15 10:45:47 2019 -0700

    net: bcmgenet: Fix RGMII_MODE_EN value for GENET v1/2/3

    [ Upstream commit efb86fede98cdc70b674692ff617b1162f642c49 ]

    The RGMII_MODE_EN bit value was 0 for GENET versions 1 through 3, and
    became 6 for GENET v4 and above, account for that difference.

    Fixes: aa09677cba42 ("net: bcmgenet: add MDIO routines")
    Signed-off-by: Florian Fainelli <f.fainelli@gmail.com>
    Acked-by: Doug Berger <opendmb@gmail.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 58cc6ab97b2f2dea9ada64e4ee7ca9e6be333cef
Author: Eric Dumazet <edumazet@google.com>
Date:   Mon Oct 14 11:22:30 2019 -0700

    net: avoid potential infinite loop in tc_ctl_action()

    [ Upstream commit 39f13ea2f61b439ebe0060393e9c39925c9ee28c ]

    tc_ctl_action() has the ability to loop forever if tcf_action_add()
    returns -EAGAIN.

    This special case has been done in case a module needed to be loaded,
    but it turns out that tcf_add_notify() could also return -EAGAIN
    if the socket sk_rcvbuf limit is hit.

    We need to separate the two cases, and only loop for the module
    loading case.

    While we are at it, add a limit of 10 attempts since unbounded
    loops are always scary.

    syzbot repro was something like :

    socket(PF_NETLINK, SOCK_RAW|SOCK_NONBLOCK, NETLINK_ROUTE) = 3
    write(3, ..., 38) = 38
    setsockopt(3, SOL_SOCKET, SO_RCVBUF, [0], 4) = 0
    sendmsg(3, {msg_name(0)=NULL, msg_iov(1)=[{..., 388}], msg_controllen=0,
msg_flags=0x10}, ...)

    NMI backtrace for cpu 0
    CPU: 0 PID: 1054 Comm: khungtaskd Not tainted 5.4.0-rc1+ #0
```

```
    Hardware name: Google Google Compute Engine/Google Compute Engine, BIOS Google
01/01/2011
    Call Trace:
     __dump_stack lib/dump_stack.c:77 [inline]
     dump_stack+0x172/0x1f0 lib/dump_stack.c:113
     nmi_cpu_backtrace.cold+0x70/0xb2 lib/nmi_backtrace.c:101
     nmi_trigger_cpumask_backtrace+0x23b/0x28b lib/nmi_backtrace.c:62
     arch_trigger_cpumask_backtrace+0x14/0x20 arch/x86/kernel/apic/hw_nmi.c:38
     trigger_all_cpu_backtrace include/linux/nmi.h:146 [inline]
     check_hung_uninterruptible_tasks kernel/hung_task.c:205 [inline]
     watchdog+0x9d0/0xef0 kernel/hung_task.c:289
     kthread+0x361/0x430 kernel/kthread.c:255
     ret_from_fork+0x24/0x30 arch/x86/entry/entry_64.S:352
    Sending NMI from CPU 0 to CPUs 1:
    NMI backtrace for cpu 1
    CPU: 1 PID: 8859 Comm: syz-executor910 Not tainted 5.4.0-rc1+ #0
    Hardware name: Google Google Compute Engine/Google Compute Engine, BIOS Google
01/01/2011
    RIP: 0010:arch_local_save_flags arch/x86/include/asm/paravirt.h:751 [inline]
    RIP: 0010:lockdep_hardirqs_off+0x1df/0x2e0 kernel/locking/lockdep.c:3453
    Code: 5c 08 00 00 5b 41 5c 41 5d 5d c3 48 c7 c0 58 1d f3 88 48 ba 00 00 00 00 00 fc ff
df 48 c1 e8 03 80 3c 10 00 0f 85 d3 00 00 00 <48> 83 3d 21 9e 99 07 00 0f 84 b9 00 00 00 9c
58 0f 1f 44 00 00 f6
    RSP: 0018:ffff8880a6f3f1b8 EFLAGS: 00000046
    RAX: 1ffffffff11e63ab RBX: ffff88808c9c6080 RCX: 0000000000000000
    RDX: dfffc0000000000 RSI: 0000000000000000 RDI: ffff88808c9c6914
    RBP: ffff8880a6f3f1d0 R08: ffff88808c9c6080 R09: fffffbfff16be5d1
    R10: fffffbfff16be5d0 R11: 0000000000000003 R12: ffffffff8746591f
    R13: ffff88808c9c6080 R14: ffffffff8746591f R15: 0000000000000003
    FS:  00000000011e4880(0000) GS:ffff8880ae900000(0000) knlGS:0000000000000000
    CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    CR2: ffffffffff600400 CR3: 00000000a8920000 CR4: 00000000001406e0
    DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000
    DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400
    Call Trace:
     trace_hardirqs_off+0x62/0x240 kernel/trace/trace_preemptirq.c:45
     __raw_spin_lock_irqsave include/linux/spinlock_api_smp.h:108 [inline]
     _raw_spin_lock_irqsave+0x6f/0xcd kernel/locking/spinlock.c:159
     __wake_up_common_lock+0xc8/0x150 kernel/sched/wait.c:122
     __wake_up+0xe/0x10 kernel/sched/wait.c:142
     netlink_unlock_table net/netlink/af_netlink.c:466 [inline]
     netlink_unlock_table net/netlink/af_netlink.c:463 [inline]
     netlink_broadcast_filtered+0x705/0xb80 net/netlink/af_netlink.c:1514
     netlink_broadcast+0x3a/0x50 net/netlink/af_netlink.c:1534
     rtnetlink_send+0xdd/0x110 net/core/rtnetlink.c:714
     tcf_add_notify net/sched/act_api.c:1343 [inline]
     tcf_action_add+0x243/0x370 net/sched/act_api.c:1362
     tc_ctl_action+0x3b5/0x4bc net/sched/act_api.c:1410
     rtnetlink_rcv_msg+0x463/0xb00 net/core/rtnetlink.c:5386
     netlink_rcv_skb+0x177/0x450 net/netlink/af_netlink.c:2477
     rtnetlink_rcv+0x1d/0x30 net/core/rtnetlink.c:5404
     netlink_unicast_kernel net/netlink/af_netlink.c:1302 [inline]
     netlink_unicast+0x531/0x710 net/netlink/af_netlink.c:1328
     netlink_sendmsg+0x8a5/0xd60 net/netlink/af_netlink.c:1917
     sock_sendmsg_nosec net/socket.c:637 [inline]
     sock_sendmsg+0xd7/0x130 net/socket.c:657
     ___sys_sendmsg+0x803/0x920 net/socket.c:2311
     __sys_sendmsg+0x105/0x1d0 net/socket.c:2356
     __do_sys_sendmsg net/socket.c:2365 [inline]
     __se_sys_sendmsg net/socket.c:2363 [inline]
     __x64_sys_sendmsg+0x78/0xb0 net/socket.c:2363
     do_syscall_64+0xfa/0x760 arch/x86/entry/common.c:290
     entry_SYSCALL_64_after_hwframe+0x49/0xbe
    RIP: 0033:0x440939
```

```
    Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
    Signed-off-by: Eric Dumazet <edumazet@google.com>
    Reported-by: syzbot+cf0adbb9c28c8866c788@syzkaller.appspotmail.com
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ac138f244c6341ddae123eb5d7b55edcd4b45f9b
Author: Stefano Brivio <sbrivio@redhat.com>
Date:   Wed Oct 16 20:52:09 2019 +0200

    ipv4: Return -ENETUNREACH if we can't create route but saddr is valid

    [ Upstream commit 595e0651d0296bad2491a4a29a7a43eae6328b02 ]

    ...instead of -EINVAL. An issue was found with older kernel versions
    while unplugging a NFS client with pending RPCs, and the wrong error
    code here prevented it from recovering once link is back up with a
    configured address.

    Incidentally, this is not an issue anymore since commit 4f8943f80883
    ("SUNRPC: Replace direct task wakeups from softirq context"), included
    in 5.2-rc7, had the effect of decoupling the forwarding of this error
    by using SO_ERROR in xs_wake_error(), as pointed out by Benjamin
    Coddington.

    To the best of my knowledge, this isn't currently causing any further
    issue, but the error code doesn't look appropriate anyway, and we
    might hit this in other paths as well.

    In detail, as analysed by Gonzalo Siero, once the route is deleted
    because the interface is down, and can't be resolved and we return
    -EINVAL here, this ends up, courtesy of inet_sk_rebuild_header(),
    as the socket error seen by tcp_write_err(), called by
    tcp_retransmit_timer().

    In turn, tcp_write_err() indirectly calls xs_error_report(), which
    wakes up the RPC pending tasks with a status of -EINVAL. This is then
    seen by call_status() in the SUN RPC implementation, which aborts the
    RPC call calling rpc_exit(), instead of handling this as a
    potentially temporary condition, i.e. as a timeout.

    Return -EINVAL only if the input parameters passed to
    ip_route_output_key_hash_rcu() are actually invalid (this is the case
    if the specified source address is multicast, limited broadcast or
    all zeroes), but return -ENETUNREACH in all cases where, at the given
    moment, the given source address doesn't allow resolving the route.

    While at it, drop the initialisation of err to -ENETUNREACH, which
    was added to __ip_route_output_key() back then by commit
    0315e3827048 ("net: Fix behaviour of unreachable, blackhole and
    prohibit routes"), but actually had no effect, as it was, and is,
    overwritten by the fib_lookup() return code assignment, and anyway
    ignored in all other branches, including the if (fl4->saddr) one:
    I find this rather confusing, as it would look like -ENETUNREACH is
    the "default" error, while that statement has no effect.

    Also note that after commit fc75fc8339e7 ("ipv4: dont create routes
    on down devices"), we would get -ENETUNREACH if the device is down,
    but -EINVAL if the source address is specified and we can't resolve
    the route, and this appears to be rather inconsistent.

    Reported-by: Stefan Walter <walteste@inf.ethz.ch>
    Analysed-by: Benjamin Coddington <bcodding@redhat.com>
    Analysed-by: Gonzalo Siero <gsierohu@redhat.com>
    Signed-off-by: Stefano Brivio <sbrivio@redhat.com>
```

```
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a9352db24328d58ce44f17191eaef0654aae8962
Author: Wei Wang <weiwan@google.com>
Date:   Wed Oct 16 12:03:15 2019 -0700

    ipv4: fix race condition between route lookup and invalidation

    [ Upstream commit 5018c59607a511cdee743b629c76206d9c9e6d7b ]

    Jesse and Ido reported the following race condition:
    <CPU A, t0> - Received packet A is forwarded and cached dst entry is
    taken from the nexthop ('nhc->nhc_rth_input'). Calls skb_dst_set()

    <t1> - Given Jesse has busy routers ("ingesting full BGP routing tables
    from multiple ISPs"), route is added / deleted and rt_cache_flush() is
    called

    <CPU B, t2> - Received packet B tries to use the same cached dst entry
    from t0, but rt_cache_valid() is no longer true and it is replaced in
    rt_cache_route() by the newer one. This calls dst_dev_put() on the
    original dst entry which assigns the blackhole netdev to 'dst->dev'

    <CPU A, t3> - dst_input(skb) is called on packet A and it is dropped due
    to 'dst->dev' being the blackhole netdev

    There are 2 issues in the v4 routing code:
    1. A per-netns counter is used to do the validation of the route. That
    means whenever a route is changed in the netns, users of all routes in
    the netns needs to redo lookup. v6 has an implementation of only
    updating fn_sernum for routes that are affected.
    2. When rt_cache_valid() returns false, rt_cache_route() is called to
    throw away the current cache, and create a new one. This seems
    unnecessary because as long as this route does not change, the route
    cache does not need to be recreated.

    To fully solve the above 2 issues, it probably needs quite some code
    changes and requires careful testing, and does not suite for net branch.

    So this patch only tries to add the deleted cached rt into the uncached
    list, so user could still be able to use it to receive packets until
    it's done.

    Fixes: 95c47f9cf5e0 ("ipv4: call dst_dev_put() properly")
    Signed-off-by: Wei Wang <weiwan@google.com>
    Reported-by: Ido Schimmel <idosch@idosch.org>
    Reported-by: Jesse Hathaway <jesse@mbuki-mvuki.org>
    Tested-by: Jesse Hathaway <jesse@mbuki-mvuki.org>
    Acked-by: Martin KaFai Lau <kafai@fb.com>
    Cc: David Ahern <dsahern@gmail.com>
    Reviewed-by: Ido Schimmel <idosch@mellanox.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit efac0f186ea654e8389f5017c7f643ef48cb4b93
Author: Kevin Hao <haokexin@gmail.com>
Date:   Fri Oct 18 10:53:14 2019 +0800

    nvme-pci: Set the prp2 correctly when using more than 4k page

    commit a4f40484e7f1dff56bb9f286cc59ffa36e0259eb upstream.

    In the current code, the nvme is using a fixed 4k PRP entry size,
    but if the kernel use a page size which is more than 4k, we should
```

```
    consider the situation that the bv_offset may be larger than the
    dev->ctrl.page_size. Otherwise we may miss setting the prp2 and then
    cause the command can't be executed correctly.

    Fixes: dff824b2aadb ("nvme-pci: optimize mapping of small single segment requests")
    Cc: stable@vger.kernel.org
    Reviewed-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Kevin Hao <haokexin@gmail.com>
    Signed-off-by: Keith Busch <kbusch@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 714eccab25f22166db5ea000f20e87ebec401bf4
Author: Yi Li <yilikernel@gmail.com>
Date:   Fri Oct 18 20:20:08 2019 -0700

    ocfs2: fix panic due to ocfs2_wq is null

    commit b918c43021baaa3648de09e19a4a3dd555a45f40 upstream.

    mount.ocfs2 failed when reading ocfs2 filesystem superblock encounters
    an error.  ocfs2_initialize_super() returns before allocating ocfs2_wq.
    ocfs2_dismount_volume() triggers the following panic.

      Oct 15 16:09:27 cnwarekv-205120 kernel: On-disk corruption discovered.Please run
fsck.ocfs2 once the filesystem is unmounted.
      Oct 15 16:09:27 cnwarekv-205120 kernel: (mount.ocfs2,22804,44):
ocfs2_read_locked_inode:537 ERROR: status = -30
      Oct 15 16:09:27 cnwarekv-205120 kernel: (mount.ocfs2,22804,44):
ocfs2_init_global_system_inodes:458 ERROR: status = -30
      Oct 15 16:09:27 cnwarekv-205120 kernel: (mount.ocfs2,22804,44):
ocfs2_init_global_system_inodes:491 ERROR: status = -30
      Oct 15 16:09:27 cnwarekv-205120 kernel: (mount.ocfs2,22804,44):
ocfs2_initialize_super:2313 ERROR: status = -30
      Oct 15 16:09:27 cnwarekv-205120 kernel: (mount.ocfs2,22804,44): ocfs2_fill_super:1033
ERROR: status = -30
      ------------[ cut here ]------------
      Oops: 0002 [#1] SMP NOPTI
      CPU: 1 PID: 11753 Comm: mount.ocfs2 Tainted: G  E
            4.14.148-200.ckv.x86_64 #1
      Hardware name: Sugon H320-G30/35N16-US, BIOS 0SSDX017 12/21/2018
      task: ffff967af0520000 task.stack: ffffa5f05484000
      RIP: 0010:mutex_lock+0x19/0x20
      Call Trace:
        flush_workqueue+0x81/0x460
        ocfs2_shutdown_local_alloc+0x47/0x440 [ocfs2]
        ocfs2_dismount_volume+0x84/0x400 [ocfs2]
        ocfs2_fill_super+0xa4/0x1270 [ocfs2]
        ? ocfs2_initialize_super.isa.211+0xf20/0xf20 [ocfs2]
        mount_bdev+0x17f/0x1c0
        mount_fs+0x3a/0x160

    Link: http://lkml.kernel.org/r/1571139611-24107-1-git-send-email-yili@winhong.com
    Signed-off-by: Yi Li <yilikernel@gmail.com>
    Reviewed-by: Joseph Qi <joseph.qi@linux.alibaba.com>
    Cc: Mark Fasheh <mark@fasheh.com>
    Cc: Joel Becker <jlbec@evilplan.org>
    Cc: Junxiao Bi <junxiao.bi@oracle.com>
    Cc: Changwei Ge <gechangwei@live.cn>
    Cc: Gang He <ghe@suse.com>
    Cc: Jun Piao <piaojun@huawei.com>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit 4efbae7d629f32c73868b16b9269380ef72bb3d2
Author: Alex Deucher <alexander.deucher@amd.com>
Date:   Wed Oct 9 13:12:37 2019 -0500

    Revert "drm/radeon: Fix EEH during kexec"

    [ Upstream commit 8d13c187c42e110625d60094668a8f778c092879 ]

    This reverts commit 6f7fe9a93e6c09bf988c5059403f5f88e17e21e6.

    This breaks some boards.  Maybe just enable this on PPC for
    now?

    Bug: https://bugzilla.kernel.org/show_bug.cgi?id=205147
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9bcbee6f0a9efd6e3ffca9dce1799f720595d95d
Author: Song Liu <songliubraving@fb.com>
Date:   Mon Oct 14 16:58:35 2019 -0700

    md/raid0: fix warning message for parameter default_layout

    [ Upstream commit 3874d73e06c9b9dc15de0b7382fc223986d75571 ]

    The message should match the parameter, i.e. raid0.default_layout.

    Fixes: c84a1372df92 ("md/raid0: avoid RAID0 data corruption due to layout confusion.")
    Cc: NeilBrown <neilb@suse.de>
    Reported-by: Ivan Topolsky <doktor.yak@gmail.com>
    Signed-off-by: Song Liu <songliubraving@fb.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5cde895ec2f9e0d143d9232738496d74550549fa
Author: Dan Williams <dan.j.williams@intel.com>
Date:   Tue Oct 15 12:54:17 2019 -0700

    libata/ahci: Fix PCS quirk application

    [ Upstream commit 09d6ac8dc51a033ae0043c1fe40b4d02563c2496 ]

    Commit c312ef176399 "libata/ahci: Drop PCS quirk for Denverton and
    beyond" got the polarity wrong on the check for which board-ids should
    have the quirk applied. The board type board_ahci_pcs7 is defined at the
    end of the list such that "pcs7" boards can be special cased in the
    future if they need the quirk. All prior Intel board ids "<
    board_ahci_pcs7" should proceed with applying the quirk.

    Reported-by: Andreas Friedrich <afrie@gmx.net>
    Reported-by: Stephen Douthit <stephend@silicom-usa.com>
    Fixes: c312ef176399 ("libata/ahci: Drop PCS quirk for Denverton and beyond")
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Dan Williams <dan.j.williams@intel.com>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 00f0daa5e864cf7fb758cea7dd548ce05dd856cd
Author: Cong Wang <xiyou.wangcong@gmail.com>
Date:   Mon Oct 7 13:26:29 2019 -0700

    net_sched: fix backward compatibility for TCA_ACT_KIND

    [ Upstream commit 4b793feccae3b06764268377a4030eb774ed924e ]
```

```
    For TCA_ACT_KIND, we have to keep the backward compatibility too,
    and rely on nla_strlcpy() to check and terminate the string with
    a NUL.

    Note for TC actions, nla_strcmp() is already used to compare kind
    strings, so we don't need to fix other places.

    Fixes: 199ce850ce11 ("net_sched: add policy validation for action attributes")
    Reported-by: Marcelo Ricardo Leitner <marcelo.leitner@gmail.com>
    Cc: Jamal Hadi Salim <jhs@mojatatu.com>
    Cc: Jiri Pirko <jiri@resnulli.us>
    Signed-off-by: Cong Wang <xiyou.wangcong@gmail.com>
    Reviewed-by: Marcelo Ricardo Leitner <marcelo.leitner@gmail.com>
    Signed-off-by: Jakub Kicinski <jakub.kicinski@netronome.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 994aae2d6655cdb05c3a39b15bb18d43ce39365f
Author: Cong Wang <xiyou.wangcong@gmail.com>
Date:   Mon Oct 7 13:26:28 2019 -0700

    net_sched: fix backward compatibility for TCA_KIND

    [ Upstream commit 6f96c3c6904c26cea9ca2726d5d8a9b0b8205b3c ]

    Marcelo noticed a backward compatibility issue of TCA_KIND
    after we move from NLA_STRING to NLA_NUL_STRING, so it is probably
    too late to change it.

    Instead, to make everyone happy, we can just insert a NUL to
    terminate the string with nla_strlcpy() like we do for TC actions.

    Fixes: 62794fc4fbf5 ("net_sched: add max len check for TCA_KIND")
    Reported-by: Marcelo Ricardo Leitner <marcelo.leitner@gmail.com>
    Cc: Jamal Hadi Salim <jhs@mojatatu.com>
    Cc: Jiri Pirko <jiri@resnulli.us>
    Signed-off-by: Cong Wang <xiyou.wangcong@gmail.com>
    Reviewed-by: Marcelo Ricardo Leitner <marcelo.leitner@gmail.com>
    Signed-off-by: Jakub Kicinski <jakub.kicinski@netronome.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3d8dcdb7ff59f8a0095a4fa0dca83e8ce1cbb60e
Author: Linus Torvalds <torvalds@linux-foundation.org>
Date:   Fri Oct 18 18:41:16 2019 -0400

    filldir[64]: remove WARN_ON_ONCE() for bad directory entries

    [ Upstream commit b9959c7a347d6adbb558fba7e36e9fef3cba3b07 ]

    This was always meant to be a temporary thing, just for testing and to
    see if it actually ever triggered.

    The only thing that reported it was syzbot doing disk image fuzzing, and
    then that warning is expected.  So let's just remove it before -rc4,
    because the extra sanity testing should probably go to -stable, but we
    don't want the warning to do so.

    Reported-by: syzbot+3031f712c7ad5dd4d926@syzkaller.appspotmail.com
    Fixes: 8a23eb804ca4 ("Make filldir[64]() verify the directory entry filename is valid")
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 952cb934aa85d27e58cf0e300180b65bc23c864e
Author: Linus Torvalds <torvalds@linux-foundation.org>
Date:   Mon Oct 7 12:56:48 2019 -0700
```

```
    uaccess: implement a proper unsafe_copy_to_user() and switch filldir over to it

    [ Upstream commit c512c69187197fe08026cb5bbe7b9709f4f89b73 ]

    In commit 9f79b78ef744 ("Convert filldir[64]() from __put_user() to
    unsafe_put_user()") I made filldir() use unsafe_put_user(), which
    improves code generation on x86 enormously.

    But because we didn't have a "unsafe_copy_to_user()", the dirent name
    copy was also done by hand with unsafe_put_user() in a loop, and it
    turns out that a lot of other architectures didn't like that, because
    unlike x86, they have various alignment issues.

    Most non-x86 architectures trap and fix it up, and some (like xtensa)
    will just fail unaligned put_user() accesses unconditionally.  Which
    makes that "copy using put_user() in a loop" not work for them at all.

    I could make that code do explicit alignment etc, but the architectures
    that don't like unaligned accesses also don't really use the fancy
    "user_access_begin/end()" model, so they might just use the regular old
    __copy_to_user() interface.

    So this commit takes that looping implementation, turns it into the x86
    version of "unsafe_copy_to_user()", and makes other architectures
    implement the unsafe copy version as __copy_to_user() (the same way they
    do for the other unsafe_xyz() accessor functions).

    Note that it only does this for the copying _to_ user space, and we
    still don't have a unsafe version of copy_from_user().

    That's partly because we have no current users of it, but also partly
    because the copy_from_user() case is slightly different and cannot
    efficiently be implemented in terms of a unsafe_get_user() loop (because
    gcc can't do asm goto with outputs).

    It would be trivial to do this using "rep movsb", which would work
    really nicely on newer x86 cores, but really badly on some older ones.

    Al Viro is looking at cleaning up all our user copy routines to make
    this all a non-issue, but for now we have this simple-but-stupid version
    for x86 that works fine for the dirent name copy case because those
    names are short strings and we simply don't need anything fancier.

    Fixes: 9f79b78ef744 ("Convert filldir[64]() from __put_user() to unsafe_put_user()")
    Reported-by: Guenter Roeck <linux@roeck-us.net>
    Reported-and-tested-by: Tony Luck <tony.luck@intel.com>
    Cc: Al Viro <viro@zeniv.linux.org.uk>
    Cc: Max Filippov <jcmvbkbc@gmail.com>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 54e21d25b72eeea08435ce8623da82fc60381ed3
Author: Linus Torvalds <torvalds@linux-foundation.org>
Date:   Sat Oct 5 11:32:52 2019 -0700

    Make filldir[64]() verify the directory entry filename is valid

    [ Upstream commit 8a23eb804ca4f2be909e372cf5a9e7b30ae476cd ]

    This has been discussed several times, and now filesystem people are
    talking about doing it individually at the filesystem layer, so head
    that off at the pass and just do it in getdents{64}().

    This is partially based on a patch by Jann Horn, but checks for NUL
    bytes as well, and somewhat simplified.
```

```
    There's also commentary about how it might be better if invalid names
    due to filesystem corruption don't cause an immediate failure, but only
    an error at the end of the readdir(), so that people can still see the
    filenames that are ok.

    There's also been discussion about just how much POSIX strictly speaking
    requires this since it's about filesystem corruption.  It's really more
    "protect user space from bad behavior" as pointed out by Jann.  But
    since Eric Biederman looked up the POSIX wording, here it is for context:

     "From readdir:

       The readdir() function shall return a pointer to a structure
       representing the directory entry at the current position in the
       directory stream specified by the argument dirp, and position the
       directory stream at the next entry. It shall return a null pointer
       upon reaching the end of the directory stream. The structure dirent
       defined in the <dirent.h> header describes a directory entry.

      From definitions:

       3.129 Directory Entry (or Link)

       An object that associates a filename with a file. Several directory
       entries can associate names with the same file.

      ...

       3.169 Filename

       A name consisting of 1 to {NAME_MAX} bytes used to name a file. The
       characters composing the name may be selected from the set of all
       character values excluding the slash character and the null byte. The
       filenames dot and dot-dot have special meaning. A filename is
       sometimes referred to as a 'pathname component'."

    Note that I didn't bother adding the checks to any legacy interfaces
    that nobody uses.

    Also note that if this ends up being noticeable as a performance
    regression, we can fix that to do a much more optimized model that
    checks for both NUL and '/' at the same time one word at a time.

    We haven't really tended to optimize 'memchr()', and it only checks for
    one pattern at a time anyway, and we really _should_ check for NUL too
    (but see the comment about "soft errors" in the code about why it
    currently only checks for '/')

    See the CONFIG_DCACHE_WORD_ACCESS case of hash_name() for how the name
    lookup code looks for pathname terminating characters in parallel.

    Link: https://lore.kernel.org/lkml/20190118161440.220134-2-jannh@google.com/
    Cc: Alexander Viro <viro@zeniv.linux.org.uk>
    Cc: Jann Horn <jannh@google.com>
    Cc: Eric W. Biederman <ebiederm@xmission.com>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 239358775239d348d46c672d74688785a15e2437
Author: Linus Torvalds <torvalds@linux-foundation.org>
Date:   Sun Oct 6 13:53:27 2019 -0700

    elf: don't use MAP_FIXED_NOREPLACE for elf executable mappings
```

```
    [ Upstream commit b212921b13bda088a004328457c5c21458262fe2 ]

    In commit 4ed28639519c ("fs, elf: drop MAP_FIXED usage from elf_map") we
    changed elf to use MAP_FIXED_NOREPLACE instead of MAP_FIXED for the
    executable mappings.

    Then, people reported that it broke some binaries that had overlapping
    segments from the same file, and commit ad55eac74f20 ("elf: enforce
    MAP_FIXED on overlaying elf segments") re-instated MAP_FIXED for some
    overlaying elf segment cases.  But only some - despite the summary line
    of that commit, it only did it when it also does a temporary brk vma for
    one obvious overlapping case.

    Now Russell King reports another overlapping case with old 32-bit x86
    binaries, which doesn't trigger that limited case.  End result: we had
    better just drop MAP_FIXED_NOREPLACE entirely, and go back to MAP_FIXED.

    Yes, it's a sign of old binaries generated with old tool-chains, but we
    do pride ourselves on not breaking existing setups.

    This still leaves MAP_FIXED_NOREPLACE in place for the load_elf_interp()
    and the old load_elf_library() use-cases, because nobody has reported
    breakage for those. Yet.

    Note that in all the cases seen so far, the overlapping elf sections
    seem to be just re-mapping of the same executable with different section
    attributes.  We could possibly introduce a new MAP_FIXED_NOFILECHANGE
    flag or similar, which acts like NOREPLACE, but allows just remapping
    the same executable file using different protection flags.

    It's not clear that would make a huge difference to anything, but if
    people really hate that "elf remaps over previous maps" behavior, maybe
    at least a more limited form of remapping would alleviate some concerns.

    Alternatively, we should take a look at our elf_map() logic to see if we
    end up not mapping things properly the first time.

    In the meantime, this is the minimal "don't do that then" patch while
    people hopefully think about it more.

    Reported-by: Russell King <linux@armlinux.org.uk>
    Fixes: 4ed28639519c ("fs, elf: drop MAP_FIXED usage from elf_map")
    Fixes: ad55eac74f20 ("elf: enforce  MAP_FIXED on overlaying elf segments")
    Cc: Michal Hocko <mhocko@suse.com>
    Cc: Kees Cook <keescook@chromium.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 19487220604b33df12847c9bb9dac470e52e31c2
Author: Linus Torvalds <torvalds@linux-foundation.org>
Date:   Sat May 21 21:59:07 2016 -0700

    Convert filldir[64]() from __put_user() to unsafe_put_user()

    [ Upstream commit 9f79b78ef74436c7507bac6bfb7b8b989263bccb ]

    We really should avoid the "__{get,put}_user()" functions entirely,
    because they can easily be mis-used and the original intent of being
    used for simple direct user accesses no longer holds in a post-SMAP/PAN
    world.

    Manually optimizing away the user access range check makes no sense any
    more, when the range check is generally much cheaper than the "enable
    user accesses" code that the __{get,put}_user() functions still need.
```

```
    So instead of __put_user(), use the unsafe_put_user() interface with
    user_access_{begin,end}() that really does generate better code these
    days, and which is generally a nicer interface.  Under some loads, the
    multiple user writes that filldir() does are actually quite noticeable.

    This also makes the dirent name copy use unsafe_put_user() with a couple
    of macros.  We do not want to make function calls with SMAP/PAN
    disabled, and the code this generates is quite good when the
    architecture uses "asm goto" for unsafe_put_user() like x86 does.

    Note that this doesn't bother with the legacy cases.  Nobody should use
    them anyway, so performance doesn't really matter there.

    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 88d2c8c76d01d65010d8a4068caad3d733dd6635
Author: Jacob Keller <jacob.e.keller@intel.com>
Date:   Fri Sep 27 16:30:27 2019 -0700

    namespace: fix namespace.pl script to support relative paths

    [ Upstream commit 82fdd12b95727640c9a8233c09d602e4518e71f7 ]

    The namespace.pl script does not work properly if objtree is not set to
    an absolute path. The do_nm function is run from within the find
    function, which changes directories.

    Because of this, appending objtree, $File::Find::dir, and $source, will
    return a path which is not valid from the current directory.

    This used to work when objtree was set to an absolute path when using
    "make namespacecheck". It appears to have not worked when calling
    ./scripts/namespace.pl directly.

    This behavior was changed in 7e1c04779efd ("kbuild: Use relative path
    for $(objtree)", 2014-05-14)

    Rather than fixing the Makefile to set objtree to an absolute path, just
    fix namespace.pl to work when srctree and objtree are relative. Also fix
    the script to use an absolute path for these by default.

    Use the File::Spec module for this purpose. It's been part of perl
    5 since 5.005.

    The curdir() function is used to get the current directory when the
    objtree and srctree aren't set in the environment.

    rel2abs() is used to convert possibly relative objtree and srctree
    environment variables to absolute paths.

    Finally, the catfile() function is used instead of string appending
    paths together, since this is more robust when joining paths together.

    Signed-off-by: Jacob Keller <jacob.e.keller@intel.com>
    Acked-by: Randy Dunlap <rdunlap@infradead.org>
    Tested-by: Randy Dunlap <rdunlap@infradead.org>
    Signed-off-by: Masahiro Yamada <yamada.masahiro@socionext.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a9ab48bd6dad17c08de1afb4bbba9a99c6b0714b
Author: Russell King <rmk+kernel@armlinux.org.uk>
Date:   Fri Oct 4 17:05:58 2019 +0100

    net: phy: fix write to mii-ctrl1000 register
```

```
    [ Upstream commit 4cf6c57e61fee954f7b7685de31b80ec26843d27 ]

    When userspace writes to the MII_ADVERTISE register, we update phylib's
    advertising mask and trigger a renegotiation.  However, writing to the
    MII_CTRL1000 register, which contains the gigabit advertisement, does
    neither.  This can lead to phylib's copy of the advertisement becoming
    de-synced with the values in the PHY register set, which can result in
    incorrect negotiation resolution.

    Fixes: 5502b218e001 ("net: phy: use phy_resolve_aneg_linkmode in genphy_read_status")
    Reviewed-by: Andrew Lunn <andrew@lunn.ch>
    Signed-off-by: Russell King <rmk+kernel@armlinux.org.uk>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ec740ada78c345c69b3a8a9d5858857c97187cf6
Author: Andrea Merello <andrea.merello@gmail.com>
Date:   Fri Oct 4 15:53:32 2019 +0200

    net: phy: allow for reset line to be tied to a sleepy GPIO controller

    [ Upstream commit ea977d19d918324ad5b66953f051a6ed07d0a3c5 ]

    mdio_device_reset() makes use of the atomic-pretending API flavor for
    handling the PHY reset GPIO line.

    I found no hint that mdio_device_reset() is called from atomic context
    and indeed it uses usleep_range() since long time, so I would assume that
    it is OK to sleep there.

    This patch switch to gpiod_set_value_cansleep() in mdio_device_reset().
    This is relevant if e.g. the PHY reset line is tied to a I2C GPIO
    controller.

    This has been tested on a ZynqMP board running an upstream 4.19 kernel and
    then hand-ported on current kernel tree.

    Signed-off-by: Andrea Merello <andrea.merello@gmail.com>
    Reviewed-by: Andrew Lunn <andrew@lunn.ch>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 068aca62c58bedba6ccec571ab9d4573e8ad1755
Author: Kai-Heng Feng <kai.heng.feng@canonical.com>
Date:   Fri Oct 4 20:51:04 2019 +0800

    r8152: Set macpassthru in reset_resume callback

    [ Upstream commit a54cdeeb04fc719e4c7f19d6e28dba7ea86cee5b ]

    r8152 may fail to establish network connection after resume from system
    suspend.

    If the USB port connects to r8152 lost its power during system suspend,
    the MAC address was written before is lost. The reason is that The MAC
    address doesn't get written again in its reset_resume callback.

    So let's set MAC address again in reset_resume callback. Also remove
    unnecessary lock as no other locking attempt will happen during
    reset_resume.

    Signed-off-by: Kai-Heng Feng <kai.heng.feng@canonical.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit f72863cc0869b87665f0a28d97438c24e4169be6
Author: Qian Cai <cai@lca.pw>
Date:   Thu Oct 3 17:36:36 2019 -0400

    s390/mm: fix -Wunused-but-set-variable warnings

    [ Upstream commit 51ce02216d4ad4e8f6a58de81d6e803cf04c418e ]

    Convert two functions to static inline to get ride of W=1 GCC warnings
    like,

    mm/gup.c: In function 'gup_pte_range':
    mm/gup.c:1816:16: warning: variable 'ptem' set but not used
    [-Wunused-but-set-variable]
      pte_t *ptep, *ptem;
                   ^~~~

    mm/mmap.c: In function 'acct_stack_growth':
    mm/mmap.c:2322:16: warning: variable 'new_start' set but not used
    [-Wunused-but-set-variable]
      unsigned long new_start;
                    ^~~~~~~~~

    Signed-off-by: Qian Cai <cai@lca.pw>
    Link: https://lore.kernel.org/lkml/1570138596-11913-1-git-send-email-cai@lca.pw/
    Signed-off-by: Christian Borntraeger <borntraeger@de.ibm.com>
    Signed-off-by: Vasily Gorbik <gor@linux.ibm.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 02e3417757279b4b5100a3cd4fd895333fa92499
Author: Randy Dunlap <rdunlap@infradead.org>
Date:   Wed Oct 2 17:08:18 2019 -0700

    lib: textsearch: fix escapes in example code

    [ Upstream commit 2105b52e30debe7f19f3218598d8ae777dcc6776 ]

    This textsearch code example does not need the '\' escapes and they can
    be misleading to someone reading the example. Also, gcc and sparse warn
    that the "\%d" is an unknown escape sequence.

    Fixes: 5968a70d7af5 ("textsearch: fix kernel-doc warnings and add kernel-api section")
    Signed-off-by: Randy Dunlap <rdunlap@infradead.org>
    Cc: "David S. Miller" <davem@davemloft.net>
    Cc: netdev@vger.kernel.org
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 29c16b5ed525546ccd7a745b62e1f6c478a78430
Author: Shuah Khan <skhan@linuxfoundation.org>
Date:   Wed Oct 2 17:14:30 2019 -0600

    selftests: kvm: Fix libkvm build error

    [ Upstream commit 6e06983dde969c15eb4fdab77f0eda8b18ea28e6 ]

    Fix the following build error from "make TARGETS=kvm kselftest":

    libkvm.a(assert.o): relocation R_X86_64_32 against `.rodata.str1.1' can not be used when
making a PIE object; recompile with -fPIC

    This error is seen when build is done from the main Makefile using
    kselftest target. In this case KBUILD_CPPFLAGS and CC_OPTION_CFLAGS
    are defined.
```

```
    When build is invoked using:

    "make -C tools/testing/selftests/kvm" KBUILD_CPPFLAGS and CC_OPTION_CFLAGS
    aren't defined.

    There is no need to pass in KBUILD_CPPFLAGS and CC_OPTION_CFLAGS for the
    check to determine if --no-pie is necessary, which is the case when these
    two aren't defined when "make -C tools/testing/selftests/kvm" runs.

    Fix it by simplifying the no-pie-option logic. With this change, both
    build variations work.

    "make TARGETS=kvm kselftest"
    "make -C tools/testing/selftests/kvm"

    Signed-off-by: Shuah Khan <skhan@linuxfoundation.org>
    Signed-off-by: Paolo Bonzini <pbonzini@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 68a8a29c82dc9573c41b8cd39637a1545855bf43
Author: Thierry Reding <treding@nvidia.com>
Date:   Wed Oct 2 16:49:46 2019 +0200

    net: stmmac: Avoid deadlock on suspend/resume

    [ Upstream commit 134cc4cefad34d8d24670d8a911b59c3b89c6731 ]

    The stmmac driver will try to acquire its private mutex during suspend
    via phylink_resolve() -> stmmac_mac_link_down() -> stmmac_eee_init().
    However, the phylink configuration is updated with the private mutex
    held already, which causes a deadlock during suspend.

    Fix this by moving the phylink configuration updates out of the region
    of code protected by the private mutex.

    Fixes: 19e13cb27b99 ("net: stmmac: Hold rtnl lock in suspend/resume callbacks")
    Suggested-by: Bitan Biswas <bbiswas@nvidia.com>
    Signed-off-by: Thierry Reding <treding@nvidia.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dc29ff9e51f9f8ea19de5f99731f11357fb1b63f
Author: Yizhuo <yzhai003@ucr.edu>
Date:   Tue Oct 1 13:24:39 2019 -0700

    net: hisilicon: Fix usage of uninitialized variable in function mdio_sc_cfg_reg_write()

    [ Upstream commit 53de429f4e88f538f7a8ec2b18be8c0cd9b2c8e1 ]

    In function mdio_sc_cfg_reg_write(), variable "reg_value" could be
    uninitialized if regmap_read() fails. However, "reg_value" is used
    to decide the control flow later in the if statement, which is
    potentially unsafe.

    Signed-off-by: Yizhuo <yzhai003@ucr.edu>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit aae053ad734915fe207536c6400ed3dbc0937e93
Author: Christophe JAILLET <christophe.jaillet@wanadoo.fr>
Date:   Tue Sep 10 05:59:07 2019 +0200

    mips: Loongson: Fix the link time qualifier of 'serial_exit()'
```

```
    [ Upstream commit 25b69a889b638b0b7e51e2c4fe717a66bec0e566 ]

    'exit' functions should be marked as __exit, not __init.

    Fixes: 85cc028817ef ("mips: make loongsoon serial driver explicitly modular")
    Signed-off-by: Christophe JAILLET <christophe.jaillet@wanadoo.fr>
    Signed-off-by: Paul Burton <paul.burton@mips.com>
    Cc: chenhc@lemote.com
    Cc: ralf@linux-mips.org
    Cc: jhogan@kernel.org
    Cc: linux-mips@vger.kernel.org
    Cc: linux-kernel@vger.kernel.org
    Cc: kernel-janitors@vger.kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3e18db7ecf6605bf11d10be93608b30f6b5c1e68
Author: Navid Emamdoost <navid.emamdoost@gmail.com>
Date:   Mon Sep 16 22:20:44 2019 -0500

    drm/amd/display: memory leak

    [ Upstream commit 055e547478a11a6360c7ce05e2afc3e366968a12 ]

    In dcn*_clock_source_create when dcn20_clk_src_construct fails allocated
    clk_src needs release.

    Signed-off-by: Navid Emamdoost <navid.emamdoost@gmail.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 56ec75797dd2b1e394cbc76e0863b4f63c43945d
Author: Navid Emamdoost <navid.emamdoost@gmail.com>
Date:   Tue Oct 1 22:46:07 2019 -0500

    drm/amdgpu: fix multiple memory leaks in acp_hw_init

    [ Upstream commit 57be09c6e8747bf48704136d9e3f92bfb93f5725 ]

    In acp_hw_init there are some allocations that needs to be released in
    case of failure:

    1- adev->acp.acp_genpd should be released if any allocation attemp for
    adev->acp.acp_cell, adev->acp.acp_res or i2s_pdata fails.
    2- all of those allocations should be released if
    mfd_add_hotplug_devices or pm_genpd_add_device fail.
    3- Release is needed in case of time out values expire.

    Reviewed-by: Christian König <christian.koenig@amd.com>
    Signed-off-by: Navid Emamdoost <navid.emamdoost@gmail.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f18ed5bee7bb8a0e99e1c7e7d45e0e51d3497248
Author: Albert Ou <aou@eecs.berkeley.edu>
Date:   Fri Sep 27 16:14:18 2019 -0700

    riscv: Fix memblock reservation for device tree blob

    [ Upstream commit 922b0375fc93fb1a20c5617e37c389c26bbccb70 ]

    This fixes an error with how the FDT blob is reserved in memblock.
    An incorrect physical address calculation exposed the FDT header to
    unintended corruption, which typically manifested with of_fdt_raw_init()
    faulting during late boot after fdt_totalsize() returned a wrong value.
    Systems with smaller physical memory sizes more frequently trigger this
```

```
    issue, as the kernel is more likely to allocate from the DMA32 zone
    where bbl places the DTB after the kernel image.

    Commit 671f9a3e2e24 ("RISC-V: Setup initial page tables in two stages")
    changed the mapping of the DTB to reside in the fixmap area.
    Consequently, early_init_fdt_reserve_self() cannot be used anymore in
    setup_bootmem() since it relies on __pa() to derive a physical address,
    which does not work with dtb_early_va that is no longer a valid kernel
    logical address.

    The reserved[0x1] region shows the effect of the pointer underflow
    resulting from the __pa(initial_boot_params) offset subtraction:

    [    0.000000] MEMBLOCK configuration:
    [    0.000000]  memory size = 0x000000001fe00000 reserved size = 0x0000000000a2e514
    [    0.000000]  memory.cnt  = 0x1
    [    0.000000]  memory[0x0]     [0x0000000080200000-0x000000009fffffff],
0x000000001fe00000 bytes flags: 0x0
    [    0.000000]  reserved.cnt  = 0x2
    [    0.000000]  reserved[0x0]   [0x0000000080200000-0x0000000080c2dfeb],
0x0000000000a2dfec bytes flags: 0x0
    [    0.000000]  reserved[0x1]   [0xffffffff080100000-0xffffffff080100527],
0x0000000000000528 bytes flags: 0x0

    With the fix applied:

    [    0.000000] MEMBLOCK configuration:
    [    0.000000]  memory size = 0x000000001fe00000 reserved size = 0x0000000000a2e514
    [    0.000000]  memory.cnt  = 0x1
    [    0.000000]  memory[0x0]     [0x0000000080200000-0x000000009fffffff],
0x000000001fe00000 bytes flags: 0x0
    [    0.000000]  reserved.cnt  = 0x2
    [    0.000000]  reserved[0x0]   [0x0000000080200000-0x0000000080c2dfeb],
0x0000000000a2dfec bytes flags: 0x0
    [    0.000000]  reserved[0x1]   [0x0000000080e00000-0x0000000080e00527],
0x0000000000000528 bytes flags: 0x0

    Fixes: 671f9a3e2e24 ("RISC-V: Setup initial page tables in two stages")
    Signed-off-by: Albert Ou <aou@eecs.berkeley.edu>
    Tested-by: Bin Meng <bmeng.cn@gmail.com>
    Reviewed-by: Anup Patel <anup@brainfault.org>
    Signed-off-by: Paul Walmsley <paul.walmsley@sifive.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c5a30567faa18cb502b66eaf6276384364b52855
Author: Palmer Dabbelt <palmer@sifive.com>
Date:   Tue Sep 24 17:15:56 2019 -0700

    RISC-V: Clear load reservations while restoring hart contexts

    [ Upstream commit 18856604b3e7090ce42d533995173ee70c24b1c9 ]

    This is almost entirely a comment.  The bug is unlikely to manifest on
    existing hardware because there is a timeout on load reservations, but
    manifests on QEMU because there is no timeout.

    Signed-off-by: Palmer Dabbelt <palmer@sifive.com>
    Reviewed-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Paul Walmsley <paul.walmsley@sifive.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6bef308a627583305906065fe2e1bb7120cbab76
Author: Oleksij Rempel <o.rempel@pengutronix.de>
Date:   Tue Oct 1 08:41:47 2019 +0200
```

```
    net: ag71xx: fix mdio subnode support

    [ Upstream commit 569aad4fcd82cba64eb10ede235d330a00f0aa09 ]

    This patch is syncing driver with actual devicetree documentation:
    Documentation/devicetree/bindings/net/qca,ar71xx.txt
    |Optional subnodes:
    |- mdio : specifies the mdio bus, used as a container for phy nodes
    |  according to phy.txt in the same directory

    The driver was working with fixed phy without any noticeable issues. This bug
    was uncovered by introducing dsa ar9331-switch driver.
    Since no one reported this bug until now, I assume no body is using it
    and this patch should not brake existing system.

    Fixes: d51b6ce441d3 ("net: ethernet: add ag71xx driver")
    Signed-off-by: Oleksij Rempel <o.rempel@pengutronix.de>
    Reviewed-by: Andrew Lunn <andrew@lunn.ch>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ca043b3fbc96c54d8d73404678ceef3042dade42
Author: Jose Abreu <Jose.Abreu@synopsys.com>
Date:   Mon Sep 30 10:19:10 2019 +0200

    net: stmmac: Do not stop PHY if WoL is enabled

    [ Upstream commit 3e2bf04fb0447aa4b967b8000125178f55ae7800 ]

    If WoL is enabled we can't really stop the PHY, otherwise we will not
    receive the WoL packet. Fix this by telling phylink that only the MAC is
    down and only stop the PHY if WoL is not enabled.

    Fixes: 74371272f97f ("net: stmmac: Convert to phylink and remove phylib logic")
    Signed-off-by: Jose Abreu <Jose.Abreu@synopsys.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4ab44a47f6cf731d07e0a8f47d9c90c58c65ca27
Author: Jose Abreu <Jose.Abreu@synopsys.com>
Date:   Mon Sep 30 10:19:09 2019 +0200

    net: stmmac: Correctly take timestamp for PTPv2

    [ Upstream commit 14f347334bf232074616e29e29103dd0c7c54dec ]

    The case for PTPV2_EVENT requires event packets to be captured so add
    this setting to the list of enabled captures.

    Fixes: 891434b18ec0 ("stmmac: add IEEE PTPv1 and PTPv2 support.")
    Signed-off-by: Jose Abreu <Jose.Abreu@synopsys.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7875dbf9c2ac9bbb84b07add5b9946387b557618
Author: Jose Abreu <Jose.Abreu@synopsys.com>
Date:   Mon Sep 30 10:19:08 2019 +0200

    net: stmmac: dwmac4: Always update the MAC Hash Filter

    [ Upstream commit f79bfda3756c50a86c0ee65091935c42c5bbe0cb ]

    We need to always update the MAC Hash Filter so that previous entries
    are invalidated.
```

```
    Found out while running stmmac selftests.

    Fixes: b8ef7020d6e5 ("net: stmmac: add support for hash table size 128/256 in dwmac4")
    Signed-off-by: Jose Abreu <Jose.Abreu@synopsys.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 28bf511fd4a010b8f8f1bc453d56d965170ab1bb
Author: Jose Abreu <Jose.Abreu@synopsys.com>
Date:   Mon Sep 30 10:19:05 2019 +0200

    net: stmmac: xgmac: Not all Unicast addresses may be available

    [ Upstream commit 9a2ae7b3960eb2426a8560cbc3251e3453230d21 ]

    Some setups may not have all Unicast addresses filters available. Let's
    check this before trying to setup filters.

    Fixes: 0efedbf11f07 ("net: stmmac: xgmac: Fix XGMAC selftests")
    Signed-off-by: Jose Abreu <Jose.Abreu@synopsys.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit abc819c529ccc13976e544bf5735cde09908ac2b
Author: Wen Yang <wenyang@linux.alibaba.com>
Date:   Sun Sep 29 15:00:47 2019 +0800

    net: dsa: rtl8366rb: add missing of_node_put after calling of_get_child_by_name

    [ Upstream commit f32eb9d80470dab05df26b6efd02d653c72e6a11 ]

    of_node_put needs to be called when the device node which is got
    from of_get_child_by_name finished using.
    irq_domain_add_linear() also calls of_node_get() to increase refcount,
    so irq_domain will not be affected when it is released.

    Fixes: d8652956cf37 ("net: dsa: realtek-smi: Add Realtek SMI driver")
    Signed-off-by: Wen Yang <wenyang@linux.alibaba.com>
    Cc: Linus Walleij <linus.walleij@linaro.org>
    Cc: Andrew Lunn <andrew@lunn.ch>
    Cc: Vivien Didelot <vivien.didelot@gmail.com>
    Cc: Florian Fainelli <f.fainelli@gmail.com>
    Cc: "David S. Miller" <davem@davemloft.net>
    Cc: netdev@vger.kernel.org
    Cc: linux-kernel@vger.kernel.org
    Reviewed-by: Linus Walleij <linus.walleij@linaro.org>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5dd45c1e86a01b6e50969eab91f36a3ed619af61
Author: Wen Yang <wenyang@linux.alibaba.com>
Date:   Sun Sep 29 14:54:24 2019 +0800

    net: mscc: ocelot: add missing of_node_put after calling of_get_child_by_name

    [ Upstream commit d2c50b1cd94528aea8c8e9abb4cce81590f32cc4 ]

    of_node_put needs to be called when the device node which is got
    from of_get_child_by_name finished using.
    In both cases of success and failure, we need to release 'ports',
    so clean up the code using goto.

    fixes: a556c76adc05 ("net: mscc: Add initial Ocelot switch support")
    Signed-off-by: Wen Yang <wenyang@linux.alibaba.com>
    Cc: Alexandre Belloni <alexandre.belloni@bootlin.com>
```

```
    Cc: Microchip Linux Driver Support <UNGLinuxDriver@microchip.com>
    Cc: "David S. Miller" <davem@davemloft.net>
    Cc: netdev@vger.kernel.org
    Cc: linux-kernel@vger.kernel.org
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit da370945277e4ed745a13b767c88e270190a06c3
Author: Pablo Neira Ayuso <pablo@netfilter.org>
Date:   Mon Sep 30 11:05:49 2019 +0200

    netfilter: nft_connlimit: disable bh on garbage collection

    [ Upstream commit 34a4c95abd25ab41fb390b985a08a651b1fa0b0f ]

    BH must be disabled when invoking nf_conncount_gc_list() to perform
    garbage collection, otherwise deadlock might happen.

      nf_conncount_add+0x1f/0x50 [nf_conncount]
      nft_connlimit_eval+0x4c/0xe0 [nft_connlimit]
      nft_dynset_eval+0xb5/0x100 [nf_tables]
      nft_do_chain+0xea/0x420 [nf_tables]
      ? sch_direct_xmit+0x111/0x360
      ? noqueue_init+0x10/0x10
      ? __qdisc_run+0x84/0x510
      ? tcp_packet+0x655/0x1610 [nf_conntrack]
      ? ip_finish_output2+0x1a7/0x430
      ? tcp_error+0x130/0x150 [nf_conntrack]
      ? nf_conntrack_in+0x1fc/0x4c0 [nf_conntrack]
      nft_do_chain_ipv4+0x66/0x80 [nf_tables]
      nf_hook_slow+0x44/0xc0
      ip_rcv+0xb5/0xd0
      ? ip_rcv_finish_core.isra.19+0x360/0x360
      __netif_receive_skb_one_core+0x52/0x70
      netif_receive_skb_internal+0x34/0xe0
      napi_gro_receive+0xba/0xe0
      e1000_clean_rx_irq+0x1e9/0x420 [e1000e]
      e1000e_poll+0xbe/0x290 [e1000e]
      net_rx_action+0x149/0x3b0
      __do_softirq+0xde/0x2d8
      irq_exit+0xba/0xc0
      do_IRQ+0x85/0xd0
      common_interrupt+0xf/0xf
      </IRQ>
      RIP: 0010:nf_conncount_gc_list+0x3b/0x130 [nf_conncount]

    Fixes: 2f971a8f4255 ("netfilter: nf_conncount: move all list iterations under spinlock")
    Reported-by: Laura Garcia Liebana <nevola@gmail.com>
    Signed-off-by: Pablo Neira Ayuso <pablo@netfilter.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 52e383a29d05c4d07bcb0fe86b63af63488679bc
Author: Miaoqing Pan <miaoqing@codeaurora.org>
Date:   Fri Sep 27 10:03:16 2019 +0800

    mac80211: fix txq null pointer dereference

    [ Upstream commit 8ed31a264065ae92058ce54aa3cc8da8d81dc6d7 ]

    If the interface type is P2P_DEVICE or NAN, read the file of
    '/sys/kernel/debug/ieee80211/phyx/netdev:wlanx/aqm' will get a
    NULL pointer dereference. As for those interface type, the
    pointer sdata->vif.txq is NULL.

    Unable to handle kernel NULL pointer dereference at virtual address 00000011
```

```
    CPU: 1 PID: 30936 Comm: cat Not tainted 4.14.104 #1
    task: ffffffc0337e4880 task.stack: ffffff800cd20000
    PC is at ieee80211_if_fmt_aqm+0x34/0xa0 [mac80211]
    LR is at ieee80211_if_fmt_aqm+0x34/0xa0 [mac80211]
    [...]
    Process cat (pid: 30936, stack limit = 0xffffff800cd20000)
    [...]
    [<ffffff8000b7cd00>] ieee80211_if_fmt_aqm+0x34/0xa0 [mac80211]
    [<ffffff8000b7c414>] ieee80211_if_read+0x60/0xbc [mac80211]
    [<ffffff8000b7ccc4>] ieee80211_if_read_aqm+0x28/0x30 [mac80211]
    [<ffffff80082eff94>] full_proxy_read+0x2c/0x48
    [<ffffff80081eef00>] __vfs_read+0x2c/0xd4
    [<ffffff80081ef084>] vfs_read+0x8c/0x108
    [<ffffff80081ef494>] SyS_read+0x40/0x7c

    Signed-off-by: Miaoqing Pan <miaoqing@codeaurora.org>
    Acked-by: Toke Høiland-Jørgensen <toke@redhat.com>
    Link: https://lore.kernel.org/r/1569549796-8223-1-git-send-email-miaoqing@codeaurora.org
    [trim useless data from commit message]
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e1c1cba89d18d8cfa6ec16bfa67c18bf4a4f0906
Author: Miaoqing Pan <miaoqing@codeaurora.org>
Date:   Thu Sep 26 16:16:50 2019 +0800

    nl80211: fix null pointer dereference

    [ Upstream commit b501426cf86e70649c983c52f4c823b3c40d72a3 ]

    If the interface is not in MESH mode, the command 'iw wlanx mpath del'
    will cause kernel panic.

    The root cause is null pointer access in mpp_flush_by_proxy(), as the
    pointer 'sdata->u.mesh.mpp_paths' is NULL for non MESH interface.

    Unable to handle kernel NULL pointer dereference at virtual address 00000068
    [...]
    PC is at _raw_spin_lock_bh+0x20/0x5c
    LR is at mesh_path_del+0x1c/0x17c [mac80211]
    [...]
    Process iw (pid: 4537, stack limit = 0xd83e0238)
    [...]
    [<c021211c>] (_raw_spin_lock_bh) from [<bf8c7648>] (mesh_path_del+0x1c/0x17c [mac80211])
    [<bf8c7648>] (mesh_path_del [mac80211]) from [<bf6cdb7c>] (extack_doit+0x20/0x68
[compat])
    [<bf6cdb7c>] (extack_doit [compat]) from [<c05c309c>] (genl_rcv_msg+0x274/0x30c)
    [<c05c309c>] (genl_rcv_msg) from [<c05c25d8>] (netlink_rcv_skb+0x58/0xac)
    [<c05c25d8>] (netlink_rcv_skb) from [<c05c2e14>] (genl_rcv+0x20/0x34)
    [<c05c2e14>] (genl_rcv) from [<c05c1f90>] (netlink_unicast+0x11c/0x204)
    [<c05c1f90>] (netlink_unicast) from [<c05c2420>] (netlink_sendmsg+0x30c/0x370)
    [<c05c2420>] (netlink_sendmsg) from [<c05886d0>] (sock_sendmsg+0x70/0x84)
    [<c05886d0>] (sock_sendmsg) from [<c0589f4c>] (___sys_sendmsg.part.3+0x188/0x228)
    [<c0589f4c>] (___sys_sendmsg.part.3) from [<c058add4>] (__sys_sendmsg+0x4c/0x70)
    [<c058add4>] (__sys_sendmsg) from [<c0208c80>] (ret_fast_syscall+0x0/0x44)
    Code: e2822c02 e2822001 e5832004 f590f000 (e1902f9f)
    ---[ end trace bbd717600f8f884d ]---

    Signed-off-by: Miaoqing Pan <miaoqing@codeaurora.org>
    Link: https://lore.kernel.org/r/1569485810-761-1-git-send-email-miaoqing@codeaurora.org
    [trim useless data from commit message]
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit bb6c8b40d3dc3892af91719cbc241f0eb44ab76f
```

```
Author: Martijn Coenen <maco@android.com>
Date:   Wed Sep 4 21:49:01 2019 +0200

    loop: change queue block size to match when using DIO

    [ Upstream commit 85560117d00f5d528e928918b8f61cadcefff98b ]

    The loop driver assumes that if the passed in fd is opened with
    O_DIRECT, the caller wants to use direct I/O on the loop device.
    However, if the underlying block device has a different block size than
    the loop block queue, direct I/O can't be enabled. Instead of requiring
    userspace to manually change the blocksize and re-enable direct I/O,
    just change the queue block sizes to match, as well as the io_min size.

    Reviewed-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Martijn Coenen <maco@android.com>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e496ac39db15db86e2bfa07c009189a650447528
Author: Ross Lagerwall <ross.lagerwall@citrix.com>
Date:   Fri Sep 27 16:49:20 2019 +0100

    xen/efi: Set nonblocking callbacks

    [ Upstream commit df359f0d09dc029829b66322707a2f558cb720f7 ]

    Other parts of the kernel expect these nonblocking EFI callbacks to
    exist and crash when running under Xen. Since the implementations of
    xen_efi_set_variable() and xen_efi_query_variable_info() do not take any
    locks, use them for the nonblocking callbacks too.

    Signed-off-by: Ross Lagerwall <ross.lagerwall@citrix.com>
    Reviewed-by: Juergen Gross <jgross@suse.com>
    Signed-off-by: Juergen Gross <jgross@suse.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3ff3df5de493e3d96495fbaf09e2becb35cf1cee
Author: Oleksij Rempel <o.rempel@pengutronix.de>
Date:   Mon Sep 30 11:39:52 2019 +0200

    MIPS: dts: ar9331: fix interrupt-controller size

    [ Upstream commit 0889d07f3e4b171c453b2aaf2b257f9074cdf624 ]

    It is two registers each of 4 byte.

    Signed-off-by: Oleksij Rempel <o.rempel@pengutronix.de>
    Signed-off-by: Paul Burton <paul.burton@mips.com>
    Cc: Rob Herring <robh+dt@kernel.org>
    Cc: Mark Rutland <mark.rutland@arm.com>
    Cc: Pengutronix Kernel Team <kernel@pengutronix.de>
    Cc: Ralf Baechle <ralf@linux-mips.org>
    Cc: James Hogan <jhogan@kernel.org>
    Cc: devicetree@vger.kernel.org
    Cc: linux-mips@vger.kernel.org
    Cc: linux-kernel@vger.kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e1aa4ebac842765c766bdc933fbac1b2309a56e2
Author: Michal Vokáč <michal.vokac@ysoft.com>
Date:   Thu Sep 26 10:59:17 2019 +0200

    net: dsa: qca8k: Use up to 7 ports for all operations
```

```
    [ Upstream commit 7ae6d93c8f052b7a77ba56ed0f654e22a2876739 ]

    The QCA8K family supports up to 7 ports. So use the existing
    QCA8K_NUM_PORTS define to allocate the switch structure and limit all
    operations with the switch ports.

    This was not an issue until commit 0394a63acfe2 ("net: dsa: enable and
    disable all ports") disabled all unused ports. Since the unused ports 7-11
    are outside of the correct register range on this switch some registers
    were rewritten with invalid content.

    Fixes: 6b93fb46480a ("net-next: dsa: add new driver for qca8xxx family")
    Fixes: a0c02161ecfc ("net: dsa: variable number of ports")
    Fixes: 0394a63acfe2 ("net: dsa: enable and disable all ports")
    Signed-off-by: Michal Vokáč <michal.vokac@ysoft.com>
    Reviewed-by: Andrew Lunn <andrew@lunn.ch>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b6e60d63be2be338787c8997cbc0740bcac8d2d6
Author: Peter Ujfalusi <peter.ujfalusi@ti.com>
Date:   Mon Sep 30 11:54:50 2019 +0300

    ARM: dts: am4372: Set memory bandwidth limit for DISPC

    [ Upstream commit f90ec6cdf674248dcad85bf9af6e064bf472b841 ]

    Set memory bandwidth limit to filter out resolutions above 720p@60Hz to
    avoid underflow errors due to the bandwidth needs of higher resolutions.

    am43xx can not provide enough bandwidth to DISPC to correctly handle
    'high' resolutions.

    Signed-off-by: Peter Ujfalusi <peter.ujfalusi@ti.com>
    Signed-off-by: Tomi Valkeinen <tomi.valkeinen@ti.com>
    Signed-off-by: Tony Lindgren <tony@atomide.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9295316e05812ca44838b1cf0a4ab0a91d1c7fd7
Author: Navid Emamdoost <navid.emamdoost@gmail.com>
Date:   Tue Sep 17 17:47:12 2019 -0500

    ieee802154: ca8210: prevent memory leak

    [ Upstream commit 6402939ec86eaf226c8b8ae00ed983936b164908 ]

    In ca8210_probe the allocated pdata needs to be assigned to
    spi_device->dev.platform_data before calling ca8210_get_platform_data.
    Othrwise when ca8210_get_platform_data fails pdata cannot be released.

    Signed-off-by: Navid Emamdoost <navid.emamdoost@gmail.com>
    Link: https://lore.kernel.org/r/20190917224713.26371-1-navid.emamdoost@gmail.com
    Signed-off-by: Stefan Schmidt <stefan@datenfreihafen.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 322b0504abf5a7ecb7d9460d92ccf389af859b22
Author: Ming Lei <ming.lei@redhat.com>
Date:   Fri Sep 27 15:24:30 2019 +0800

    blk-mq: honor IO scheduler for multiqueue devices

    [ Upstream commit a12de1d42d74ef3c80e9fb9a2da94daaef747869 ]

    If a device is using multiple queues, the IO scheduler may be bypassed.
    This may hurt performance for some slow MQ devices, and it also breaks
```

```
    zoned devices which depend on mq-deadline for respecting the write order
    in one zone.

    Don't bypass io scheduler if we have one setup.

    This patch can double sequential write performance basically on MQ
    scsi_debug when mq-deadline is applied.

    Cc: Bart Van Assche <bvanassche@acm.org>
    Cc: Hannes Reinecke <hare@suse.com>
    Cc: Dave Chinner <dchinner@redhat.com>
    Reviewed-by: Javier González <javier@javigon.com>
    Reviewed-by: Damien Le Moal <damien.lemoal@wdc.com>
    Signed-off-by: Ming Lei <ming.lei@redhat.com>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8bdf437e0b6e082faa317ee9b66d5749845e7af6
Author: Sagi Grimberg <sagi@grimberg.me>
Date:   Tue Sep 24 11:27:05 2019 -0700

    nvme-rdma: fix possible use-after-free in connect timeout

    [ Upstream commit 67b483dd03c4cd9e90e4c3943132dce514ea4e88 ]

    If the connect times out, we may have already destroyed the
    queue in the timeout handler, so test if the queue is still
    allocated in the connect error handler.

    Reported-by: Yi Zhang <yi.zhang@redhat.com>
    Signed-off-by: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b1fad53d75b17c19e04d4ecaf1e9b01da1d7c77c
Author: Navid Emamdoost <navid.emamdoost@gmail.com>
Date:   Tue Sep 24 23:30:30 2019 -0500

    drm/komeda: prevent memory leak in komeda_wb_connector_add

    [ Upstream commit a0ecd6fdbf5d648123a7315c695fb6850d702835 ]

    In komeda_wb_connector_add if drm_writeback_connector_init fails the
    allocated memory for kwb_conn should be released.

    Signed-off-by: Navid Emamdoost <navid.emamdoost@gmail.com>
    Reviewed-by: James Qian Wang (Arm Technology China) <james.qian.wang@arm.com>
    Signed-off-by: james qian wang (Arm Technology China) <james.qian.wang@arm.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20190925043031.32308-1-
navid.emamdoost@gmail.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 76d609da9ed1cc0dc780e2b539d7b827ce28f182
Author: Marta Rybczynska <mrybczyn@kalray.eu>
Date:   Tue Sep 24 15:14:52 2019 +0200

    nvme: allow 64-bit results in passthru commands

    [ Upstream commit 65e68edce0db433aa0c2b26d7dc14fbbbeb89fbb ]

    It is not possible to get 64-bit results from the passthru commands,
    what prevents from getting for the Capabilities (CAP) property value.

    As a result, it is not possible to implement IOL's NVMe Conformance
    test 4.3 Case 1 for Fabrics targets [1] (page 123).
```

```
    This issue has been already discussed [2], but without a solution.

    This patch solves the problem by adding new ioctls with a new
    passthru structure, including 64-bit results. The older ioctls stay
    unchanged.

    [1] https://www.iol.unh.edu/sites/default/files/testsuites/nvme/UNH-
IOL_NVMe_Conformance_Test_Suite_v11.0.pdf
    [2] http://lists.infradead.org/pipermail/linux-nvme/2018-June/018791.html

    Signed-off-by: Marta Rybczynska <marta.rybczynska@kalray.eu>
    Reviewed-by: Keith Busch <kbusch@kernel.org>
    Reviewed-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7af162625b8e9b9de00f3afca0815c4a93bbd4ed
Author: Jian-Hong Pan <jian-hong@endlessm.com>
Date:   Tue Sep 24 17:42:41 2019 +0800

    nvme: Add quirk for Kingston NVME SSD running FW E8FK11.T

    [ Upstream commit 19ea025e1d28c629b369c3532a85b3df478cc5c6 ]

    Kingston NVME SSD with firmware version E8FK11.T has no interrupt after
    resume with actions related to suspend to idle. This patch applied
    NVME_QUIRK_SIMPLE_SUSPEND quirk to fix this issue.

    Fixes: d916b1be94b6 ("nvme-pci: use host managed power state for suspend")
    Buglink: https://bugzilla.kernel.org/show_bug.cgi?id=204887
    Signed-off-by: Jian-Hong Pan <jian-hong@endlessm.com>
    Signed-off-by: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f5c23bd2a5b057f07e598959bccc346f112e3914
Author: Gabriel Craciunescu <nix.or.die@gmail.com>
Date:   Mon Sep 23 20:22:56 2019 +0200

    Added QUIRKs for ADATA XPG SX8200 Pro 512GB

    [ Upstream commit f03e42c6af60f778a6d1ccfb857db9b2ec835279 ]

    Booting with default_ps_max_latency_us >6000 makes the device fail.
    Also SUBNQN is NULL and gives a warning on each boot/resume.
     $ nvme id-ctrl /dev/nvme0 | grep ^subnqn
       subnqn    : (null)

    I use this device with an Acer Nitro 5 (AN515-43-R8BF) Laptop.
    To be sure is not a Laptop issue only, I tested the device on
    my server board  with the same results.
    ( with 2x,4x link on the board and 4x link on a PCI-E card ).

    Signed-off-by: Gabriel Craciunescu <nix.or.die@gmail.com>
    Reviewed-by: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit fc41cea6a453f06898bb206d948f501c79606f31
Author: Max Gurtovoy <maxg@mellanox.com>
Date:   Sat Sep 21 23:58:19 2019 +0300

    nvme-rdma: Fix max_hw_sectors calculation

    [ Upstream commit ff13c1b87c97275b82b2af99044b4abf6861b28f ]
```

```
    By default, the NVMe/RDMA driver should support max io_size of 1MiB (or
    upto the maximum supported size by the HCA). Currently, one will see that
    /sys/class/block/<bdev>/queue/max_hw_sectors_kb is 1020 instead of 1024.

    A non power of 2 value can cause performance degradation due to
    unnecessary splitting of IO requests and unoptimized allocation units.

    The number of pages per MR has been fixed here, so there is no longer any
    need to reduce max_sectors by 1.

    Reviewed-by: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Max Gurtovoy <maxg@mellanox.com>
    Signed-off-by: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e92ee7f0b23a4697f02886fceb9748fa9c72bfd7
Author: Dan Carpenter <dan.carpenter@oracle.com>
Date:   Mon Sep 23 17:18:36 2019 +0300

    nvme: fix an error code in nvme_init_subsystem()

    [ Upstream commit bc4f6e06a90ea016855fc67212b4d500145f0b8a ]

    "ret" should be a negative error code here, but it's either success or
    possibly uninitialized.

    Fixes: 32fd90c40768 ("nvme: change locking for the per-subsystem controller list")
    Signed-off-by: Dan Carpenter <dan.carpenter@oracle.com>
    Reviewed-by: Keith Busch <kbusch@kernel.org>
    Signed-off-by: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 984890ff39fa1f92cc3c76c33b9031ac38614afa
Author: Mario Limonciello <mario.limonciello@dell.com>
Date:   Wed Sep 18 13:15:55 2019 -0500

    nvme-pci: Save PCI state before putting drive into deepest state

    [ Upstream commit 7cbb5c6f9aa7cfda7175d82a9cf77a92965b0c5e ]

    The action of saving the PCI state will cause numerous PCI configuration
    space reads which depending upon the vendor implementation may cause
    the drive to exit the deepest NVMe state.

    In these cases ASPM will typically resolve the PCIe link state and APST
    may resolve the NVMe power state.  However it has also been observed
    that this register access after quiesced will cause PC10 failure
    on some device combinations.

    To resolve this, move the PCI state saving to before SetFeatures has been
    called.  This has been proven to resolve the issue across a 5000 sample
    test on previously failing disk/system combinations.

    Signed-off-by: Mario Limonciello <mario.limonciello@dell.com>
    Reviewed-by: Keith Busch <kbusch@kernel.org>
    Signed-off-by: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b9e1d38d0e979b48b63c06ebe82d17b12e269531
Author: Wunderlich, Mark <mark.wunderlich@intel.com>
Date:   Wed Sep 18 23:36:37 2019 +0000

    nvme-tcp: fix wrong stop condition in io_work

    [ Upstream commit ddef29578a81a1d4d8f2b26a7adbfe21407ee3ea ]
```

```
    Allow the do/while statement to continue if current time
    is not after the proposed time 'deadline'. Intent is to
    allow loop to proceed for a specific time period. Currently
    the loop, as coded, will exit after first pass.

    Signed-off-by: Mark Wunderlich <mark.wunderlich@intel.com>
    Reviewed-by: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 282c88c2c8e87b0ab15ca3decebe5bdc013cd891
Author: Tony Lindgren <tony@atomide.com>
Date:   Tue Sep 24 16:19:00 2019 -0700

    ARM: OMAP2+: Fix warnings with broken omap2_set_init_voltage()

    [ Upstream commit cf395f7ddb9ebc6b2d28d83b53d18aa4e7c19701 ]

    This code is currently unable to find the dts opp tables as ti-cpufreq
    needs to set them up first based on speed binning.

    We stopped initializing the opp tables with platform code years ago for
    device tree based booting with commit 92d51856d740 ("ARM: OMAP3+: do not
    register non-dt OPP tables for device tree boot"), and all of mach-omap2
    is now booting using device tree.

    We currently get the following errors on init:

    omap2_set_init_voltage: unable to find boot up OPP for vdd_mpu
    omap2_set_init_voltage: unable to set vdd_mpu
    omap2_set_init_voltage: unable to find boot up OPP for vdd_core
    omap2_set_init_voltage: unable to set vdd_core
    omap2_set_init_voltage: unable to find boot up OPP for vdd_iva
    omap2_set_init_voltage: unable to set vdd_iva

    Let's just drop the unused code. Nowadays ti-cpufreq should be used to
    to initialize things properly.

    Cc: Adam Ford <aford173@gmail.com>
    Cc: André Roth <neolynx@gmail.com>
    Cc: "H. Nikolaus Schaller" <hns@goldelico.com>
    Cc: Nishanth Menon <nm@ti.com>
    Cc: Tero Kristo <t-kristo@ti.com>
    Tested-by: Adam Ford <aford173@gmail.com> #logicpd-torpedo-37xx-devkit
    Signed-off-by: Tony Lindgren <tony@atomide.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a37119168d61a8a439552e50f9057bef43af428f
Author: Tony Lindgren <tony@atomide.com>
Date:   Tue Sep 24 09:25:51 2019 -0700

    ARM: OMAP2+: Add missing LCDC midlemode for am335x

    [ Upstream commit 17529d43b21c72466e9109d602c6f5c360a1a9e8 ]

    TRM "Table 13-34. SYSCONFIG Register Field Descriptions" lists both
    standbymode and idlemode that should be just the sidle and midle
    registers where midle is currently unconfigured for lcdc_sysc. As
    the dts data has been generated based on lcdc_sysc, we now have an
    empty "ti,sysc-midle" property.

    And so we currently get a warning for lcdc because of a difference
    with dts provided configuration compared to the legacy platform
    data. This is because lcdc has SYSC_HAS_MIDLEMODE configured in
```

```
    the platform data without configuring the modes.

    Let's fix the issue by adding the missing midlemode to lcdc_sysc,
    and configuring the "ti,sysc-midle" property based on the TRM
    values.

    Fixes: f711c575cfec ("ARM: dts: am335x: Add l4 interconnect hierarchy and ti-sysc data")
    Cc: Jyri Sarha <jsarha@ti.com>
    Cc: Keerthy <j-keerthy@ti.com>
    Cc: Robert Nelson <robertcnelson@gmail.com>
    Cc: Suman Anna <s-anna@ti.com>
    Signed-off-by: Tony Lindgren <tony@atomide.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dae37e0c76387cebe21f5b6866fd776b2f70108e
Author: Tony Lindgren <tony@atomide.com>
Date:   Tue Sep 24 09:25:52 2019 -0700

    ARM: OMAP2+: Fix missing reset done flag for am3 and am43

    [ Upstream commit 8ad8041b98c665b6147e607b749586d6e20ba73a ]

    For ti,sysc-omap4 compatible devices with no sysstatus register, we do have
    reset done status available in the SOFTRESET bit that clears when the reset
    is done. This is documented for example in am437x TRM for DMTIMER_TIOCP_CFG
    register. The am335x TRM just says that SOFTRESET bit value 1 means reset is
    ongoing, but it behaves the same way clearing after reset is done.

    With the ti-sysc driver handling this automatically based on no sysstatus
    register defined, we see warnings if SYSC_HAS_RESET_STATUS is missing in the
    legacy platform data:

    ti-sysc 48042000.target-module: sysc_flags 00000222 != 00000022
    ti-sysc 48044000.target-module: sysc_flags 00000222 != 00000022
    ti-sysc 48046000.target-module: sysc_flags 00000222 != 00000022
    ...

    Let's fix these warnings by adding SYSC_HAS_RESET_STATUS. Let's also
    remove the useless parentheses while at it.

    If it turns out we do have ti,sysc-omap4 compatible devices without a
    working SOFTRESET bit we can set up additional quirk handling for it.

    Signed-off-by: Tony Lindgren <tony@atomide.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 85381ee9adff0996dd4302132379ea520a9ec4af
Author: Tony Lindgren <tony@atomide.com>
Date:   Tue Sep 24 09:24:28 2019 -0700

    ARM: dts: Fix gpio0 flags for am335x-icev2

    [ Upstream commit 4ef5d76b453908f21341e661a9b6f96862f6f589 ]

    The ti,no-idle-on-init and ti,no-reset-on-init flags need to be at
    the interconnect target module level for the modules that have it
    defined. Otherwise we get the following warnings:

    dts flag should be at module level for ti,no-idle-on-init
    dts flag should be at module level for ti,no-reset-on-init

    Fixes: 87fc89ced3a7 ("ARM: dts: am335x: Move l4 child devices to probe them with ti-
sysc")
    Cc: Lokesh Vutla <lokeshvutla@ti.com>
    Reported-by: Suman Anna <s-anna@ti.com>
```

```
    Reviewed-by: Lokesh Vutla <lokeshvutla@ti.com>
    Signed-off-by: Tony Lindgren <tony@atomide.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 33a89a02d0e12241168e286d4492cfa2d9c2ce1b
Author: Quinn Tran <qutran@marvell.com>
Date:   Thu Sep 12 11:09:10 2019 -0700

    scsi: qla2xxx: Fix N2N link up fail

    [ Upstream commit f3f1938bb673b1b5ad182c4608f5f8a24921eea3 ]

    During link up/bounce, qla driver would do command flush as part of
    cleanup.  In this case, the flush can intefere with FW state.  This patch
    allows FW to be in control of link up.

    Link: https://lore.kernel.org/r/20190912180918.6436-7-hmadhani@marvell.com
    Signed-off-by: Quinn Tran <qutran@marvell.com>
    Signed-off-by: Himanshu Madhani <hmadhani@marvell.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f5c6e4d6069e91085aec4499b8355b8355822660
Author: Quinn Tran <qutran@marvell.com>
Date:   Thu Sep 12 11:09:09 2019 -0700

    scsi: qla2xxx: Fix N2N link reset

    [ Upstream commit 7f2a398d59d658818f3d219645164676fbbc88e8 ]

    Fix stalled link recovery for N2N with FC-NVMe connection.

    Link: https://lore.kernel.org/r/20190912180918.6436-6-hmadhani@marvell.com
    Signed-off-by: Quinn Tran <qutran@marvell.com>
    Signed-off-by: Himanshu Madhani <hmadhani@marvell.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a3bd7819d017ffb66310897ccd2533c49a514c0d
Author: Quinn Tran <qutran@marvell.com>
Date:   Thu Sep 12 11:09:07 2019 -0700

    scsi: qla2xxx: Fix stale mem access on driver unload

    [ Upstream commit fd5564ba54e0d8a9e3e823d311b764232e09eb5f ]

    On driver unload, 'remove_one' thread was allowed to advance, while session
    cleanup still lag behind.  This patch ensures session deletion will finish
    before remove_one can advance.

    Link: https://lore.kernel.org/r/20190912180918.6436-4-hmadhani@marvell.com
    Signed-off-by: Quinn Tran <qutran@marvell.com>
    Signed-off-by: Himanshu Madhani <hmadhani@marvell.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4cfdcf0c8e96ef8b7356dd50b48ad85989a3044d
Author: Quinn Tran <qutran@marvell.com>
Date:   Thu Sep 12 11:09:06 2019 -0700

    scsi: qla2xxx: Fix unbound sleep in fcport delete path.

    [ Upstream commit c3b6a1d397420a0fdd97af2f06abfb78adc370df ]

    There are instances, though rare, where a LOGO request cannot be sent out
```

```
    and the thread in free session done can wait indefinitely. Fix this by
    putting an upper bound to sleep.

    Link: https://lore.kernel.org/r/20190912180918.6436-3-hmadhani@marvell.com
    Signed-off-by: Quinn Tran <qutran@marvell.com>
    Signed-off-by: Himanshu Madhani <hmadhani@marvell.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9537351680e10e14b6aa53136fd4c7a555e21e99
Author: Himanshu Madhani <hmadhani@marvell.com>
Date:   Thu Sep 12 11:09:05 2019 -0700

    scsi: qla2xxx: Silence fwdump template message

    [ Upstream commit 248a445adfc8c33ffd67cf1f2e336578e34f9e21 ]

    Print if fwdt template is present or not, only when
    ql2xextended_error_logging is enabled.

    Link: https://lore.kernel.org/r/20190912180918.6436-2-hmadhani@marvell.com
    Signed-off-by: Himanshu Madhani <hmadhani@marvell.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3a276e2b9aec19e12ac5542febd4011843762ffa
Author: Xiang Chen <chenxiang66@hisilicon.com>
Date:   Sat Sep 7 09:07:30 2019 +0800

    scsi: megaraid: disable device when probe failed after enabled device

    [ Upstream commit 70054aa39a013fa52eff432f2223b8bd5c0048f8 ]

    For pci device, need to disable device when probe failed after enabled
    device.

    Link: https://lore.kernel.org/r/1567818450-173315-1-git-send-email-
chenxiang66@hisilicon.com
    Signed-off-by: Xiang Chen <chenxiang66@hisilicon.com>
    Reviewed-by: John Garry <john.garry@huawei.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 76cc66f70e1322b8edce44bbf35232364ebb9547
Author: Stanley Chu <stanley.chu@mediatek.com>
Date:   Wed Sep 18 12:20:38 2019 +0800

    scsi: ufs: skip shutdown if hba is not powered

    [ Upstream commit f51913eef23f74c3bd07899dc7f1ed6df9e521d8 ]

    In some cases, hba may go through shutdown flow without successful
    initialization and then make system hang.

    For example, if ufshcd_change_power_mode() gets error and leads to
    ufshcd_hba_exit() to release resources of the host, future shutdown flow
    may hang the system since the host register will be accessed in unpowered
    state.

    To solve this issue, simply add checking to skip shutdown for above kind of
    situation.

    Link: https://lore.kernel.org/r/1568780438-28753-1-git-send-email-
stanley.chu@mediatek.com
    Signed-off-by: Stanley Chu <stanley.chu@mediatek.com>
```

```
    Acked-by: Bean Huo <beanhuo@micron.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 23abaa12c4c869acb0d518a265dd487b2a1f8276
Author: Balbir Singh <sblbir@amzn.com>
Date:   Wed Sep 18 00:27:20 2019 +0000

    nvme-pci: Fix a race in controller removal

    [ Upstream commit b224726de5e496dbf78147a66755c3d81e28bdd2 ]

    User space programs like udevd may try to read to partitions at the
    same time the driver detects a namespace is unusable, and may deadlock
    if revalidate_disk() is called while such a process is waiting to
    enter the frozen queue. On detecting a dead namespace, move the disk
    revalidate after unblocking dispatchers that may be holding bd_butex.

    changelog Suggested-by: Keith Busch <kbusch@kernel.org>
    Signed-off-by: Balbir Singh <sblbir@amzn.com>
    Reviewed-by: Keith Busch <kbusch@kernel.org>
    Signed-off-by: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 823be46365f8bdf9564b8de07aa392d1bfa68bcb
Author: Tony Lindgren <tony@atomide.com>
Date:   Mon Sep 23 10:32:38 2019 -0700

    ARM: dts: Fix wrong clocks for dra7 mcasp

    [ Upstream commit 2d3c8ba3cffa00f76bedb713c8c2126c82d8cd13 ]

    The ahclkr clkctrl clock bit 28 only exists for mcasp 1 and 2 on dra7.
    This causes the following warning on beagle-x15:

    ti-sysc 48468000.target-module: could not add child clock ahclkr: -19

    Also the mcasp clkctrl clock bits are wrong:

    For mcasp1 and 2 we have four clocks at bits 28, 24, 22 and 0:

    bit 28 is ahclkr
    bit 24 is ahclkx
    bit 22 is auxclk
    bit 0 is fck

    For mcasp3 to 8 we have three clocks at bits 24, 22 and 0.

    bit 24 is ahclkx
    bit 22 is auxclk
    bit 0 is fck

    We do not have currently mapped auxclk at bit 22 for the drivers, that can
    be added if needed.

    Fixes: 5241ccbf2819 ("ARM: dts: Add missing ranges for dra7 mcasp l3 ports")
    Cc: Suman Anna <s-anna@ti.com>
    Cc: Tero Kristo <t-kristo@ti.com>
    Signed-off-by: Tony Lindgren <tony@atomide.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit be8070f81f2dacb830cb30e1eb50aabbac934c03
Author: Tony Lindgren <tony@atomide.com>
Date:   Mon Sep 23 10:32:37 2019 -0700
```

```
    clk: ti: dra7: Fix mcasp8 clock bits

    [ Upstream commit dd8882a255388ba66175098b1560d4f81c100d30 ]

    There's a typo for dra7 mcasp clkctrl bit, it should be 22 like the other
    macasp instances, and not 24. And in dra7xx_clks[] we have the bits wrong
    way around.

    Fixes: dffa9051d546 ("clk: ti: dra7: add new clkctrl data")
    Cc: linux-clk@vger.kernel.org
    Cc: Michael Turquette <mturquette@baylibre.com>
    Cc: Stephen Boyd <sboyd@kernel.org>
    Cc: Suman Anna <s-anna@ti.com>
    Cc: Tero Kristo <t-kristo@ti.com>
    Acked-by: Stephen Boyd <sboyd@kernel.org>
    Signed-off-by: Tony Lindgren <tony@atomide.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8ad9188c4378ec79973c3b255702e0df78af462f
Author: Lowry Li (Arm Technology China) <Lowry.Li@arm.com>
Date:   Wed Jul 31 11:04:45 2019 +0000

    drm: Clear the fence pointer when writeback job signaled

    [ Upstream commit b1066a123538044117f0a78ba8c6a50cf5a04c86 ]

    During it signals the completion of a writeback job, after releasing
    the out_fence, we'd clear the pointer.

    Check if fence left over in drm_writeback_cleanup_job(), release it.

    Signed-off-by: Lowry Li (Arm Technology China) <lowry.li@arm.com>
    Reviewed-by: Brian Starkey <brian.starkey@arm.com>
    Reviewed-by: James Qian Wang (Arm Technology China) <james.qian.wang@arm.com>
    Signed-off-by: james qian wang (Arm Technology China) <james.qian.wang@arm.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/1564571048-15029-3-git-send-email-
lowry.li@arm.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 51a75141d04a85480f817704beac86909da8193f
Author: Lowry Li (Arm Technology China) <Lowry.Li@arm.com>
Date:   Wed Jul 31 11:04:38 2019 +0000

    drm: Free the writeback_job when it with an empty fb

    [ Upstream commit 8581d51055a08cc6eb061c8856062290e8582ce4 ]

    Adds the check if the writeback_job with an empty fb, then it should
    be freed in atomic_check phase.

    With this change, the driver users will not check empty fb case any more.
    So refined accordingly.

    Signed-off-by: Lowry Li (Arm Technology China) <lowry.li@arm.com>
    Reviewed-by: Liviu Dudau <liviu.dudau@arm.com>
    Reviewed-by: James Qian Wang (Arm Technology China) <james.qian.wang@arm.com>
    Signed-off-by: james qian wang (Arm Technology China) <james.qian.wang@arm.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/1564571048-15029-2-git-send-email-
lowry.li@arm.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```