```
commit e282393f9d0cd66cee8c68a80f4936f46c449b2d
Author: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
Date:   Fri Jun 9 10:48:26 2023 +0200

    Linux 6.3.7

    Link: https://lore.kernel.org/r/20230607200922.978677727@linuxfoundation.org
    Tested-by: Ronald Warsow <rwarsow@gmx.de>
    Tested-by: Shuah Khan <skhan@linuxfoundation.org>
    Tested-by: Florian Fainelli <florian.fainelli@broadcom.com>
    Tested-by: Markus Reichelt <lkt+2023@mareichelt.com>
    Tested-by: Chris Paterson (CIP) <chris.paterson2@renesas.com>
    Tested-by: Conor Dooley <conor.dooley@microchip.com>
    Tested-by: Ron Economos <re@w6rz.net>
    Tested-by: Linux Kernel Functional Testing <lkft@linaro.org>
    Tested-by: Bagas Sanjaya <bagasdotme@gmail.com>
    Tested-by: Justin M. Forbes <jforbes@fedoraproject.org>
    Link: https://lore.kernel.org/r/20230608175722.489602717@linuxfoundation.org
    Tested-by: Ronald Warsow <rwarsow@gmx.de>
    Tested-by: Florian Fainelli <florian.fainelli@broadcom.com>
    Tested-by: Linux Kernel Functional Testing <lkft@linaro.org>
    Tested-by: Chris Paterson (CIP) <chris.paterson2@renesas.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 9ca777c1ef9a20953aa252923f53fe42b5b986c0
Author: Theodore Ts'o <tytso@mit.edu>
Date:   Fri May 26 23:57:29 2023 -0400

    ext4: enable the lazy init thread when remounting read/write

    commit eb1f822c76beeaa76ab8b6737ab9dc9f9798408c upstream.

    In commit a44be64bbecb ("ext4: don't clear SB_RDONLY when remounting
    r/w until quota is re-enabled") we defer clearing tyhe SB_RDONLY flag
    in struct super.  However, we didn't defer when we checked sb_rdonly()
    to determine the lazy itable init thread should be enabled, with the
    next result that the lazy inode table initialization would not be
    properly started.  This can cause generic/231 to fail in ext4's
    nojournal mode.

    Fix this by moving when we decide to start or stop the lazy itable
    init thread to after we clear the SB_RDONLY flag when we are
    remounting the file system read/write.

    Fixes a44be64bbecb ("ext4: don't clear SB_RDONLY when remounting r/w until...")

    Signed-off-by: Theodore Ts'o <tytso@mit.edu>
    Link: https://lore.kernel.org/r/20230527035729.1001605-1-tytso@mit.edu
    Signed-off-by: Theodore Ts'o <tytso@mit.edu>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e6d1562dd4e9cf8da89edd21c4453893bfe0d4ac
Author: Nathan Chancellor <nathan@kernel.org>
Date:   Mon Jun 5 14:15:08 2023 -0700

    riscv: vmlinux.lds.S: Explicitly handle '.got' section

    This patch is for linux-6.3.y only, it has no direct mainline
    equivalent.

    LLVM 17 will now use the GOT for extern weak symbols when using the
    medany model, which causes a linker orphan section warning on
    linux-6.3.y:
```

```
       ld.lld: warning: <internal>:(.got) is being placed in '.got'

    This is not an issue in mainline because handling of the .got section
    was added by commit 39b33072941f ("riscv: Introduce CONFIG_RELOCATABLE")
    and further extended by commit 26e7aacb83df ("riscv: Allow to downgrade
    paging mode from the command line") in 6.4-rc1. Neither of these changes
    are suitable for stable, so add explicit handling of the .got section in
    a standalone change to align 6.3 and mainline, which addresses the
    warning.

    This is only an issue for 6.3 because commit f4b71bff8d85 ("riscv:
    select ARCH_WANT_LD_ORPHAN_WARN for !XIP_KERNEL") landed in 6.3-rc1, so
    earlier releases will not see this warning because it will not be
    enabled.

    Closes: https://github.com/ClangBuiltLinux/linux/issues/1865
    Link: https://github.com/llvm/llvm-
project/commit/a178ba9fbd0a27057dc2fa4cb53c76caa013caac
    Signed-off-by: Nathan Chancellor <nathan@kernel.org>
    Reviewed-by: Conor Dooley <conor.dooley@microchip.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 69ebe82c73f4f9f4b49ed3b35ce347af20716d0a
Author: Darrick J. Wong <djwong@kernel.org>
Date:   Wed Apr 12 15:49:23 2023 +1000

    xfs: verify buffer contents when we skip log replay

    commit 22ed903eee23a5b174e240f1cdfa9acf393a5210 upstream.

    syzbot detected a crash during log recovery:

    XFS (loop0): Mounting V5 Filesystem bfdc47fc-10d8-4eed-a562-11a831b3f791
    XFS (loop0): Torn write (CRC failure) detected at log block 0x180. Truncating head block
from 0x200.
    XFS (loop0): Starting recovery (logdev: internal)
    ==================================================================
    BUG: KASAN: slab-out-of-bounds in xfs_btree_lookup_get_block+0x15c/0x6d0
fs/xfs/libxfs/xfs_btree.c:1813
    Read of size 8 at addr ffff88807e89f258 by task syz-executor132/5074

    CPU: 0 PID: 5074 Comm: syz-executor132 Not tainted 6.2.0-rc1-syzkaller #0
    Hardware name: Google Google Compute Engine/Google Compute Engine, BIOS Google
10/26/2022
    Call Trace:
     <TASK>
     __dump_stack lib/dump_stack.c:88 [inline]
     dump_stack_lvl+0x1b1/0x290 lib/dump_stack.c:106
     print_address_description+0x74/0x340 mm/kasan/report.c:306
     print_report+0x107/0x1f0 mm/kasan/report.c:417
     kasan_report+0xcd/0x100 mm/kasan/report.c:517
     xfs_btree_lookup_get_block+0x15c/0x6d0 fs/xfs/libxfs/xfs_btree.c:1813
     xfs_btree_lookup+0x346/0x12c0 fs/xfs/libxfs/xfs_btree.c:1913
     xfs_btree_simple_query_range+0xde/0x6a0 fs/xfs/libxfs/xfs_btree.c:4713
     xfs_btree_query_range+0x2db/0x380 fs/xfs/libxfs/xfs_btree.c:4953
     xfs_refcount_recover_cow_leftovers+0x2d1/0xa60 fs/xfs/libxfs/xfs_refcount.c:1946
     xfs_reflink_recover_cow+0xab/0x1b0 fs/xfs/xfs_reflink.c:930
     xlog_recover_finish+0x824/0x920 fs/xfs/xfs_log_recover.c:3493
     xfs_log_mount_finish+0x1ec/0x3d0 fs/xfs/xfs_log.c:829
     xfs_mountfs+0x146a/0x1ef0 fs/xfs/xfs_mount.c:933
     xfs_fs_fill_super+0xf95/0x11f0 fs/xfs/xfs_super.c:1666
     get_tree_bdev+0x400/0x620 fs/super.c:1282
     vfs_get_tree+0x88/0x270 fs/super.c:1489
     do_new_mount+0x289/0xad0 fs/namespace.c:3145
     do_mount fs/namespace.c:3488 [inline]
```

```
      __do_sys_mount fs/namespace.c:3697 [inline]
      __se_sys_mount+0x2d3/0x3c0 fs/namespace.c:3674
      do_syscall_x64 arch/x86/entry/common.c:50 [inline]
      do_syscall_64+0x3d/0xb0 arch/x86/entry/common.c:80
      entry_SYSCALL_64_after_hwframe+0x63/0xcd
    RIP: 0033:0x7f89fa3f4aca
    Code: 83 c4 08 5b 5d c3 66 2e 0f 1f 84 00 00 00 00 00 c3 66 2e 0f 1f 84 00 00 00 00 00
0f 1f 44 00 00 49 89 ca b8 a5 00 00 00 0f 05 <48> 3d 01 f0 ff ff 73 01 c3 48 c7 c1 c0 ff ff
ff f7 d8 64 89 01 48
    RSP: 002b:00007fffd5fb5ef8 EFLAGS: 00000206 ORIG_RAX: 00000000000000a5
    RAX: ffffffffffffffda RBX: 00646975756f6e2c RCX: 00007f89fa3f4aca
    RDX: 0000000020000100 RSI: 0000000020009640 RDI: 00007fffd5fb5f10
    RBP: 00007fffd5fb5f10 R08: 00007fffd5fb5f50 R09: 000000000000970d
    R10: 0000000000200800 R11: 0000000000000206 R12: 0000000000000004
    R13: 0000555556c6b2c0 R14: 0000000000200800 R15: 00007fffd5fb5f50
     </TASK>

    The fuzzed image contains an AGF with an obviously garbage
    agf_refcount_level value of 32, and a dirty log with a buffer log item
    for that AGF.  The ondisk AGF has a higher LSN than the recovered log
    item.  xlog_recover_buf_commit_pass2 reads the buffer, compares the
    LSNs, and decides to skip replay because the ondisk buffer appears to be
    newer.

    Unfortunately, the ondisk buffer is corrupt, but recovery just read the
    buffer with no buffer ops specified:

            error = xfs_buf_read(mp->m_ddev_targp, buf_f->blf_blkno,
                            buf_f->blf_len, buf_flags, &bp, NULL);

    Skipping the buffer leaves its contents in memory unverified.  This sets
    us up for a kernel crash because xfs_refcount_recover_cow_leftovers
    reads the buffer (which is still around in XBF_DONE state, so no read
    verification) and creates a refcountbt cursor of height 32.  This is
    impossible so we run off the end of the cursor object and crash.

    Fix this by invoking the verifier on all skipped buffers and aborting
    log recovery if the ondisk buffer is corrupt.  It might be smarter to
    force replay the log item atop the buffer and then see if it'll pass the
    write verifier (like ext4 does) but for now let's go with the
    conservative option where we stop immediately.

    Link: https://syzkaller.appspot.com/bug?extid=7e9494b8b399902e994e
    Signed-off-by: Darrick J. Wong <djwong@kernel.org>
    Reviewed-by: Dave Chinner <dchinner@redhat.com>
    Signed-off-by: Dave Chinner <david@fromorbit.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 9b53bb3b6289ef809e9558e60290316abf6cd6ea
Author: Vasant Hegde <vasant.hegde@amd.com>
Date:   Thu May 18 05:43:51 2023 +0000

    iommu/amd/pgtbl_v2: Fix domain max address

    commit 11c439a19466e7feaccdbce148a75372fddaf4e9 upstream.

    IOMMU v2 page table supports 4 level (47 bit) or 5 level (56 bit) virtual
    address space. Current code assumes it can support 64bit IOVA address
    space. If IOVA allocator allocates virtual address > 47/56 bit (depending
    on page table level) then it will do wrong mapping and cause invalid
    translation.

    Hence adjust aperture size to use max address supported by the page table.

    Reported-by: Jerry Snitselaar <jsnitsel@redhat.com>
```

```
    Fixes: aaac38f61487 ("iommu/amd: Initial support for AMD IOMMU v2 page table")
    Cc: <Stable@vger.kernel.org>  # v6.0+
    Cc: Suravee Suthikulpanit <suravee.suthikulpanit@amd.com>
    Signed-off-by: Vasant Hegde <vasant.hegde@amd.com>
    Reviewed-by: Jerry Snitselaar <jsnitsel@redhat.com>
    Link: https://lore.kernel.org/r/20230518054351.9626-1-vasant.hegde@amd.com
    Signed-off-by: Joerg Roedel <jroedel@suse.de>
    [ Modified to work with "V2 with 4 level page table" only - Vasant ]
    Signed-off-by: Vasant Hegde <vasant.hegde@amd.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 7511a699c2265790ccaf3f3c1c57545405627075
Author: Lino Sanfilippo <l.sanfilippo@kunbus.com>
Date:   Thu Nov 24 14:55:34 2022 +0100

    tpm, tpm_tis: Request threaded interrupt handler

    commit 0c7e66e5fd69bf21034c9a9b081d7de7c3eb2cea upstream.

    The TIS interrupt handler at least has to read and write the interrupt
    status register. In case of SPI both operations result in a call to
    tpm_tis_spi_transfer() which uses the bus_lock_mutex of the spi device
    and thus must only be called from a sleepable context.

    To ensure this request a threaded interrupt handler.

    Signed-off-by: Lino Sanfilippo <l.sanfilippo@kunbus.com>
    Tested-by: Michael Niewöhner <linux@mniewoehner.de>
    Tested-by: Jarkko Sakkinen <jarkko@kernel.org>
    Reviewed-by: Jarkko Sakkinen <jarkko@kernel.org>
    Signed-off-by: Jarkko Sakkinen <jarkko@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ea115e9e2cf6b2dd97e4ca85d98e679b51d34dc8
Author: Jim Wylder <jwylder@google.com>
Date:   Wed May 17 10:20:11 2023 -0500

    regmap: Account for register length when chunking

    commit 3981514180c987a79ea98f0ae06a7cbf58a9ac0f upstream.

    Currently, when regmap_raw_write() splits the data, it uses the
    max_raw_write value defined for the bus.  For any bus that includes
    the target register address in the max_raw_write value, the chunked
    transmission will always exceed the maximum transmission length.
    To avoid this problem, subtract the length of the register and the
    padding from the maximum transmission.

    Signed-off-by: Jim Wylder <jwylder@google.com
    Link: https://lore.kernel.org/r/20230517152444.3690870-2-jwylder@google.com
    Signed-off-by: Mark Brown <broonie@kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 6b31c11d879e147715e50a5bd8e1f0294c342292
Author: Roberto Sassu <roberto.sassu@huawei.com>
Date:   Thu Dec 8 10:56:46 2022 +0100

    KEYS: asymmetric: Copy sig and digest in public_key_verify_signature()

    commit c3d03e8e35e005e1a614e51bb59053eeb5857f76 upstream.

    Commit ac4e97abce9b8 ("scatterlist: sg_set_buf() argument must be in linear
    mapping") checks that both the signature and the digest reside in the
    linear mapping area.
```

```
    However, more recently commit ba14a194a434c ("fork: Add generic vmalloced
    stack support") made it possible to move the stack in the vmalloc area,
    which is not contiguous, and thus not suitable for sg_set_buf() which needs
    adjacent pages.

    Always make a copy of the signature and digest in the same buffer used to
    store the key and its parameters, and pass them to sg_init_one(). Prefer it
    to conditionally doing the copy if necessary, to keep the code simple. The
    buffer allocated with kmalloc() is in the linear mapping area.

    Cc: stable@vger.kernel.org # 4.9.x
    Fixes: ba14a194a434 ("fork: Add generic vmalloced stack support")
    Link: https://lore.kernel.org/linux-integrity/Y4pIpxbjBdajymBJ@sol.localdomain/
    Suggested-by: Eric Biggers <ebiggers@kernel.org>
    Signed-off-by: Roberto Sassu <roberto.sassu@huawei.com>
    Reviewed-by: Eric Biggers <ebiggers@google.com>
    Tested-by: Stefan Berger <stefanb@linux.ibm.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 060fb8489a51544de2a8db2ba5e7506a5082cdaf
Author: Kuan-Ting Chen <h3xrabbit@gmail.com>
Date:   Fri May 19 23:00:24 2023 +0900

    ksmbd: fix multiple out-of-bounds read during context decoding

    commit 0512a5f89e1fae74251fde6893ff634f1c96c6fb upstream.

    Check the remaining data length before accessing the context structure
    to ensure that the entire structure is contained within the packet.
    Additionally, since the context data length `ctxt_len` has already been
    checked against the total packet length `len_of_ctxts`, update the
    comparison to use `ctxt_len`.

    Cc: stable@vger.kernel.org
    Signed-off-by: Kuan-Ting Chen <h3xrabbit@gmail.com>
    Acked-by: Namjae Jeon <linkinjeon@kernel.org>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 0ec8ad33f2aca16ad7946ff9b015677a99484993
Author: Kuan-Ting Chen <h3xrabbit@gmail.com>
Date:   Fri May 19 22:59:28 2023 +0900

    ksmbd: fix slab-out-of-bounds read in smb2_handle_negotiate

    commit d738950f112c8f40f0515fe967db998e8235a175 upstream.

    Check request_buf length first to avoid out-of-bounds read by
    req->DialectCount.

    [ 3350.990282] BUG: KASAN: slab-out-of-bounds in smb2_handle_negotiate+0x35d7/0x3e60
    [ 3350.990282] Read of size 2 at addr ffff88810ad61346 by task kworker/5:0/276
    [ 3351.000406] Workqueue: ksmbd-io handle_ksmbd_work
    [ 3351.003499] Call Trace:
    [ 3351.006473]  <TASK>
    [ 3351.006473]  dump_stack_lvl+0x8d/0xe0
    [ 3351.006473]  print_report+0xcc/0x620
    [ 3351.006473]  kasan_report+0x92/0xc0
    [ 3351.006473]  smb2_handle_negotiate+0x35d7/0x3e60
    [ 3351.014760]  ksmbd_smb_negotiate_common+0x7a7/0xf00
    [ 3351.014760]  handle_ksmbd_work+0x3f7/0x12d0
    [ 3351.014760]  process_one_work+0xa85/0x1780

    Cc: stable@vger.kernel.org
    Signed-off-by: Kuan-Ting Chen <h3xrabbit@gmail.com>
```

```
    Acked-by: Namjae Jeon <linkinjeon@kernel.org>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 82c0ee2a15ba1e8406109269f9f9f6a9c1cea29b7e
Author: Namjae Jeon <linkinjeon@kernel.org>
Date:   Fri May 19 23:11:33 2023 +0900

    ksmbd: fix incorrect AllocationSize set in smb2_get_info

    commit 6cc2268f5647cbfde3d4fc2e4ee005070ea3a8d2 upstream.

    If filesystem support sparse file, ksmbd should return allocated size
    using ->i_blocks instead of stat->size. This fix generic/694 xfstests.

    Cc: stable@vger.kernel.org
    Signed-off-by: Namjae Jeon <linkinjeon@kernel.org>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f8f40dd95cf3ca42837b1c7aa082128e13f3b215
Author: Namjae Jeon <linkinjeon@kernel.org>
Date:   Fri May 19 23:09:48 2023 +0900

    ksmbd: fix UAF issue from opinfo->conn

    commit 36322523dddb11107e9f7f528675a0dec2536103 upstream.

    If opinfo->conn is another connection and while ksmbd send oplock break
    request to cient on current connection, The connection for opinfo->conn
    can be disconnect and conn could be freed. When sending oplock break
    request, this ksmbd_conn can be used and cause user-after-free issue.
    When getting opinfo from the list, ksmbd check connection is being
    released. If it is not released, Increase ->r_count to wait that connection
    is freed.

    Cc: stable@vger.kernel.org
    Reported-by: Per Forlin <per.forlin@axis.com>
    Tested-by: Per Forlin <per.forlin@axis.com>
    Signed-off-by: Namjae Jeon <linkinjeon@kernel.org>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit eff55712c34cf405b810ebf66e76ad764c21140d
Author: Namjae Jeon <linkinjeon@kernel.org>
Date:   Fri May 12 23:29:12 2023 +0900

    ksmbd: fix credit count leakage

    commit 84c5aa47925a1f40d698b6a6a2bf67e99617433d upstream.

    This patch fix the failure from smb2.credits.single_req_credits_granted
    test. When client send 8192 credit request, ksmbd return 8191 credit
    granted. ksmbd should give maximum possible credits that must be granted
    within the range of not exceeding the max credit to client.

    Cc: stable@vger.kernel.org
    Signed-off-by: Namjae Jeon <linkinjeon@kernel.org>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 36461b145ed863845494416a1c8ef727df9df114
Author: Sean Christopherson <seanjc@google.com>
Date:   Fri Jun 2 16:32:48 2023 -0700
```

```
    KVM: x86: Bail from kvm_recalculate_phys_map() if x2APIC ID is out-of-bounds

    commit 4364b287982bd05bfafa461c80650c732001974b upstream.

    Bail from kvm_recalculate_phys_map() and disable the optimized map if the
    target vCPU's x2APIC ID is out-of-bounds, i.e. if the vCPU was added
    and/or enabled its local APIC after the map was allocated.  This fixes an
    out-of-bounds access bug in the !x2apic_format path where KVM would write
    beyond the end of phys_map.

    Check the x2APIC ID regardless of whether or not x2APIC is enabled,
    as KVM's hardcodes x2APIC ID to be the vCPU ID, i.e. it can't change, and
    the map allocation in kvm_recalculate_apic_map() doesn't check for x2APIC
    being enabled, i.e. the check won't get false postivies.

    Note, this also affects the x2apic_format path, which previously just
    ignored the "x2apic_id > new->max_apic_id" case.  That too is arguably a
    bug fix, as ignoring the vCPU meant that KVM would not send interrupts to
    the vCPU until the next map recalculation.  In practice, that "bug" is
    likely benign as a newly present vCPU/APIC would immediately trigger a
    recalc.  But, there's no functional downside to disabling the map, and
    a future patch will gracefully handle the -E2BIG case by retrying instead
    of simply disabling the optimized map.

    Opportunistically add a sanity check on the xAPIC ID size, along with a
    comment explaining why the xAPIC ID is guaranteed to be "good".

    Reported-by: Michal Luczaj <mhal@rbox.co>
    Fixes: 5b84b0291702 ("KVM: x86: Honor architectural behavior for aliased 8-bit APIC
IDs")
    Cc: stable@vger.kernel.org
    Link: https://lore.kernel.org/r/20230602233250.1014316-2-seanjc@google.com
    Signed-off-by: Sean Christopherson <seanjc@google.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 7ed93eed4a1a742bae2eb967612d1710cd2e2aff
Author: Sean Christopherson <seanjc@google.com>
Date:   Thu Jun 1 18:19:19 2023 -0700

    KVM: x86: Account fastpath-only VM-Exits in vCPU stats

    commit 8b703a49c9df5e74870381ad7ba9c85d8a74ed2c upstream.

    Increment vcpu->stat.exits when handling a fastpath VM-Exit without
    going through any part of the "slow" path.  Not bumping the exits stat
    can result in wildly misleading exit counts, e.g. if the primary reason
    the guest is exiting is to program the TSC deadline timer.

    Fixes: 404d5d7bff0d ("KVM: X86: Introduce more exit_fastpath_completion enum values")
    Cc: stable@vger.kernel.org
    Link: https://lore.kernel.org/r/20230602011920.787844-2-seanjc@google.com
    Signed-off-by: Sean Christopherson <seanjc@google.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 2b46a2c480e31062715f423837cefca013824f1e
Author: Sean Christopherson <seanjc@google.com>
Date:   Thu Jun 1 18:01:37 2023 -0700

    KVM: x86/mmu: Grab memslot for correct address space in NX recovery worker

    commit 817fa998362d6ea9fabd5e97af8e9e2eb5f0e6f2 upstream.

    Factor in the address space (non-SMM vs. SMM) of the target shadow page
    when recovering potential NX huge pages, otherwise KVM will retrieve the
    wrong memslot when zapping shadow pages that were created for SMM.  The
```

```
    bug most visibly manifests as a WARN on the memslot being non-NULL, but
    the worst case scenario is that KVM could unaccount the shadow page
    without ensuring KVM won't install a huge page, i.e. if the non-SMM slot
    is being dirty logged, but the SMM slot is not.

     ------------[ cut here ]------------
     WARNING: CPU: 1 PID: 3911 at arch/x86/kvm/mmu/mmu.c:7015
     kvm_nx_huge_page_recovery_worker+0x38c/0x3d0 [kvm]
     CPU: 1 PID: 3911 Comm: kvm-nx-lpage-re
     RIP: 0010:kvm_nx_huge_page_recovery_worker+0x38c/0x3d0 [kvm]
     RSP: 0018:ffff99b284f0be68 EFLAGS: 00010246
     RAX: 0000000000000000 RBX: ffff99b284edd000 RCX: 0000000000000000
     RDX: 0000000000000000 RSI: 0000000000000000 RDI: 0000000000000000
     RBP: ffff9271397024e0 R08: 0000000000000000 R09: ffff927139702450
     R10: 0000000000000000 R11: 0000000000000001 R12: ffff99b284f0be98
     R13: 0000000000000000 R14: ffff9270991fcd80 R15: 0000000000000003
     FS:  0000000000000000(0000) GS:ffff927f9f640000(0000) knlGS:0000000000000000
     CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
     CR2: 00007f0aacad3ae0 CR3: 000000088fc2c005 CR4: 0000000003726e0
     Call Trace:
      <TASK>
    __pfx_kvm_nx_huge_page_recovery_worker+0x10/0x10 [kvm]
      kvm_vm_worker_thread+0x106/0x1c0 [kvm]
      kthread+0xd9/0x100
      ret_from_fork+0x2c/0x50
      </TASK>
     ---[ end trace 0000000000000000 ]---

    This bug was exposed by commit edbdb43fc96b ("KVM: x86: Preserve TDP MMU
    roots until they are explicitly invalidated"), which allowed KVM to retain
    SMM TDP MMU roots effectively indefinitely.  Before commit edbdb43fc96b,
    KVM would zap all SMM TDP MMU roots and thus all SMM TDP MMU shadow pages
    once all vCPUs exited SMM, which made the window where this bug (recovering
    an SMM NX huge page) could be encountered quite tiny.  To hit the bug, the
    NX recovery thread would have to run while at least one vCPU was in SMM.
    Most VMs typically only use SMM during boot, and so the problematic shadow
    pages were gone by the time the NX recovery thread ran.

    Now that KVM preserves TDP MMU roots until they are explicitly invalidated
    (e.g. by a memslot deletion), the window to trigger the bug is effectively
    never closed because most VMMs don't delete memslots after boot (except
    for a handful of special scenarios).

    Fixes: eb298605705a ("KVM: x86/mmu: Do not recover dirty-tracked NX Huge Pages")
    Reported-by: Fabio Coatti <fabio.coatti@gmail.com>
    Closes:
https://lore.kernel.org/all/CADpTngX9LESCdHVu_2mQkNGena_Ng2CphWNwsRGSMxzDsTjU2A@mail.gmail.c
om
    Cc: stable@vger.kernel.org
    Link: https://lore.kernel.org/r/20230602010137.784664-1-seanjc@google.com
    Signed-off-by: Sean Christopherson <seanjc@google.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e02e3f510d046e5afa1798a7cbfa001f7457da0f
Author: Oliver Upton <oliver.upton@linux.dev>
Date:   Tue May 30 19:32:13 2023 +0000

    KVM: arm64: Drop last page ref in kvm_pgtable_stage2_free_removed()

    commit f6a27d6dc51b288106adaf053cff9c9b9cc12c4e upstream.

    The reference count on page table allocations is increased for every
    'counted' PTE (valid or donated) in the table in addition to the initial
    reference from ->zalloc_page(). kvm_pgtable_stage2_free_removed() fails
    to drop the last reference on the root of the table walk, meaning we
```

```
    leak memory.

    Fix it by dropping the last reference after the free walker returns,
    at which point all references for 'counted' PTEs have been released.

    Cc: stable@vger.kernel.org
    Fixes: 5c359cca1faf ("KVM: arm64: Tear down unlinked stage-2 subtree after break-before-
make")
    Reported-by: Yu Zhao <yuzhao@google.com>
    Signed-off-by: Oliver Upton <oliver.upton@linux.dev>
    Tested-by: Yu Zhao <yuzhao@google.com>
    Signed-off-by: Marc Zyngier <maz@kernel.org>
    Link: https://lore.kernel.org/r/20230530193213.1663411-1-oliver.upton@linux.dev
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 5bd8f8522e4bbd4677a973e70260a77c95a34806
Author: Akihiko Odaki <akihiko.odaki@daynix.com>
Date:   Tue May 30 11:46:51 2023 +0900

    KVM: arm64: Populate fault info for watchpoint

    commit 811154e234db72f0a11557a84ba9640f8b3bc823 upstream.

    When handling ESR_ELx_EC_WATCHPT_LOW, far_el2 member of struct
    kvm_vcpu_fault_info will be copied to far member of struct
    kvm_debug_exit_arch and exposed to the userspace. The userspace will
    see stale values from older faults if the fault info does not get
    populated.

    Fixes: 8fb2046180a0 ("KVM: arm64: Move early handlers to per-EC handlers")
    Suggested-by: Marc Zyngier <maz@kernel.org>
    Signed-off-by: Akihiko Odaki <akihiko.odaki@daynix.com>
    Signed-off-by: Marc Zyngier <maz@kernel.org>
    Link: https://lore.kernel.org/r/20230530024651.10014-1-akihiko.odaki@daynix.com
    Cc: stable@vger.kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 75d4bb5941fcdda8c62f07e58f765897140f6388
Author: Mirsad Goran Todorovac <mirsad.todorovac@alu.unizg.hr>
Date:   Tue May 9 10:47:49 2023 +0200

    test_firmware: fix the memory leak of the allocated firmware buffer

    commit 48e156023059e57a8fc68b498439832f7600ffff upstream.

    The following kernel memory leak was noticed after running
    tools/testing/selftests/firmware/fw_run_tests.sh:

    [root@pc-mtodorov firmware]# cat /sys/kernel/debug/kmemleak
    .
    .
    .
    unreferenced object 0xffff955389bc3400 (size 1024):
      comm "test_firmware-0", pid 5451, jiffies 4294944822 (age 65.652s)
      hex dump (first 32 bytes):
        47 48 34 35 36 37 0a 00 00 00 00 00 00 00 00 00  GH4567..........
        00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
      backtrace:
        [<ffffffff962f5dec>] slab_post_alloc_hook+0x8c/0x3c0
        [<ffffffff962fcca4>] __kmem_cache_alloc_node+0x184/0x240
        [<ffffffff962704de>] kmalloc_trace+0x2e/0xc0
        [<ffffffff9665b42d>] test_fw_run_batch_request+0x9d/0x180
        [<ffffffff95fd813b>] kthread+0x10b/0x140
        [<ffffffff95e033e9>] ret_from_fork+0x29/0x50
    unreferenced object 0xffff9553c334b400 (size 1024):
```

```
        comm "test_firmware-1", pid 5452, jiffies 4294944822 (age 65.652s)
        hex dump (first 32 bytes):
          47 48 34 35 36 37 0a 00 00 00 00 00 00 00 00 00  GH4567..........
          00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
        backtrace:
          [<ffffffff962f5dec>] slab_post_alloc_hook+0x8c/0x3c0
          [<ffffffff962fcca4>] __kmem_cache_alloc_node+0x184/0x240
          [<ffffffff962704de>] kmalloc_trace+0x2e/0xc0
          [<ffffffff9665b42d>] test_fw_run_batch_request+0x9d/0x180
          [<ffffffff95fd813b>] kthread+0x10b/0x140
          [<ffffffff95e033e9>] ret_from_fork+0x29/0x50
      unreferenced object 0xffff9553c334f000 (size 1024):
        comm "test_firmware-2", pid 5453, jiffies 4294944822 (age 65.652s)
        hex dump (first 32 bytes):
          47 48 34 35 36 37 0a 00 00 00 00 00 00 00 00 00  GH4567..........
          00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
        backtrace:
          [<ffffffff962f5dec>] slab_post_alloc_hook+0x8c/0x3c0
          [<ffffffff962fcca4>] __kmem_cache_alloc_node+0x184/0x240
          [<ffffffff962704de>] kmalloc_trace+0x2e/0xc0
          [<ffffffff9665b42d>] test_fw_run_batch_request+0x9d/0x180
          [<ffffffff95fd813b>] kthread+0x10b/0x140
          [<ffffffff95e033e9>] ret_from_fork+0x29/0x50
      unreferenced object 0xffff9553c3348400 (size 1024):
        comm "test_firmware-3", pid 5454, jiffies 4294944822 (age 65.652s)
        hex dump (first 32 bytes):
          47 48 34 35 36 37 0a 00 00 00 00 00 00 00 00 00  GH4567..........
          00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
        backtrace:
          [<ffffffff962f5dec>] slab_post_alloc_hook+0x8c/0x3c0
          [<ffffffff962fcca4>] __kmem_cache_alloc_node+0x184/0x240
          [<ffffffff962704de>] kmalloc_trace+0x2e/0xc0
          [<ffffffff9665b42d>] test_fw_run_batch_request+0x9d/0x180
          [<ffffffff95fd813b>] kthread+0x10b/0x140
          [<ffffffff95e033e9>] ret_from_fork+0x29/0x50
      [root@pc-mtodorov firmware]#

      Note that the size 1024 corresponds to the size of the test firmware
      buffer. The actual number of the buffers leaked is around 70-110,
      depending on the test run.

      The cause of the leak is the following:

      request_partial_firmware_into_buf() and request_firmware_into_buf()
      provided firmware buffer isn't released on release_firmware(), we
      have allocated it and we are responsible for deallocating it manually.
      This is introduced in a number of context where previously only
      release_firmware() was called, which was insufficient.

      Reported-by: Mirsad Goran Todorovac <mirsad.todorovac@alu.unizg.hr>
      Fixes: 7feebfa487b92 ("test_firmware: add support for request_firmware_into_buf")
      Cc: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
      Cc: Dan Carpenter <error27@gmail.com>
      Cc: Takashi Iwai <tiwai@suse.de>
      Cc: Luis Chamberlain <mcgrof@kernel.org>
      Cc: Russ Weight <russell.h.weight@intel.com>
      Cc: Tianfei zhang <tianfei.zhang@intel.com>
      Cc: Christophe JAILLET <christophe.jaillet@wanadoo.fr>
      Cc: Zhengchao Shao <shaozhengchao@huawei.com>
      Cc: Colin Ian King <colin.i.king@gmail.com>
      Cc: linux-kernel@vger.kernel.org
      Cc: Kees Cook <keescook@chromium.org>
      Cc: Scott Branden <sbranden@broadcom.com>
      Cc: Luis R. Rodriguez <mcgrof@kernel.org>
      Cc: linux-kselftest@vger.kernel.org
```

```
    Cc: stable@vger.kernel.org # v5.4
    Signed-off-by: Mirsad Goran Todorovac <mirsad.todorovac@alu.unizg.hr>
    Link: https://lore.kernel.org/r/20230509084746.48259-3-mirsad.todorovac@alu.unizg.hr
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e97e83f657984e52e7d1ee9afaff0ffe359bce94
Author: Mirsad Goran Todorovac <mirsad.todorovac@alu.unizg.hr>
Date:   Tue May 9 10:47:47 2023 +0200

    test_firmware: fix a memory leak with reqs buffer

    commit be37bed754ed90b2655382f93f9724b3c1aae847 upstream.

    Dan Carpenter spotted that test_fw_config->reqs will be leaked if
    trigger_batched_requests_store() is called two or more times.
    The same appears with trigger_batched_requests_async_store().

    This bug wasn't trigger by the tests, but observed by Dan's visual
    inspection of the code.

    The recommended workaround was to return -EBUSY if test_fw_config->reqs
    is already allocated.

    Fixes: 7feebfa487b92 ("test_firmware: add support for request_firmware_into_buf")
    Cc: Luis Chamberlain <mcgrof@kernel.org>
    Cc: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Cc: Russ Weight <russell.h.weight@intel.com>
    Cc: Tianfei Zhang <tianfei.zhang@intel.com>
    Cc: Shuah Khan <shuah@kernel.org>
    Cc: Colin Ian King <colin.i.king@gmail.com>
    Cc: Randy Dunlap <rdunlap@infradead.org>
    Cc: linux-kselftest@vger.kernel.org
    Cc: stable@vger.kernel.org # v5.4
    Suggested-by: Dan Carpenter <error27@gmail.com>
    Suggested-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Mirsad Goran Todorovac <mirsad.todorovac@alu.unizg.hr>
    Reviewed-by: Dan Carpenter <dan.carpenter@linaro.org>
    Acked-by: Luis Chamberlain <mcgrof@kernel.org>
    Link: https://lore.kernel.org/r/20230509084746.48259-2-mirsad.todorovac@alu.unizg.hr
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ebfd582b5b02f3f2c83099c4cbc8062f7a650d4b
Author: Mirsad Goran Todorovac <mirsad.todorovac@alu.unizg.hr>
Date:   Tue May 9 10:47:45 2023 +0200

    test_firmware: prevent race conditions by a correct implementation of locking

    commit 4acfe3dfde685a5a9eaec5555351918e2d7266a1 upstream.

    Dan Carpenter spotted a race condition in a couple of situations like
    these in the test_firmware driver:

    static int test_dev_config_update_u8(const char *buf, size_t size, u8 *cfg)
    {
            u8 val;
            int ret;

            ret = kstrtou8(buf, 10, &val);
            if (ret)
                    return ret;

            mutex_lock(&test_fw_mutex);
            *(u8 *)cfg = val;
```

```
        mutex_unlock(&test_fw_mutex);

        /* Always return full write size even if we didn't consume all */
        return size;
}

static ssize_t config_num_requests_store(struct device *dev,
                                         struct device_attribute *attr,
                                         const char *buf, size_t count)
{
        int rc;

        mutex_lock(&test_fw_mutex);
        if (test_fw_config->reqs) {
                pr_err("Must call release_all_firmware prior to changing config\n");
                rc = -EINVAL;
                mutex_unlock(&test_fw_mutex);
                goto out;
        }
        mutex_unlock(&test_fw_mutex);

        rc = test_dev_config_update_u8(buf, count,
                                       &test_fw_config->num_requests);

out:
        return rc;
}

static ssize_t config_read_fw_idx_store(struct device *dev,
                                        struct device_attribute *attr,
                                        const char *buf, size_t count)
{
        return test_dev_config_update_u8(buf, count,
                                         &test_fw_config->read_fw_idx);
}
```

The function test_dev_config_update_u8() is called from both the locked
and the unlocked context, function config_num_requests_store() and
config_read_fw_idx_store() which can both be called asynchronously as
they are driver's methods, while test_dev_config_update_u8() and siblings
change their argument pointed to by u8 *cfg or similar pointer.

To avoid deadlock on test_fw_mutex, the lock is dropped before calling
test_dev_config_update_u8() and re-acquired within test_dev_config_update_u8()
itself, but alas this creates a race condition.

Having two locks wouldn't assure a race-proof mutual exclusion.

This situation is best avoided by the introduction of a new, unlocked
function __test_dev_config_update_u8() which can be called from the locked
context and reducing test_dev_config_update_u8() to:

```
static int test_dev_config_update_u8(const char *buf, size_t size, u8 *cfg)
{
        int ret;

        mutex_lock(&test_fw_mutex);
        ret = __test_dev_config_update_u8(buf, size, cfg);
        mutex_unlock(&test_fw_mutex);

        return ret;
}
```

doing the locking and calling the unlocked primitive, which enables both
locked and unlocked versions without duplication of code.

```
    The similar approach was applied to all functions called from the locked
    and the unlocked context, which safely mitigates both deadlocks and race
    conditions in the driver.

    __test_dev_config_update_bool(), __test_dev_config_update_u8() and
    __test_dev_config_update_size_t() unlocked versions of the functions
    were introduced to be called from the locked contexts as a workaround
    without releasing the main driver's lock and thereof causing a race
    condition.

    The test_dev_config_update_bool(), test_dev_config_update_u8() and
    test_dev_config_update_size_t() locked versions of the functions
    are being called from driver methods without the unnecessary multiplying
    of the locking and unlocking code for each method, and complicating
    the code with saving of the return value across lock.

    Fixes: 7feebfa487b92 ("test_firmware: add support for request_firmware_into_buf")
    Cc: Luis Chamberlain <mcgrof@kernel.org>
    Cc: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Cc: Russ Weight <russell.h.weight@intel.com>
    Cc: Takashi Iwai <tiwai@suse.de>
    Cc: Tianfei Zhang <tianfei.zhang@intel.com>
    Cc: Shuah Khan <shuah@kernel.org>
    Cc: Colin Ian King <colin.i.king@gmail.com>
    Cc: Randy Dunlap <rdunlap@infradead.org>
    Cc: linux-kselftest@vger.kernel.org
    Cc: stable@vger.kernel.org # v5.4
    Suggested-by: Dan Carpenter <error27@gmail.com>
    Signed-off-by: Mirsad Goran Todorovac <mirsad.todorovac@alu.unizg.hr>
    Link: https://lore.kernel.org/r/20230509084746.48259-1-mirsad.todorovac@alu.unizg.hr
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b95a6a3c1aa062d4e3aba9e35da88757434abce4
Author: Maninder Singh <maninder1.s@samsung.com>
Date:   Mon May 29 16:43:37 2023 +0530

    powerpc/xmon: Use KSYM_NAME_LEN in array size

    commit 719dfd5925e186e09a2a6f23016936ac436f3d78 upstream.

    kallsyms_lookup() which in turn calls kallsyms_lookup_buildid() writes
    to index "KSYM_NAME_LEN - 1".

    Thus the array passed as namebuf to kallsyms_lookup() should be
    KSYM_NAME_LEN in size.

    In xmon.c the array was defined to be "128" bytes directly, without
    using KSYM_NAME_LEN. Commit b8a94bfb3395 ("kallsyms: increase maximum
    kernel symbol length to 512") changed the value to 512, but missed
    updating the xmon code.

    Fixes: b8a94bfb3395 ("kallsyms: increase maximum kernel symbol length to 512")
    Cc: stable@vger.kernel.org # v6.1+
    Co-developed-by: Onkarnath <onkarnath.1@samsung.com>
    Signed-off-by: Onkarnath <onkarnath.1@samsung.com>
    Signed-off-by: Maninder Singh <maninder1.s@samsung.com>
    [mpe: Tweak change log wording and fix commit reference]
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Link: https://msgid.link/20230529111337.352990-2-maninder1.s@samsung.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1d50232736aaeacb17bb9b1191e8fe4384ab5b4f
Author: Herve Codina <herve.codina@bootlin.com>
Date:   Tue May 23 10:59:02 2023 +0200
```

```
    serial: cpm_uart: Fix a COMPILE_TEST dependency

    commit 7183c37fd53eee1e795206e625da12a5d7ec1e1a upstream.

    In a COMPILE_TEST configuration, the cpm_uart driver uses symbols from
    the cpm_uart_cpm2.c file. This file is compiled only when CONFIG_CPM2 is
    set.

    Without this dependency, the linker fails with some missing symbols for
    COMPILE_TEST configuration that needs SERIAL_CPM without enabling CPM2.

    This lead to:
      depends on CPM2 || CPM1 || (PPC32 && CPM2 && COMPILE_TEST)

    This dependency does not make sense anymore and can be simplified
    removing all the COMPILE_TEST part.

    Signed-off-by: Herve Codina <herve.codina@bootlin.com>
    Reported-by: kernel test robot <lkp@intel.com>
    Link: https://lore.kernel.org/oe-kbuild-all/202305160221.9XgweObz-lkp@intel.com/
    Fixes: e3e7b13bffae ("serial: allow COMPILE_TEST for some drivers")
    Cc: stable <stable@kernel.org>
    Link: https://lore.kernel.org/r/20230523085902.75837-3-herve.codina@bootlin.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b84462c0c7c873a687aad2025e876f097fe6562e
Author: Christophe JAILLET <christophe.jaillet@wanadoo.fr>
Date:   Sun May 14 13:25:42 2023 +0200

    serial: 8250_tegra: Fix an error handling path in tegra_uart_probe()

    commit 134f49dec0b6aca3259cd8259de4c572048bd207 upstream.

    If an error occurs after reset_control_deassert(), it must be re-asserted,
    as already done in the .remove() function.

    Fixes: c6825c6395b7 ("serial: 8250_tegra: Create Tegra specific 8250 driver")
    Cc: stable <stable@kernel.org>
    Signed-off-by: Christophe JAILLET <christophe.jaillet@wanadoo.fr>
    Link:
https://lore.kernel.org/r/f8130f35339cc80edc6b9aac4bb2a60b60a226bf.1684063511.git.christophe
.jaillet@wanadoo.fr
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ef77d0b42836e4bd15dd309f27083f19499be78a
Author: Helge Deller <deller@gmx.de>
Date:   Sat May 27 08:41:09 2023 +0200

    fbcon: Fix null-ptr-deref in soft_cursor

    commit d78bd6cc68276bd57f766f7cb98bfe32c23ab327 upstream.

    syzbot repored this bug in the softcursor code:

    BUG: KASAN: null-ptr-deref in soft_cursor+0x384/0x6b4
drivers/video/fbdev/core/softcursor.c:70
    Read of size 16 at addr 0000000000000200 by task kworker/u4:1/12

    CPU: 0 PID: 12 Comm: kworker/u4:1 Not tainted 6.4.0-rc3-syzkaller-geb0f1697d729 #0
    Hardware name: Google Google Compute Engine/Google Compute Engine, BIOS Google
04/28/2023
    Workqueue: events_power_efficient fb_flashcursor
    Call trace:
```

```
       dump_backtrace+0x1b8/0x1e4 arch/arm64/kernel/stacktrace.c:233
       show_stack+0x2c/0x44 arch/arm64/kernel/stacktrace.c:240
       __dump_stack lib/dump_stack.c:88 [inline]
       dump_stack_lvl+0xd0/0x124 lib/dump_stack.c:106
       print_report+0xe4/0x514 mm/kasan/report.c:465
       kasan_report+0xd4/0x130 mm/kasan/report.c:572
       kasan_check_range+0x264/0x2a4 mm/kasan/generic.c:187
       __asan_memcpy+0x3c/0x84 mm/kasan/shadow.c:105
       soft_cursor+0x384/0x6b4 drivers/video/fbdev/core/softcursor.c:70
       bit_cursor+0x113c/0x1a64 drivers/video/fbdev/core/bitblit.c:377
       fb_flashcursor+0x35c/0x54c drivers/video/fbdev/core/fbcon.c:380
       process_one_work+0x788/0x12d4 kernel/workqueue.c:2405
       worker_thread+0x8e0/0xfe8 kernel/workqueue.c:2552
       kthread+0x288/0x310 kernel/kthread.c:379
       ret_from_fork+0x10/0x20 arch/arm64/kernel/entry.S:853

    This fix let bit_cursor() bail out early when a font bitmap
    isn't available yet.

    Signed-off-by: Helge Deller <deller@gmx.de>
    Reported-by: syzbot+d910bd780e6efac35869@syzkaller.appspotmail.com
    Acked-by: Sam Ravnborg <sam@ravnborg.org>
    Cc: stable@kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 8664693a8ff29b9f3fa082f6c750078a4c70cb03
Author: Theodore Ts'o <tytso@mit.edu>
Date:   Tue May 23 23:49:51 2023 -0400

    ext4: add lockdep annotations for i_data_sem for ea_inode's

    commit aff3bea95388299eec63440389b4545c8041b357 upstream.

    Treat i_data_sem for ea_inodes as being in their own lockdep class to
    avoid lockdep complaints about ext4_setattr's use of inode_lock() on
    normal inodes potentially causing lock ordering with i_data_sem on
    ea_inodes in ext4_xattr_inode_write().  However, ea_inodes will be
    operated on by ext4_setattr(), so this isn't a problem.

    Cc: stable@kernel.org
    Link: https://syzkaller.appspot.com/bug?extid=298c5d8fb4a128bc27b0
    Reported-by: syzbot+298c5d8fb4a128bc27b0@syzkaller.appspotmail.com
    Signed-off-by: Theodore Ts'o <tytso@mit.edu>
    Link: https://lore.kernel.org/r/20230524034951.779531-5-tytso@mit.edu
    Signed-off-by: Theodore Ts'o <tytso@mit.edu>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 9d993659ed77cd61c61f8da774ef1a7611e7f0e5
Author: Theodore Ts'o <tytso@mit.edu>
Date:   Tue May 23 23:49:50 2023 -0400

    ext4: disallow ea_inodes with extended attributes

    commit 2bc7e7c1a3bc9bd0cbf0f71006f6fe7ef24a00c2 upstream.

    An ea_inode stores the value of an extended attribute; it can not have
    extended attributes itself, or this will cause recursive nightmares.
    Add a check in ext4_iget() to make sure this is the case.

    Cc: stable@kernel.org
    Reported-by: syzbot+e44749b6ba4d0434cd47@syzkaller.appspotmail.com
    Signed-off-by: Theodore Ts'o <tytso@mit.edu>
    Link: https://lore.kernel.org/r/20230524034951.779531-4-tytso@mit.edu
    Signed-off-by: Theodore Ts'o <tytso@mit.edu>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit 982f2501e753410670e65a778c6d997ffd9c8cf2
Author: Theodore Ts'o <tytso@mit.edu>
Date:   Tue May 23 23:49:49 2023 -0400

    ext4: set lockdep subclass for the ea_inode in ext4_xattr_inode_cache_find()

    commit b928dfdcb27d8fa59917b794cfba53052a2f050f upstream.

    If the ea_inode has been pushed out of the inode cache while there is
    still a reference in the mb_cache, the lockdep subclass will not be
    set on the inode, which can lead to some lockdep false positives.

    Fixes: 33d201e0277b ("ext4: fix lockdep warning about recursive inode locking")
    Cc: stable@kernel.org
    Reported-by: syzbot+d4b971e744b1f5439336@syzkaller.appspotmail.com
    Signed-off-by: Theodore Ts'o <tytso@mit.edu>
    Link: https://lore.kernel.org/r/20230524034951.779531-3-tytso@mit.edu
    Signed-off-by: Theodore Ts'o <tytso@mit.edu>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 2f1dace530e86840119f1cb320331e3feb7c3916
Author: Theodore Ts'o <tytso@mit.edu>
Date:   Tue May 23 23:49:48 2023 -0400

    ext4: add EA_INODE checking to ext4_iget()

    commit b3e6bcb94590dea45396b9481e47b809b1be4afa upstream.

    Add a new flag, EXT4_IGET_EA_INODE which indicates whether the inode
    is expected to have the EA_INODE flag or not.  If the flag is not
    set/clear as expected, then fail the iget() operation and mark the
    file system as corrupted.

    This commit also makes the ext4_iget() always perform the
    is_bad_inode() check even when the inode is already inode cache.  This
    allows us to remove the is_bad_inode() check from the callers of
    ext4_iget() in the ea_inode code.

    Reported-by: syzbot+cbb68193bdb95af4340a@syzkaller.appspotmail.com
    Reported-by: syzbot+62120febbd1ee3c3c860@syzkaller.appspotmail.com
    Reported-by: syzbot+edce54daffee36421b4c@syzkaller.appspotmail.com
    Cc: stable@kernel.org
    Signed-off-by: Theodore Ts'o <tytso@mit.edu>
    Link: https://lore.kernel.org/r/20230524034951.779531-2-tytso@mit.edu
    Signed-off-by: Theodore Ts'o <tytso@mit.edu>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 4d4479bfe0b2986d8d23eeb60614418dbcadd504
Author: Paolo Abeni <pabeni@redhat.com>
Date:   Wed May 31 12:37:08 2023 -0700

    mptcp: fix active subflow finalization

    commit 55b47ca7d80814ceb63d64e032e96cd6777811e5 upstream.

    Active subflow are inserted into the connection list at creation time.
    When the MPJ handshake completes successfully, a new subflow creation
    netlink event is generated correctly, but the current code wrongly
    avoid initializing a couple of subflow data.

    The above will cause misbehavior on a few exceptional events: unneeded
    mptcp-level retransmission on msk-level sequence wrap-around and infinite
    mapping fallback even when a MPJ socket is present.
```

```
    Address the issue factoring out the needed initialization in a new helper
    and invoking the latter from __mptcp_finish_join() time for passive
    subflow and from mptcp_finish_join() for active ones.

    Fixes: 0530020a7c8f ("mptcp: track and update contiguous data status")
    Cc: stable@vger.kernel.org
    Reviewed-by: Mat Martineau <martineau@kernel.org>
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Mat Martineau <martineau@kernel.org>
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 80fdab6e0af92815944b31831d4c22b22b415d79
Author: Paolo Abeni <pabeni@redhat.com>
Date:   Wed May 31 12:37:03 2023 -0700

    mptcp: fix connect timeout handling

    commit 786fc12457268cc9b555dde6c22ae7300d4b40e1 upstream.

    Ondrej reported a functional issue WRT timeout handling on connect
    with a nice reproducer.

    The problem is that the current mptcp connect waits for both the
    MPTCP socket level timeout, and the first subflow socket timeout.
    The latter is not influenced/touched by the exposed setsockopt().

    Overall the above makes the SO_SNDTIMEO a no-op on connect.

    Since mptcp_connect is invoked via inet_stream_connect and the
    latter properly handle the MPTCP level timeout, we can address the
    issue making the nested subflow level connect always unblocking.

    This also allow simplifying a bit the code, dropping an ugly hack
    to handle the fastopen and custom proto_ops connect.

    The issues predates the blamed commit below, but the current resolution
    requires the infrastructure introduced there.

    Fixes: 54f1944ed6d2 ("mptcp: factor out mptcp_connect()")
    Reported-by: Ondrej Mosnacek <omosnace@redhat.com>
    Closes: https://github.com/multipath-tcp/mptcp_net-next/issues/399
    Cc: stable@vger.kernel.org
    Reviewed-by: Mat Martineau <martineau@kernel.org>
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Mat Martineau <martineau@kernel.org>
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 25dcf9962755b8e5f40db026ef0310efec250b32
Author: Matthieu Baerts <matthieu.baerts@tessares.net>
Date:   Sun May 28 19:35:33 2023 +0200

    selftests: mptcp: userspace pm: skip if MPTCP is not supported

    commit 63212608a92a1ff10ae56dbb14e9fb685f7e4ffa upstream.

    Selftests are supposed to run on any kernels, including the old ones not
    supporting MPTCP.

    A new check is then added to make sure MPTCP is supported. If not, the
    test stops and is marked as "skipped".

    Link: https://github.com/multipath-tcp/mptcp_net-next/issues/368
    Fixes: 259a834fadda ("selftests: mptcp: functional tests for the userspace PM type")
```

```
    Cc: stable@vger.kernel.org
    Acked-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Matthieu Baerts <matthieu.baerts@tessares.net>
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 05accac4ca39819bb486608c7365f1102d9a2297
Author: Matthieu Baerts <matthieu.baerts@tessares.net>
Date:   Sun May 28 19:35:32 2023 +0200

    selftests: mptcp: sockopt: skip if MPTCP is not supported

    commit cf6f0fda7af7e8e016070bfee6b189e671a0c776 upstream.

    Selftests are supposed to run on any kernels, including the old ones not
    supporting MPTCP.

    A new check is then added to make sure MPTCP is supported. If not, the
    test stops and is marked as "skipped".

    Link: https://github.com/multipath-tcp/mptcp_net-next/issues/368
    Fixes: dc65fe82fb07 ("selftests: mptcp: add packet mark test case")
    Cc: stable@vger.kernel.org
    Acked-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Matthieu Baerts <matthieu.baerts@tessares.net>
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 65de584c14242257bf5cf0cfb69c7332dd970d3a
Author: Matthieu Baerts <matthieu.baerts@tessares.net>
Date:   Sun May 28 19:35:31 2023 +0200

    selftests: mptcp: simult flows: skip if MPTCP is not supported

    commit 9161f21c74a1a0e7bb39eb84ea0c86b23c92fc87 upstream.

    Selftests are supposed to run on any kernels, including the old ones not
    supporting MPTCP.

    A new check is then added to make sure MPTCP is supported. If not, the
    test stops and is marked as "skipped".

    Link: https://github.com/multipath-tcp/mptcp_net-next/issues/368
    Fixes: 1a418cb8e888 ("mptcp: simult flow self-tests")
    Cc: stable@vger.kernel.org
    Acked-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Matthieu Baerts <matthieu.baerts@tessares.net>
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a5126f8bb34f69f9f371c4efcb61af30b517e1c6
Author: Matthieu Baerts <matthieu.baerts@tessares.net>
Date:   Sun May 28 19:35:30 2023 +0200

    selftests: mptcp: diag: skip if MPTCP is not supported

    commit 46565acdd29facbf418a11e4a3791b3c8967308d upstream.

    Selftests are supposed to run on any kernels, including the old ones not
    supporting MPTCP.

    A new check is then added to make sure MPTCP is supported. If not, the
    test stops and is marked as "skipped".

    Link: https://github.com/multipath-tcp/mptcp_net-next/issues/368
```

```
    Fixes: df62f2ec3df6 ("selftests/mptcp: add diag interface tests")
    Cc: stable@vger.kernel.org
    Acked-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Matthieu Baerts <matthieu.baerts@tessares.net>
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 02336f170a15ae5e1ab5bcfd70d085048ee29722
Author: Matthieu Baerts <matthieu.baerts@tessares.net>
Date:   Sun May 28 19:35:26 2023 +0200

    selftests: mptcp: join: avoid using 'cmp --bytes'

    commit d328fe87067480cf2bd0b58dab428a98d31dbb7e upstream.

    BusyBox's 'cmp' command doesn't support the '--bytes' parameter.

    Some CIs -- i.e. LKFT -- use BusyBox and have the mptcp_join.sh test
    failing [1] because their 'cmp' command doesn't support this '--bytes'
    option:

        cmp: unrecognized option '--bytes=1024'
        BusyBox v1.35.0 () multi-call binary.

        Usage: cmp [-ls] [-n NUM] FILE1 [FILE2]

    Instead, 'head --bytes' can be used as this option is supported by
    BusyBox. A temporary file is needed for this operation.

    Because it is apparently quite common to use BusyBox, it is certainly
    better to backport this fix to impacted kernels.

    Fixes: 6bf41020b72b ("selftests: mptcp: update and extend fastclose test-cases")
    Cc: stable@vger.kernel.org
    Link: https://qa-reports.linaro.org/lkft/linux-mainline-master/build/v6.3-rc5-5-
g148341f0a2f5/testrun/16088933/suite/kselftest-net-mptcp/test/net_mptcp_userspace_pm_sh/log
[1]
    Suggested-by: Paolo Abeni <pabeni@redhat.com>
    Reviewed-by: Mat Martineau <martineau@kernel.org>
    Signed-off-by: Matthieu Baerts <matthieu.baerts@tessares.net>
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit fd52ef6f25e9fe2a2c6ed7ae7811a08d9491c5d7
Author: Matthieu Baerts <matthieu.baerts@tessares.net>
Date:   Sun May 28 19:35:29 2023 +0200

    selftests: mptcp: join: skip if MPTCP is not supported

    commit 715c78a82e00f848f99ef76e6f6b89216ccba268 upstream.

    Selftests are supposed to run on any kernels, including the old ones not
    supporting MPTCP.

    A new check is then added to make sure MPTCP is supported. If not, the
    test stops and is marked as "skipped".

    Link: https://github.com/multipath-tcp/mptcp_net-next/issues/368
    Fixes: b08fbf241064 ("selftests: add test-cases for MPTCP MP_JOIN")
    Cc: stable@vger.kernel.org
    Acked-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Matthieu Baerts <matthieu.baerts@tessares.net>
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit 2e5d56884f96bbc7421d61e31996b00820c66822
Author: Matthieu Baerts <matthieu.baerts@tessares.net>
Date:   Sun May 28 19:35:28 2023 +0200

    selftests: mptcp: pm nl: skip if MPTCP is not supported

    commit 0f4955a40dafe18a1122e3714d8173e4b018e869 upstream.

    Selftests are supposed to run on any kernels, including the old ones not
    supporting MPTCP.

    A new check is then added to make sure MPTCP is supported. If not, the
    test stops and is marked as "skipped".

    Link: https://github.com/multipath-tcp/mptcp_net-next/issues/368
    Fixes: eedbc685321b ("selftests: add PM netlink functional tests")
    Cc: stable@vger.kernel.org
    Acked-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Matthieu Baerts <matthieu.baerts@tessares.net>
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1ef290e48739a763beabb1f417211e69389f6b34
Author: Matthieu Baerts <matthieu.baerts@tessares.net>
Date:   Sun May 28 19:35:27 2023 +0200

    selftests: mptcp: connect: skip if MPTCP is not supported

    commit d83013bdf90a7994a474b0e650a7fc94b0d4ded6 upstream.

    Selftests are supposed to run on any kernels, including the old ones not
    supporting MPTCP.

    A new check is then added to make sure MPTCP is supported. If not, the
    test stops and is marked as "skipped". Note that this check can also
    mark the test as failed if 'SELFTESTS_MPTCP_LIB_EXPECT_ALL_FEATURES' env
    var is set to 1: by doing that, we can make sure a test is not being
    skipped by mistake.

    A new shared file is added here to be able to re-used the same check in
    the different selftests we have.

    Link: https://github.com/multipath-tcp/mptcp_net-next/issues/368
    Fixes: 048d19d444be ("mptcp: add basic kselftest for mptcp")
    Cc: stable@vger.kernel.org
    Acked-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Matthieu Baerts <matthieu.baerts@tessares.net>
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit db7a1b5e8b13326c98c9da340ae67c3d92092e1f
Author: Pietro Borrello <borrello@diag.uniroma1.it>
Date:   Sat Jan 28 16:23:41 2023 +0000

    tracing/probe: trace_probe_primary_from_call(): checked list_first_entry

    commit 81d0fa4cb4fc0e1a49c2b22f92c43d9fe972ebcf upstream.

    All callers of trace_probe_primary_from_call() check the return
    value to be non NULL. However, the function returns
    list_first_entry(&tpe->probes, ...) which can never be NULL.
    Additionally, it does not check for the list being possibly empty,
    possibly causing a type confusion on empty lists.
    Use list_first_entry_or_null() which solves both problems.
```

```
    Link: https://lore.kernel.org/linux-trace-kernel/20230128-list-entry-null-check-v1-1-
8bde6a3da2ef@diag.uniroma1.it/

    Fixes: 60d53e2c3b75 ("tracing/probe: Split trace_event related data from trace_probe")
    Signed-off-by: Pietro Borrello <borrello@diag.uniroma1.it>
    Reviewed-by: Steven Rostedt (Google) <rostedt@goodmis.org>
    Acked-by: Masami Hiramatsu (Google) <mhiramat@kernel.org>
    Acked-by: Mukesh Ojha <quic_mojha@quicinc.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Masami Hiramatsu (Google) <mhiramat@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 8d505d06d7330f5d67d3e5e9e1c647fb0b10ddad
Author: Steven Rostedt (Google) <rostedt@goodmis.org>
Date:   Tue May 23 22:11:08 2023 -0400

    tracing/histograms: Allow variables to have some modifiers

    commit e30fbc618e97b38dbb49f1d44dcd0778d3f23b8c upstream.

    Modifiers are used to change the behavior of keys. For instance, they
    can grouped into buckets, converted to syscall names (from the syscall
    identifier), show task->comm of the current pid, be an array of longs
    that represent a stacktrace, and more.

    It was found that nothing stopped a value from taking a modifier. As
    values are simple counters. If this happened, it would call code that
    was not expecting a modifier and crash the kernel. This was fixed by
    having the ___create_val_field() function test if a modifier was present
    and fail if one was. This fixed the crash.

    Now there's a problem with variables. Variables are used to pass fields
    from one event to another. Variables are allowed to have some modifiers,
    as the processing may need to happen at the time of the event (like
    stacktraces and comm names of the current pid). The issue is that it too
    uses __create_val_field(). Now that fails on modifiers, variables can no
    longer use them (this is a regression).

    As not all modifiers are for variables, have them use a separate check.

    Link: https://lore.kernel.org/linux-trace-
kernel/20230523221108.064a5d82@rorschach.local.home

    Cc: stable@vger.kernel.org
    Cc: Masami Hiramatsu <mhiramat@kernel.org>
    Cc: Tom Zanussi <zanussi@kernel.org>
    Cc: Mark Rutland <mark.rutland@arm.com>
    Fixes: e0213434fe3e4 ("tracing: Do not let histogram values have some modifiers")
    Signed-off-by: Steven Rostedt (Google) <rostedt@goodmis.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 7bd6b2d4b64d86ca1e838b812a8fb2d37f3e6f98
Author: Daniel Bristot de Oliveira <bristot@kernel.org>
Date:   Thu May 11 18:32:01 2023 +0200

    tracing/timerlat: Always wakeup the timerlat thread

    commit 632478a05821bc1c9b55c3a1dd0fb1be7bfa1acc upstream.

    While testing rtla timerlat auto analysis, I reach a condition where
    the interface was not receiving tracing data. I was able to manually
    reproduce the problem with these steps:

      # echo 0 > tracing_on                # disable trace
      # echo 1 > osnoise/stop_tracing_us   # stop trace if timerlat irq > 1 us
```

```
       # echo timerlat > current_tracer      # enable timerlat tracer
       # sleep 1                             # wait... that is the time when rtla
                                             # apply configs like prio or cgroup
       # echo 1 > tracing_on                 # start tracing
       # cat trace
       # tracer: timerlat
       #
       #                                _-----=> irqs-off
       #                               / _----=> need-resched
       #                              | / _---=> hardirq/softirq
       #                              || / _--=> preempt-depth
       #                              ||| / _-=> migrate-disable
       #                              |||| /     delay
       #                              |||||            ACTIVATION
       #           TASK-PID      CPU# |||||   TIMESTAMP   ID            CONTEXT
  LATENCY
       #              | |         |   |||||      |         |                  |
  |
              NOTHING!
```

    Then, trying to enable tracing again with echo 1 > tracing_on resulted
    in no change: the trace was still not tracing.

    This problem happens because the timerlat IRQ hits the stop tracing
    condition while tracing is off, and do not wake up the timerlat thread,
    so the timerlat threads are kept sleeping forever, resulting in no
    trace, even after re-enabling the tracer.

    Avoid this condition by always waking up the threads, even after stopping
    tracing, allowing the tracer to return to its normal operating after
    a new tracing on.

    Link: https://lore.kernel.org/linux-trace-
  kernel/1ed8f830638b20a39d535d27d908e319a9a3c4e2.1683822622.git.bristot@kernel.org

    Cc: Juri Lelli <juri.lelli@redhat.com>
    Cc: stable@vger.kernel.org
    Fixes: a955d7eac177 ("trace: Add timerlat tracer")
    Signed-off-by: Daniel Bristot de Oliveira <bristot@kernel.org>
    Signed-off-by: Steven Rostedt (Google) <rostedt@goodmis.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

  commit 51f49155b28dae32c23bc49892e1b16a4459377c
  Author: Arnd Bergmann <arnd@arndb.de>
  Date: Mon Apr 17 22:56:50 2023 +0200

    mtdchar: mark bits of ioctl handler noinline

    commit 0ea923f443350c8c5cca6eef5b748d52b903f46c upstream.

    The addition of the mtdchar_read_ioctl() function caused the stack usage
    of mtdchar_ioctl() to grow beyond the warning limit on 32-bit architectures
    with gcc-13:

    drivers/mtd/mtdchar.c: In function 'mtdchar_ioctl':
    drivers/mtd/mtdchar.c:1229:1: error: the frame size of 1488 bytes is larger than 1024
  bytes [-Werror=frame-larger-than=]

    Mark both the read and write portions as noinline_for_stack to ensure
    they don't get inlined and use separate stack slots to reduce the
    maximum usage, both in the mtdchar_ioctl() and combined with any
    of its callees.

    Fixes: 095bb6e44eb1 ("mtdchar: add MEMREAD ioctl")
    Cc: stable@vger.kernel.org

```
    Signed-off-by: Arnd Bergmann <arnd@arndb.de>
    Reviewed-by: Richard Weinberger <richard@nod.at>
    Signed-off-by: Miquel Raynal <miquel.raynal@bootlin.com>
    Link: https://lore.kernel.org/linux-mtd/20230417205654.1982368-1-arnd@kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 7346f5c04ffb912ea69184a2a20999aae0740664
Author: Paul Moore <paul@paul-moore.com>
Date:   Thu Jun 1 10:21:21 2023 -0400

    selinux: don't use make's grouped targets feature yet

    commit 42c4e97e06a839b07d834f640a10911ad84ec8b3 upstream.

    The Linux Kernel currently only requires make v3.82 while the grouped
    target functionality requires make v4.3.  Removed the grouped target
    introduced in 4ce1f694eb5d ("selinux: ensure av_permissions.h is
    built when needed") as well as the multiple header file targets in
    the make rule.  This effectively reverts the problem commit.

    We will revisit this change when make >= 4.3 is required by the rest
    of the kernel.

    Cc: stable@vger.kernel.org
    Fixes: 4ce1f694eb5d ("selinux: ensure av_permissions.h is built when needed")
    Reported-by: Erwan Velu <e.velu@criteo.com>
    Reported-by: Luiz Capitulino <luizcap@amazon.com>
    Tested-by: Luiz Capitulino <luizcap@amazon.com>
    Signed-off-by: Paul Moore <paul@paul-moore.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit aa7af57765f155b9864c0ef31300978b43a4b112
Author: Ben Noordhuis <info@bnoordhuis.nl>
Date:   Sat May 6 11:55:02 2023 +0200

    io_uring: undeprecate epoll_ctl support

    commit 4ea0bf4b98d66a7a790abb285539f395596bae92 upstream.

    Libuv recently started using it so there is at least one consumer now.

    Cc: stable@vger.kernel.org
    Fixes: 61a2732af4b0 ("io_uring: deprecate epoll_ctl support")
    Link: https://github.com/libuv/libuv/pull/3979
    Signed-off-by: Ben Noordhuis <info@bnoordhuis.nl>
    Link: https://lore.kernel.org/r/20230506095502.13401-1-info@bnoordhuis.nl
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 438f8c93a7e5aa0f94907423ffaa5fe9438536bc
Author: Ism Hong <ism.hong@gmail.com>
Date:   Thu Jun 1 17:53:55 2023 +0800

    riscv: perf: Fix callchain parse error with kernel tracepoint events

    commit 9a7e8ec0d4cc64870ea449b4fce5779b77496cbb upstream.

    For RISC-V, when tracing with tracepoint events, the IP and status are
    set to 0, preventing the perf code parsing the callchain and resolving
    the symbols correctly.

     ./ply 'tracepoint:kmem/kmem_cache_alloc { @[stack]=count(); }'
     @:
     { <STACKID4294967282> }: 1
```

```
    The fix is to implement perf_arch_fetch_caller_regs for riscv, which
    fills several necessary registers used for callchain unwinding,
    including epc, sp, s0 and status. It's similar to commit b3eac0265bf6
    ("arm: perf: Fix callchain parse error with kernel tracepoint events")
    and commit 5b09a094f2fb ("arm64: perf: Fix callchain parse error with
    kernel tracepoint events").

    With this patch, callchain can be parsed correctly as:

     ./ply 'tracepoint:kmem/kmem_cache_alloc { @[stack]=count(); }'
     @:
     {
             __traceiter_kmem_cache_alloc+68
             __traceiter_kmem_cache_alloc+68
             kmem_cache_alloc+354
             __sigqueue_alloc+94
             __send_signal_locked+646
             send_signal_locked+154
             do_send_sig_info+84
             __kill_pgrp_info+130
             kill_pgrp+60
             isig+150
             n_tty_receive_signal_char+36
             n_tty_receive_buf_standard+2214
             n_tty_receive_buf_common+280
             n_tty_receive_buf2+26
             tty_ldisc_receive_buf+34
             tty_port_default_receive_buf+62
             flush_to_ldisc+158
             process_one_work+458
             worker_thread+138
             kthread+178
             riscv_cpufeature_patch_func+832
      }: 1

    Signed-off-by: Ism Hong <ism.hong@gmail.com>
    Link: https://lore.kernel.org/r/20230601095355.1168910-1-ism.hong@gmail.com
    Fixes: 178e9fc47aae ("perf: riscv: preliminary RISC-V support")
    Cc: stable@vger.kernel.org
    Signed-off-by: Palmer Dabbelt <palmer@rivosinc.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 59219fca4176e9ca249bde4fb37e23dbd3375c08
Author: Peter Rosin <peda@axentia.se>
Date:   Tue May 23 19:20:47 2023 +0200

    dmaengine: at_hdmac: Extend the Flow Controller bitfield to three bits

    commit e14fd2af7a1d621c167dad761f729135a7a76ff4 upstream.

    Some chips have two bits (e.g SAMA5D3), and some have three (e.g.
    SAM9G45). A field width of three is compatible as long as valid
    values are used for the different chips.

    There is no current use of any value needing three bits, so the
    fixed bug is relatively benign.

    Fixes: d8840a7edcf0 ("dmaengine: at_hdmac: Use bitfield access macros")
    Cc: stable@vger.kernel.org
    Reviewed-by: Tudor Ambarus <tudor.ambarus@linaro.org>
    Signed-off-by: Peter Rosin <peda@axentia.se>
    Link: https://lore.kernel.org/r/e2c898ba-c3a3-5dd3-384b-0585661c79f2@axentia.se
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit ef863e6bda5e3f4e5b4712007a0ccb59e8beb757
Author: Peter Rosin <peda@axentia.se>
Date:   Tue May 23 19:20:37 2023 +0200

    dmaengine: at_hdmac: Repair bitfield macros for peripheral ID handling

    commit 2a6c7e8cc74e58ba94b8c897035a8ef7f7349f76 upstream.

    The MSB part of the peripheral IDs need to go into the ATC_SRC_PER_MSB
    and ATC_DST_PER_MSB fields. Not the LSB part.

    This fixes a severe regression for TSE-850 devices (compatible
    axentia,tse850v3) where output to the audio I2S codec (the main
    purpose of the device) simply do not work.

    Fixes: d8840a7edcf0 ("dmaengine: at_hdmac: Use bitfield access macros")
    Cc: stable@vger.kernel.org
    Signed-off-by: Peter Rosin <peda@axentia.se>
    Reviewed-by: Tudor Ambarus <tudor.ambarus@linaro.org>
    Link: https://lore.kernel.org/r/01e5dae1-d4b0-cf31-516b-423b11b077f1@axentia.se
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a494435f05c289b4a675ec6ca2c876f3d1e52584
Author: Benjamin Tissoires <benjamin.tissoires@redhat.com>
Date:   Mon Jun 5 18:56:36 2023 +0200

    HID: hidpp: terminate retry loop on success

    commit 7c28afd5512e371773dbb2bf95a31ed5625651d9 upstream.

    It seems we forgot the normal case to terminate the retry loop,
    making us asking 3 times each command, which is probably a little bit
    too much.

    And remove the ugly "goto exit" that can be replaced by a simpler "break"

    Fixes: 586e8fede795 ("HID: logitech-hidpp: Retry commands when device is busy")
    Suggested-by: Mark Lord <mlord@pobox.com>
    Tested-by: Mark Lord <mlord@pobox.com>
    Signed-off-by: Benjamin Tissoires <benjamin.tissoires@redhat.com>
    Signed-off-by: Jiri Kosina <jkosina@suse.cz>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1d454bcf877ea17be5ad695967a9e722a6112eb8
Author: Sasha Levin <sashal@kernel.org>
Date:   Tue Jun 6 10:37:40 2023 -0400

    btrfs: call btrfs_orig_bbio_end_io in btrfs_end_bio_work

    [ Upstream commit 45c2f36871955b51b4ce083c447388d8c72d6b91 ]

    When I implemented the storage layer bio splitting, I was under the
    assumption that we'll never split metadata bios.  But Qu reminded me that
    this can actually happen with very old file systems with unaligned
    metadata chunks and RAID0.

    I still haven't seen such a case in practice, but we better handled this
    case, especially as it is fairly easily to do not calling the ->end_io
    method directly in btrfs_end_io_work, and using the proper
    btrfs_orig_bbio_end_io helper instead.

    In addition to the old file system with unaligned metadata chunks case
    documented in the commit log, the combination of the new scrub code
    with Johannes pending raid-stripe-tree also triggers this case.  We
```

```
    spent some time debugging it and found that this patch solves
    the problem.

    Fixes: 103c19723c80 ("btrfs: split the bio submission path into a separate file")
    CC: stable@vger.kernel.org # 6.3+
    Reviewed-by: Johannes Thumshirn <johannes.thumshirn@wdc.com>
    Tested-by: Johannes Thumshirn <johannes.thumshirn@wdc.com>
    Signed-off-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 772235bb9c5996eacd5fedc26d0ad5c09017c834
Author: Ekansh Gupta <quic_ekangupt@quicinc.com>
Date:   Tue May 23 16:25:47 2023 +0100

    misc: fastrpc: Pass proper scm arguments for secure map request

    [ Upstream commit a6e766dea0a22918735176e4af862d535962f11e ]

    If a map request is made with securemap attribute, the memory
    ownership needs to be reassigned to new VMID to allow access
    from protection domain. Currently only DSP VMID is passed to
    the reassign call which is incorrect as only a combination of
    HLOS and DSP VMID is allowed for memory ownership reassignment
    and passing only DSP VMID will cause assign call failure.

    Also pass proper restoring permissions to HLOS as the source
    permission will now carry both HLOS and DSP VMID permission.

    Change is also made to get valid physical address from
    scatter/gather for this allocation request.

    Fixes: e90d91190619 ("misc: fastrpc: Add support to secure memory map")
    Cc: stable <stable@kernel.org>
    Tested-by: Ekansh Gupta <quic_ekangupt@quicinc.com>
    Signed-off-by: Ekansh Gupta <quic_ekangupt@quicinc.com>
    Signed-off-by: Srinivas Kandagatla <srinivas.kandagatla@linaro.org>
    Link: https://lore.kernel.org/r/20230523152550.438363-2-srinivas.kandagatla@linaro.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6b952a4b2420faa7eccd5c246e767d21349bcd8a
Author: Elliot Berman <quic_eberman@quicinc.com>
Date:   Mon Feb 13 10:18:29 2023 -0800

    firmware: qcom_scm: Use fixed width src vm bitmap

    [ Upstream commit 968a26a07f75377afbd4f7bb18ef587a1443c244 ]

    The maximum VMID for assign_mem is 63. Use a u64 to represent this
    bitmap instead of architecture-dependent "unsigned int" which varies in
    size on 32-bit and 64-bit platforms.

    Acked-by: Kalle Valo <kvalo@kernel.org> (ath10k)
    Tested-by: Gokul krishna Krishnakumar <quic_gokukris@quicinc.com>
    Signed-off-by: Elliot Berman <quic_eberman@quicinc.com>
    Reviewed-by: Bjorn Andersson <andersson@kernel.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230213181832.3489174-1-quic_eberman@quicinc.com
    Stable-dep-of: a6e766dea0a2 ("misc: fastrpc: Pass proper scm arguments for secure map
request")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a5187ae55c6435a91dd20083b0352863606e4e14
Author: Lucas De Marchi <lucas.demarchi@intel.com>
```

```
Date:   Thu Jun 1 14:23:31 2023 -0700

    module/decompress: Fix error checking on zstd decompression

    commit fadb74f9f2f609238070c7ca1b04933dc9400e4a upstream.

    While implementing support for in-kernel decompression in kmod,
    finit_module() was returning a very suspicious value:

            finit_module(3, "", MODULE_INIT_COMPRESSED_FILE) = 18446744072717407296

    It turns out the check for module_get_next_page() failing is wrong,
    and hence the decompression was not really taking place. Invert
    the condition to fix it.

    Fixes: 169a58ad824d ("module/decompress: Support zstd in-kernel decompression")
    Cc: stable@kernel.org
    Cc: Luis Chamberlain <mcgrof@kernel.org>
    Cc: Dmitry Torokhov <dmitry.torokhov@gmail.com>
    Cc: Stephen Boyd <swboyd@chromium.org>
    Signed-off-by: Lucas De Marchi <lucas.demarchi@intel.com>
    Signed-off-by: Luis Chamberlain <mcgrof@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b3b340ae1697cd594fdfcdbdd81094b098e2d98b
Author: Lino Sanfilippo <l.sanfilippo@kunbus.com>
Date:   Tue May 30 18:41:16 2023 +0200

    tpm, tpm_tis: correct tpm_tis_flags enumeration values

    commit 4ecd704a4c51fd95973fcc3a60444e0e24eb9439 upstream.

    With commit 858e8b792d06 ("tpm, tpm_tis: Avoid cache incoherency in test
    for interrupts") bit accessor functions are used to access flags in
    tpm_tis_data->flags.

    However these functions expect bit numbers, while the flags are defined
    as bit masks in enum tpm_tis_flag.

    Fix this inconsistency by using numbers instead of masks also for the
    flags in the enum.

    Reported-by: Pavel Machek <pavel@denx.de>
    Fixes: 858e8b792d06 ("tpm, tpm_tis: Avoid cache incoherency in test for interrupts")
    Signed-off-by: Lino Sanfilippo <l.sanfilippo@kunbus.com>
    Cc: stable@vger.kernel.org
    Reviewed-by: Pavel Machek <pavel@denx.de>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit d24943e27c1683e41f11586d703bff4d02c309a2
Author: Jon Pan-Doh <pandoh@google.com>
Date:   Wed Apr 26 13:32:56 2023 -0700

    iommu/amd: Fix domain flush size when syncing iotlb

    commit 2212fc2acf3f6ee690ea36506fb882a19d1bfcab upstream.

    When running on an AMD vIOMMU, we observed multiple invalidations (of
    decreasing power of 2 aligned sizes) when unmapping a single page.

    Domain flush takes gather bounds (end-start) as size param. However,
    gather->end is defined as the last inclusive address (start + size - 1).
    This leads to an off by 1 error.
```

```
    With this patch, verified that 1 invalidation occurs when unmapping a
    single page.

    Fixes: a270be1b3fdf ("iommu/amd: Use only natural aligned flushes in a VM")
    Cc: stable@vger.kernel.org # >= 5.15
    Signed-off-by: Jon Pan-Doh <pandoh@google.com>
    Tested-by: Sudheer Dantuluri <dantuluris@google.com>
    Suggested-by: Gary Zibrat <gzibrat@google.com>
    Reviewed-by: Vasant Hegde <vasant.hegde@amd.com>
    Acked-by: Nadav Amit <namit@vmware.com>
    Link: https://lore.kernel.org/r/20230426203256.237116-1-pandoh@google.com
    Signed-off-by: Joerg Roedel <jroedel@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit eecd757d1931337f3e9e048e27fef7b45bb4967d
Author: Gaurav Batra <gbatra@linux.vnet.ibm.com>
Date:   Thu May 25 09:34:54 2023 -0500

    powerpc/iommu: Limit number of TCEs to 512 for H_STUFF_TCE hcall

    commit 9d2ccf00bddc268045e3d65a8108d61ada0e4b4e upstream.

    Currently in tce_freemulti_pSeriesLP() there is no limit on how many
    TCEs are passed to the H_STUFF_TCE hcall. This has not caused an issue
    until now, but newer firmware releases have started enforcing a limit of
    512 TCEs per call.

    The limit is correct per the specification (PAPR v2.12 § 14.5.4.2.3).

    The code has been in it's current form since it was initially merged.

    Cc: stable@vger.kernel.org
    Signed-off-by: Gaurav Batra <gbatra@linux.vnet.ibm.com>
    Reviewed-by: Brian King <brking@linux.vnet.ibm.com>
    [mpe: Tweak change log wording & add PAPR reference]
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Link: https://msgid.link/20230525143454.56878-1-gbatra@linux.vnet.ibm.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ea42f1800b0773b04ab0b04433dd0f38aa6f4135
Author: Damien Le Moal <dlemoal@kernel.org>
Date:   Mon May 29 16:32:37 2023 +0900

    block: fix revalidate performance regression

    commit 47fe1c3064c6bc1bfa3c032ff78e603e5dd6e5bc upstream.

    The scsi driver function sd_read_block_characteristics() always calls
    disk_set_zoned() to a disk zoned model correctly, in case the device
    model changed. This is done even for regular disks to set the zoned
    model to BLK_ZONED_NONE and free any zone related resources if the drive
    previously was zoned.

    This behavior significantly impact the time it takes to revalidate disks
    on a large system as the call to disk_clear_zone_settings() done from
    disk_set_zoned() for the BLK_ZONED_NONE case results in the device
    request queued to be frozen, even if there are no zone resources to
    free.

    Avoid this overhead for non-zoned devices by not calling
    disk_clear_zone_settings() in disk_set_zoned() if the device model
    was already set to BLK_ZONED_NONE, which is always the case for regular
    devices.

    Reported by: Brian Bunker <brian@purestorage.com>
```

```
    Fixes: 508aebb80527 ("block: introduce blk_queue_clear_zone_settings()")
    Cc: stable@vger.kernel.org
    Signed-off-by: Damien Le Moal <dlemoal@kernel.org>
    Reviewed-by: Ming Lei <ming.lei@redhat.com>
    Link: https://lore.kernel.org/r/20230529073237.1339862-1-dlemoal@kernel.org
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 267d486f8fd690ef7381f022200295cf7a5b680e
Author: Johan Hovold <johan+linaro@kernel.org>
Date:   Tue May 2 12:38:10 2023 +0200

    phy: qcom-qmp-pcie-msm8996: fix init-count imbalance

    commit e42f110700ed7293700c26145e1ed07ea05ac3f6 upstream.

    The init counter is not decremented on initialisation errors, which
    prevents retrying initialisation.

    Add the missing decrement on initialisation errors so that the counter
    reflects the state of the device.

    Fixes: e78f3d15e115 ("phy: qcom-qmp: new qmp phy driver for qcom-chipsets")
    Cc: stable@vger.kernel.org      # 4.12
    Signed-off-by: Johan Hovold <johan+linaro@kernel.org>
    Reviewed-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Link: https://lore.kernel.org/r/20230502103810.12061-3-johan+linaro@kernel.org
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 2323102033878a4fc0847303e124cee3be25f530
Author: Johan Hovold <johan+linaro@kernel.org>
Date:   Tue May 2 12:38:09 2023 +0200

    phy: qcom-qmp-combo: fix init-count imbalance

    commit 9bf03a0cbd80a256bc1e1c4bcc80bc2b06b8b2b9 upstream.

    The init counter is not decremented on initialisation errors, which
    prevents retrying initialisation and can lead to the runtime suspend
    callback attempting to disable resources that have never been enabled.

    Add the missing decrement on initialisation errors so that the counter
    reflects the state of the device.

    Fixes: e78f3d15e115 ("phy: qcom-qmp: new qmp phy driver for qcom-chipsets")
    Cc: stable@vger.kernel.org      # 4.12
    Signed-off-by: Johan Hovold <johan+linaro@kernel.org>
    Reviewed-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Link: https://lore.kernel.org/r/20230502103810.12061-2-johan+linaro@kernel.org
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit cd001ac15114a35004c006ab41ea7da6a94252bf
Author: pengfuyuan <pengfuyuan@kylinos.cn>
Date:   Tue May 23 15:09:55 2023 +0800

    btrfs: fix csum_tree_block page iteration to avoid tripping on -Werror=array-bounds

    commit 5ad9b4719fc9bc4715c7e19875a962095b0577e7 upstream.

    When compiling on a MIPS 64-bit machine we get these warnings:

        In file included from ./arch/mips/include/asm/cacheflush.h:13,
```

```
                         from ./include/linux/cacheflush.h:5,
                         from ./include/linux/highmem.h:8,
                         from ./include/linux/bvec.h:10,
                         from ./include/linux/blk_types.h:10,
                         from ./include/linux/blkdev.h:9,
                         from fs/btrfs/disk-io.c:7:
        fs/btrfs/disk-io.c: In function ‘csum_tree_block’:
        fs/btrfs/disk-io.c:100:34: error: array subscript 1 is above array bounds of ‘struct
page *[1]’ [-Werror=array-bounds]
          100 |   kaddr = page_address(buf->pages[i]);
              |                        ~~~~~~~~~~^~~
        ./include/linux/mm.h:2135:48: note: in definition of macro ‘page_address’
         2135 | #define page_address(page) lowmem_page_address(page)
              |                                                ^~~~
        cc1: all warnings being treated as errors
```

    We can check if i overflows to solve the problem. However, this doesn't make
    much sense, since i == 1 and num_pages == 1 doesn't execute the body of the loop.
    In addition, i < num_pages can also ensure that buf->pages[i] will not cross
    the boundary. Unfortunately, this doesn't help with the problem observed here:
    gcc still complains.

    To fix this add a compile-time condition for the extent buffer page
    array size limit, which would eventually lead to eliminating the whole
    for loop.

    CC: stable@vger.kernel.org # 5.10+
    Signed-off-by: pengfuyuan <pengfuyuan@kylinos.cn>
    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 55eb7a45276f9f9e3e307556dae79a60caac6f43
Author: Sherry Sun <sherry.sun@nxp.com>
Date: Fri May 19 17:47:51 2023 +0800

    tty: serial: fsl_lpuart: use UARTCTRL_TXINV to send break instead of UARTCTRL_SBK

    commit 2474e05467c00f7d51af3039b664de6886325257 upstream.

    LPUART IP now has two known bugs, one is that CTS has higher priority
    than the break signal, which causes the break signal sending through
    UARTCTRL_SBK may impacted by the CTS input if the HW flow control is
    enabled. It exists on all platforms we support in this driver.
    So we add a workaround patch for this issue: commit c4c81db5cf8b
    ("tty: serial: fsl_lpuart: disable the CTS when send break signal").

    Another IP bug is i.MX8QM LPUART may have an additional break character
    being sent after SBK was cleared. It may need to add some delay between
    clearing SBK and re-enabling CTS to ensure that the SBK latch are
    completely cleared.

    But we found that during the delay period before CTS is enabled, there
    is still a risk that Bluetooth data in TX FIFO may be sent out during
    this period because of break off and CTS disabled(even if BT sets CTS
    line deasserted, data is still sent to BT).

    Due to this risk, we have to drop the CTS-disabling workaround for SBK
    bugs, use TXINV seems to be a better way to replace SBK feature and
    avoid above risk. Also need to disable the transmitter to prevent any
    data from being sent out during break, then invert the TX line to send
    break. Then disable the TXINV when turn off break and re-enable
    transmitter.

    Fixes: c4c81db5cf8b ("tty: serial: fsl_lpuart: disable the CTS when send break signal")

```
    Cc: stable <stable@kernel.org>
    Signed-off-by: Sherry Sun <sherry.sun@nxp.com>
    Link: https://lore.kernel.org/r/20230519094751.28948-1-sherry.sun@nxp.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 54a71ee6480496392ed46e1563be9b729440619b
Author: Marek Vasut <marex@denx.de>
Date:   Sat May 13 21:23:52 2023 +0200

    mmc: pwrseq: sd8787: Fix WILC CHIP_EN and RESETN toggling order

    commit 0b5d5c436a5c572a45f976cfd34a6741e143e5d9 upstream.

    Chapter "5.3 Power-Up/Down Sequence" of WILC1000 [1] and WILC3000 [2]
    states that CHIP_EN must be pulled HIGH first, RESETN second. Fix the
    order of these signals in the driver.

    Use the mmc_pwrseq_ops as driver data as the delay between signals is
    specific to SDIO card type anyway.

    [1]
https://ww1.microchip.com/downloads/aemDocuments/documents/WSG/ProductDocuments/DataSheets/A
TWILC1000-MR110XB-IEEE-802.11-b-g-n-Link-Controller-Module-DS70005326E.pdf
    [2]
https://ww1.microchip.com/downloads/aemDocuments/documents/OTH/ProductDocuments/DataSheets/I
EEE-802.11-b-g-n-Link-Controller-Module-with-Integrated-Bluetooth-5.0-DS70005327B.pdf

    Fixes: b2832b96fcf5 ("mmc: pwrseq: sd8787: add support for wilc1000")
    Signed-off-by: Marek Vasut <marex@denx.de>
    Reviewed-by: Claudiu Beznea <claudiu.beznea@microchip.com>
    Cc: stable@vger.kernel.org
    Link: https://lore.kernel.org/r/20230513192352.479627-1-marex@denx.de
    Signed-off-by: Ulf Hansson <ulf.hansson@linaro.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 2d57d3be0a5e9498a1858531fe5a708ba73c14c6
Author: Deren Wu <deren.wu@mediatek.com>
Date:   Sat May 13 22:48:15 2023 +0800

    mmc: vub300: fix invalid response handling

    commit a99d21cefd351c8aaa20b83a3c942340e5789d45 upstream.

    We may get an empty response with zero length at the beginning of
    the driver start and get following UBSAN error. Since there is no
    content(SDRT_NONE) for the response, just return and skip the response
    handling to avoid this problem.

    Test pass : SDIO wifi throughput test with this patch

    [  126.980684] UBSAN: array-index-out-of-bounds in drivers/mmc/host/vub300.c:1719:12
    [  126.980709] index -1 is out of range for type 'u32 [4]'
    [  126.980729] CPU: 4 PID: 9 Comm: kworker/u16:0 Tainted: G            E      6.3.0-rc4-
mtk-local-202304272142 #1
    [  126.980754] Hardware name: Intel(R) Client Systems NUC8i7BEH/NUC8BEB, BIOS
BECFL357.86A.0081.2020.0504.1834 05/04/2020
    [  126.980770] Workqueue: kvub300c vub300_cmndwork_thread [vub300]
    [  126.980833] Call Trace:
    [  126.980845]  <TASK>
    [  126.980860]  dump_stack_lvl+0x48/0x70
    [  126.980895]  dump_stack+0x10/0x20
    [  126.980916]  ubsan_epilogue+0x9/0x40
    [  126.980944]  __ubsan_handle_out_of_bounds+0x70/0x90
    [  126.980979]  vub300_cmndwork_thread+0x58e7/0x5e10 [vub300]
    [  126.981018]  ? _raw_spin_unlock+0x18/0x40
```

```
[  126.981042]  ? finish_task_switch+0x175/0x6f0
[  126.981070]  ? __switch_to+0x42e/0xda0
[  126.981089]  ? __switch_to_asm+0x3a/0x80
[  126.981129]  ? __pfx_vub300_cmndwork_thread+0x10/0x10 [vub300]
[  126.981174]  ? __kasan_check_read+0x11/0x20
[  126.981204]  process_one_work+0x7ee/0x13d0
[  126.981246]  worker_thread+0x53c/0x1240
[  126.981291]  kthread+0x2b8/0x370
[  126.981312]  ? __pfx_worker_thread+0x10/0x10
[  126.981336]  ? __pfx_kthread+0x10/0x10
[  126.981359]  ret_from_fork+0x29/0x50
[  126.981400]  </TASK>

    Fixes: 88095e7b473a ("mmc: Add new VUB300 USB-to-SD/SDIO/MMC driver")
    Signed-off-by: Deren Wu <deren.wu@mediatek.com>
    Cc: stable@vger.kernel.org
    Link:
https://lore.kernel.org/r/048cd6972c50c33c2e8f81d5228fed928519918b.1683987673.git.deren.wu@m
ediatek.com
    Signed-off-by: Ulf Hansson <ulf.hansson@linaro.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1c2436dd9499d6a1c2ee3d5c757b32dd433aa25c
Author: Tim Huang <Tim.Huang@amd.com>
Date:   Mon May 22 23:17:28 2023 +0800

    drm/amd/pm: reverse mclk and fclk clocks levels for renoir

    commit 55e02c14f9b5fd973ba32a16a715baa42617f9c6 upstream.

    This patch reverses the DPM clocks levels output of pp_dpm_mclk
    and pp_dpm_fclk for renoir.

    On dGPUs and older APUs we expose the levels from lowest clocks
    to highest clocks. But for some APUs, the clocks levels are
    given the reversed orders by PMFW. Like the memory DPM clocks
    that are exposed by pp_dpm_mclk.

    It's not intuitive that they are reversed on these APUs. All tools
    and software that talks to the driver then has to know different ways
    to interpret the data depending on the asic.

    So we need to reverse them to expose the clocks levels from the
    driver consistently.

    Signed-off-by: Tim Huang <Tim.Huang@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 54f719be7b198aa879193418a8b53484fa22908b
Author: Tim Huang <Tim.Huang@amd.com>
Date:   Sun May 21 10:35:59 2023 +0800

    drm/amd/pm: reverse mclk and fclk clocks levels for yellow carp

    commit f1373a97a41f429e0095d4be388092ffa3c1a157 upstream.

    This patch reverses the DPM clocks levels output of pp_dpm_mclk
    and pp_dpm_fclk.

    On dGPUs and older APUs we expose the levels from lowest clocks
    to highest clocks. But for some APUs, the clocks levels that from
    the DFPstateTable are given the reversed orders by PMFW. Like the
```

```
    memory DPM clocks that are exposed by pp_dpm_mclk.

    It's not intuitive that they are reversed on these APUs. All tools
    and software that talks to the driver then has to know different ways
    to interpret the data depending on the asic.

    So we need to reverse them to expose the clocks levels from the
    driver consistently.

    Signed-off-by: Tim Huang <Tim.Huang@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 62cad6eed314ec6551fdcbc8f321807e64979e0e
Author: Tim Huang <Tim.Huang@amd.com>
Date:   Sun May 21 10:28:05 2023 +0800

    drm/amd/pm: reverse mclk clocks levels for SMU v13.0.5

    commit c1d35412b3e826ae8119e3fb5f51dd0fa5b6b567 upstream.

    This patch reverses the DPM clocks levels output of pp_dpm_mclk.

    On dGPUs and older APUs we expose the levels from lowest clocks
    to highest clocks. But for some APUs, the clocks levels that from
    the DFPstateTable are given the reversed orders by PMFW. Like the
    memory DPM clocks that are exposed by pp_dpm_mclk.

    It's not intuitive that they are reversed on these APUs. All tools
    and software that talks to the driver then has to know different ways
    to interpret the data depending on the asic.

    So we need to reverse them to expose the clocks levels from the
    driver consistently.

    Signed-off-by: Tim Huang <Tim.Huang@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 2b75325d0c8dd5260f380c64c3abd1dd475c1208
Author: Guchun Chen <guchun.chen@amd.com>
Date:   Wed May 24 15:03:02 2023 +0800

    drm/amd/pm: resolve reboot exception for si oland

    commit e490d60a2f76bff636c68ce4fe34c1b6c34bbd86 upstream.

    During reboot test on arm64 platform, it may failure on boot.

    The error message are as follows:
    [    1.706570][ 3] [  T273] [drm:si_thermal_enable_alert [amdgpu]] *ERROR* Could not
enable thermal interrupts.
    [    1.716547][ 3] [  T273] [drm:amdgpu_device_ip_late_init [amdgpu]] *ERROR* late_init
of IP block <si_dpm> failed -22
    [    1.727064][ 3] [  T273] amdgpu 0000:02:00.0: amdgpu_device_ip_late_init failed
    [    1.734367][ 3] [  T273] amdgpu 0000:02:00.0: Fatal error during GPU init

    v2: squash in built warning fix (Alex)

    Signed-off-by: Zhenneng Li <lizhenneng@kylinos.cn>
    Reviewed-by: Guchun Chen <guchun.chen@amd.com>
```

```
    Signed-off-by: Guchun Chen <guchun.chen@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 70bbff3b5fb36d770675c2d6f7651e28f243cb86
Author: Tim Huang <Tim.Huang@amd.com>
Date:   Sun May 21 11:10:19 2023 +0800

    drm/amd/pm: reverse mclk and fclk clocks levels for vangogh

    commit bfc03568d9d81332382c73a1985a90c4506bd36c upstream.

    This patch reverses the DPM clocks levels output of pp_dpm_mclk
    and pp_dpm_fclk.

    On dGPUs and older APUs we expose the levels from lowest clocks
    to highest clocks. But for some APUs, the clocks levels that from
    the DFPstateTable are given the reversed orders by PMFW. Like the
    memory DPM clocks that are exposed by pp_dpm_mclk.

    It's not intuitive that they are reversed on these APUs. All tools
    and software that talks to the driver then has to know different ways
    to interpret the data depending on the asic.

    So we need to reverse them to expose the clocks levels from the
    driver consistently.

    Signed-off-by: Tim Huang <Tim.Huang@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 d336b34755729821566a3ecbafa2b4ef9316a29e
Author: Tim Huang <Tim.Huang@amd.com>
Date:   Sun May 21 09:24:00 2023 +0800

    drm/amd/pm: reverse mclk and fclk clocks levels for SMU v13.0.4

    commit 6a07826f2057b5fa1c479ba56460195882464270 upstream.

    This patch reverses the DPM clocks levels output of pp_dpm_mclk
    and pp_dpm_fclk.

    On dGPUs and older APUs we expose the levels from lowest clocks
    to highest clocks. But for some APUs, the clocks levels that from
    the DFPstateTable are given the reversed orders by PMFW. Like the
    memory DPM clocks that are exposed by pp_dpm_mclk.

    It's not intuitive that they are reversed on these APUs. All tools
    and software that talks to the driver then has to know different ways
    to interpret the data depending on the asic.

    So we need to reverse them to expose the clocks levels from the
    driver consistently.

    Signed-off-by: Tim Huang <Tim.Huang@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 17502a5a431054cb032487de18d92e307d36ab5a
Author: Ikshwaku Chauhan <ikshwaku.chauhan@amd.com>
```

```
Date:   Thu May 25 10:57:26 2023 +0530

    drm/amdgpu: enable tmz by default for GC 11.0.1

    commit 663b930e24842f3d3bb79418bb5cd8d01b40c559 upstream.

    Add IP GC 11.0.1 in the list of target to have
    tmz enabled by default.

    Signed-off-by: Ikshwaku Chauhan <ikshwaku.chauhan@amd.com>
    Reviewed-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Cc: stable@vger.kernel.org # 6.1.x
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 6c72ae7cfc498539e2d5942daf855262ef23a746
Author: Damien Le Moal <dlemoal@kernel.org>
Date:   Mon May 22 20:09:57 2023 +0900

    ata: libata-scsi: Use correct device no in ata_find_dev()

    commit 7f875850f20a42f488840c9df7af91ef7db2d576 upstream.

    For devices not attached to a port multiplier and managed directly by
    libata, the device number passed to ata_find_dev() must always be lower
    than the maximum number of devices returned by ata_link_max_devices().
    That is 1 for SATA devices or 2 for an IDE link with master+slave
    devices. This device number is the SCSI device ID which matches these
    constraints as the IDs are generated per port and so never exceed the
    maximum number of devices for the link being used.

    However, for libsas managed devices, SCSI device IDs are assigned per
    struct scsi_host, leading to device IDs for SATA devices that can be
    well in excess of libata per-link maximum number of devices. This
    results in ata_find_dev() to always return NULL for libsas managed
    devices except for the first device of the target scsi_host with ID
    (device number) equal to 0. This issue is visible by executing the
    hdparm utility, which fails. E.g.:

    hdparm -i /dev/sdX
    /dev/sdX:
      HDIO_GET_IDENTITY failed: No message of desired type

    Fix this by rewriting ata_find_dev() to ignore the device number for
    non-PMP attached devices with a link with at most 1 device, that is SATA
    devices. For these, the device number 0 is always used to
    return the correct pointer to the struct ata_device of the port link.
    This change excludes IDE master/slave setups (maximum number of devices
    per link is 2) and port-multiplier attached devices. Also, to be
    consistant with the fact that SCSI device IDs and channel numbers used
    as device numbers are both unsigned int, change the devno argument of
    ata_find_dev() to unsigned int.

    Reported-by: Xingui Yang <yangxingui@huawei.com>
    Fixes: 41bda9c98035 ("libata-link: update hotplug to handle PMP links")
    Cc: stable@vger.kernel.org
    Signed-off-by: Damien Le Moal <dlemoal@kernel.org>
    Reviewed-by: Jason Yan <yanaijie@huawei.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1d56b5ba6ef2249ad784b5dc79b96f93c33acaab
Author: Bart Van Assche <bvanassche@acm.org>
Date:   Mon May 29 12:50:34 2023 -0700

    scsi: stex: Fix gcc 13 warnings
```

```
    commit 6d074ce231772c66e648a61f6bd2245e7129d1f5 upstream.

    gcc 13 may assign another type to enumeration constants than gcc 12. Split
    the large enum at the top of source file stex.c such that the type of the
    constants used in time expressions is changed back to the same type chosen
    by gcc 12. This patch suppresses compiler warnings like this one:

    In file included from ./include/linux/bitops.h:7,
                     from ./include/linux/kernel.h:22,
                     from drivers/scsi/stex.c:13:
    drivers/scsi/stex.c: In function ‘stex_common_handshake’:
    ./include/linux/typecheck.h:12:25: error: comparison of distinct pointer types lacks a
  cast [-Werror]
       12 |         (void)(&__dummy == &__dummy2); \
          |                         ^~
    ./include/linux/jiffies.h:106:10: note: in expansion of macro ‘typecheck’
      106 |          typecheck(unsigned long, b) && \
          |          ^~~~~~~~~
    drivers/scsi/stex.c:1035:29: note: in expansion of macro ‘time_after’
     1035 |                         if (time_after(jiffies, before + MU_MAX_DELAY * HZ)) {
          |                             ^~~~~~~~~~

    See also https://gcc.gnu.org/bugzilla/show_bug.cgi?id=107405.

    Cc: stable@vger.kernel.org
    Acked-by: Randy Dunlap <rdunlap@infradead.org>
    Tested-by: Randy Dunlap <rdunlap@infradead.org> # build-tested
    Signed-off-by: Bart Van Assche <bvanassche@acm.org>
    Link: https://lore.kernel.org/r/20230529195034.3077-1-bvanassche@acm.org
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

  commit 4ca7101fa43f43b3c51ec3330a592adc694b900a
  Author: Richard Acayan <mailingradian@gmail.com>
  Date:   Tue May 23 16:25:50 2023 +0100

    misc: fastrpc: reject new invocations during device removal

    commit 46248400d81e2aa0b65cd659d6f40188192a58b6 upstream.

    The channel's rpmsg object allows new invocations to be made. After old
    invocations are already interrupted, the driver shouldn't try to invoke
    anymore. Invalidating the rpmsg at the end of the driver removal
    function makes it easy to cause a race condition in userspace. Even
    closing a file descriptor before the driver finishes its cleanup can
    cause an invocation via fastrpc_release_current_dsp_process() and
    subsequent timeout.

    Invalidate the channel before the invocations are interrupted to make
    sure that no invocations can be created to hang after the device closes.

    Fixes: c68cfb718c8f ("misc: fastrpc: Add support for context Invoke method")
    Cc: stable <stable@kernel.org>
    Signed-off-by: Richard Acayan <mailingradian@gmail.com>
    Signed-off-by: Srinivas Kandagatla <srinivas.kandagatla@linaro.org>
    Link: https://lore.kernel.org/r/20230523152550.438363-5-srinivas.kandagatla@linaro.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

  commit 4553293e11ba77871f7c4df9a2f542a99738b149
  Author: Richard Acayan <mailingradian@gmail.com>
  Date:   Tue May 23 16:25:49 2023 +0100

    misc: fastrpc: return -EPIPE to invocations on device removal
```

```
    commit b6a062853ddf6b4f653af2d8b75ba45bb9a036ad upstream.

    The return value is initialized as -1, or -EPERM. The completion of an
    invocation implies that the return value is set appropriately, but
    "Permission denied" does not accurately describe the outcome of the
    invocation. Set the invocation's return value to a more appropriate
    "Broken pipe", as the cleanup breaks the driver's connection with rpmsg.

    Fixes: c68cfb718c8f ("misc: fastrpc: Add support for context Invoke method")
    Cc: stable <stable@kernel.org>
    Signed-off-by: Richard Acayan <mailingradian@gmail.com>
    Reviewed-by: Srinivas Kandagatla <srinivas.kandagatla@linaro.org>
    Signed-off-by: Srinivas Kandagatla <srinivas.kandagatla@linaro.org>
    Link: https://lore.kernel.org/r/20230523152550.438363-4-srinivas.kandagatla@linaro.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 8ee6aaaf6ab4c099724c3f77259d0b4b6c81f204
Author: Ekansh Gupta <quic_ekangupt@quicinc.com>
Date:   Tue May 23 16:25:48 2023 +0100

    misc: fastrpc: Reassign memory ownership only for remote heap

    commit 3c7d0079a1831118ef232bd9c2f34d058a1f31c2 upstream.

    The userspace map request for remote heap allocates CMA memory.
    The ownership of this memory needs to be reassigned to proper
    owners to allow access from the protection domain running on
    DSP. This reassigning of ownership is not correct if done for
    any other supported flags.

    When any other flag is requested from userspace, fastrpc is
    trying to reassign the ownership of memory and this reassignment
    is getting skipped for remote heap request which is incorrect.
    Add proper flag check to reassign the memory only if remote heap
    is requested.

    Fixes: 532ad70c6d44 ("misc: fastrpc: Add mmap request assigning for static PD pool")
    Cc: stable <stable@kernel.org>
    Tested-by: Ekansh Gupta <quic_ekangupt@quicinc.com>
    Signed-off-by: Ekansh Gupta <quic_ekangupt@quicinc.com>
    Reviewed-by: Srinivas Kandagatla <srinivas.kandagatla@linaro.org>
    Signed-off-by: Srinivas Kandagatla <srinivas.kandagatla@linaro.org>
    Link: https://lore.kernel.org/r/20230523152550.438363-3-srinivas.kandagatla@linaro.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b27d5c4a2902eaad1467568cd04c16faf54f5ec7
Author: Yu Kuai <yukuai3@huawei.com>
Date:   Wed May 24 09:41:18 2023 +0800

    md/raid5: fix miscalculation of 'end_sector' in raid5_read_one_chunk()

    commit 8557dc27126949c702bd3aafe8a7e0b7e4fcb44c upstream.

    'end_sector' is compared to 'rdev->recovery_offset', which is offset to
    rdev, however, commit e82ed3a4fbb5 ("md/raid6: refactor
    raid5_read_one_chunk") changes the calculation of 'end_sector' to offset
    to the array. Fix this miscalculation.

    Fixes: e82ed3a4fbb5 ("md/raid6: refactor raid5_read_one_chunk")
    Cc: stable@vger.kernel.org # v5.12+
    Signed-off-by: Yu Kuai <yukuai3@huawei.com>
    Reviewed-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Song Liu <song@kernel.org>
    Link: https://lore.kernel.org/r/20230524014118.3172781-1-yukuai1@huaweicloud.com
```

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

commit 7d4d05d3b15bf005c4e82981e7b0ba0455c5b024
Author: Uttkarsh Aggarwal <quic_uaggarwa@quicinc.com>
Date:   Thu May 25 14:58:54 2023 +0530

    usb: gadget: f_fs: Add unbind event before functionfs_unbind

    commit efb6b535207395a5c7317993602e2503ca8cb4b3 upstream.

    While exercising the unbind path, with the current implementation
    the functionfs_unbind would be calling which waits for the ffs->mutex
    to be available, however within the same time ffs_ep0_read is invoked
    & if no setup packets are pending, it will invoke function
    wait_event_interruptible_exclusive_locked_irq which by definition waits
    for the ev.count to be increased inside the same mutex for which
    functionfs_unbind is waiting.
    This creates deadlock situation because the functionfs_unbind won't
    get the lock until ev.count is increased which can only happen if
    the caller ffs_func_unbind can proceed further.

    Following is the illustration:

            CPU1                            CPU2

    ffs_func_unbind()               ffs_ep0_read()
                                    mutex_lock(ffs->mutex)
                                    wait_event(ffs->ev.count)
    functionfs_unbind()
      mutex_lock(ffs->mutex)
      mutex_unlock(ffs->mutex)

    ffs_event_add()

    <deadlock>

    Fix this by moving the event unbind before functionfs_unbind
    to ensure the ev.count is incrased properly.

    Fixes: 6a19da111057 ("usb: gadget: f_fs: Prevent race during ffs_ep0_queue_wait")
    Cc: stable <stable@kernel.org>
    Signed-off-by: Uttkarsh Aggarwal <quic_uaggarwa@quicinc.com>
    Link: https://lore.kernel.org/r/20230525092854.7992-1-quic_uaggarwa@quicinc.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a7fe2d28d77f28f3aff4594f520785274512f419
Author: Frank Li <Frank.Li@nxp.com>
Date:   Thu May 18 11:49:45 2023 -0400

    usb: cdns3: fix NCM gadget RX speed 20x slow than expection at iMX8QM

    commit dbe678f6192f27879ac9ff6bc7a1036aad85aae9 upstream.

    At iMX8QM platform, enable NCM gadget and run 'iperf3 -s'.
    At host, run 'iperf3 -V -c fe80::6863:98ff:feef:3e0%enxc6e147509498'

    [  5]   0.00-1.00   sec  1.55 MBytes  13.0 Mbits/sec   90   4.18 KBytes
    [  5]   1.00-2.00   sec  1.44 MBytes  12.0 Mbits/sec   75   4.18 KBytes
    [  5]   2.00-3.00   sec  1.48 MBytes  12.4 Mbits/sec   75   4.18 KBytes

    Expected speed should be bigger than 300Mbits/sec.

    The root cause of this performance drop was found to be data corruption
    happening at 4K borders in some Ethernet packets, leading to TCP
    checksum errors. This corruption occurs from the position
```

```
    (4K - (address & 0x7F)) to 4K. The u_ether function's allocation of
    skb_buff reserves 64B, meaning all RX addresses resemble 0xXXXX0040.

    Force trb_burst_size to 16 can fix this problem.

    Cc: stable@vger.kernel.org
    Fixes: 7733f6c32e36 ("usb: cdns3: Add Cadence USB3 DRD Driver")
    Signed-off-by: Frank Li <Frank.Li@nxp.com>
    Link: https://lore.kernel.org/r/20230518154946.3666662-1-Frank.Li@nxp.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f154055beea248087d03d83076c24d94cbba3ba4
Author: Marek Vasut <marex@denx.de>
Date:   Mon May 15 19:24:56 2023 +0200

    dt-bindings: usb: snps,dwc3: Fix "snps,hsphy_interface" type

    commit 7b32040f6d7f885ffc09a6df7c17992d56d2eab8 upstream.

    The "snps,hsphy_interface" is string, not u8. Fix the type.

    Fixes: 389d77658801 ("dt-bindings: usb: Convert DWC USB3 bindings to DT schema")
    Cc: stable <stable@kernel.org>
    Reviewed-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Signed-off-by: Marek Vasut <marex@denx.de>
    Link: https://lore.kernel.org/r/20230515172456.179049-1-marex@denx.de
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 32e56ba493f6367acc89e4f6a4d25dc02c0d87f5
Author: Sebastian Krzyszkowiak <sebastian.krzyszkowiak@puri.sm>
Date:   Fri May 26 16:38:11 2023 +0200

    net: usb: qmi_wwan: Set DTR quirk for BroadMobi BM818

    commit 36936a56e1814f6c526fe71fbf980beab4f5577a upstream.

    BM818 is based on Qualcomm MDM9607 chipset.

    Fixes: 9a07406b00cd ("net: usb: qmi_wwan: Add the BroadMobi BM818 card")
    Cc: stable@vger.kernel.org
    Signed-off-by: Sebastian Krzyszkowiak <sebastian.krzyszkowiak@puri.sm>
    Acked-by: Bjørn Mork <bjorn@mork.no>
    Link: https://lore.kernel.org/r/20230526-bm818-dtr-v1-1-64bbfa6ba8af@puri.sm
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ad27bd5e4a421e366772dd10da33b8cbfa240365
Author: Lukas Bulwahn <lukas.bulwahn@gmail.com>
Date:   Mon May 8 06:02:08 2023 +0200

    iio: dac: build ad5758 driver when AD5758 is selected

    commit a146eccb68be161ae9eab5f3f68bb0ed7c0fbaa8 upstream.

    Commit 28d1a7ac2a0d ("iio: dac: Add AD5758 support") adds the config AD5758
    and the corresponding driver ad5758.c. In the Makefile, the ad5758 driver
    is however included when AD5755 is selected, not when AD5758 is selected.

    Probably, this was simply a mistake that happened by copy-and-paste and
    forgetting to adjust the actual line. Surprisingly, no one has ever noticed
    that this driver is actually only included when AD5755 is selected and that
    the config AD5758 has actually no effect on the build.

    Fixes: 28d1a7ac2a0d ("iio: dac: Add AD5758 support")
    Signed-off-by: Lukas Bulwahn <lukas.bulwahn@gmail.com>
```

```
    Link: https://lore.kernel.org/r/20230508040208.12033-1-lukas.bulwahn@gmail.com
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit bd844f8b87670cf8b709830f605863ced5578979
Author: Sean Nyekjaer <sean@geanix.com>
Date:   Wed May 3 18:20:28 2023 +0200

    iio: adc: stm32-adc: skip adc-diff-channels setup if none is present

    commit 9c0d6ccd7d6bbd275e390b55a3390b4274291d95 upstream.

    If no adc differential channels are defined driver will fail with EINVAL:
    stm32-adc: probe of 48003000.adc:adc@0 failed with error -22

    Fix this by skipping the initialization if no channels are defined.

    This applies only to the legacy way of initializing adc channels.

    Fixes: d7705f35448a ("iio: adc: stm32-adc: convert to device properties")
    Signed-off-by: Sean Nyekjaer <sean@geanix.com>
    Link: https://lore.kernel.org/r/20230503162029.3654093-1-sean@geanix.com
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b91145204a9aa55a5c1d30b36ebb75d003b45393
Author: Paul Cercueil <paul@crapouillou.net>
Date:   Thu Mar 30 12:21:00 2023 +0200

    iio: adc: ad7192: Change "shorted" channels to differential

    commit e55245d115bb9054cb72cdd5dda5660f4484873a upstream.

    The AD7192 provides a specific channel configuration where both negative
    and positive inputs are connected to AIN2. This was represented in the
    ad7192 driver as a IIO channel with .channel = 2 and .extended_name set
    to "shorted".

    The problem with this approach, is that the driver provided two IIO
    channels with the identifier .channel = 2; one "shorted" and the other
    not. This goes against the IIO ABI, as a channel identifier should be
    unique.

    Address this issue by changing "shorted" channels to being differential
    instead, with channel 2 vs. itself, as we're actually measuring AIN2 vs.
    itself.

    Note that the fix tag is for the commit that moved the driver out of
    staging. The bug existed before that, but backporting would become very
    complex further down and unlikely to happen.

    Fixes: b581f748cce0 ("staging: iio: adc: ad7192: move out of staging")
    Signed-off-by: Paul Cercueil <paul@crapouillou.net>
    Co-developed-by: Alisa Roman <alisa.roman@analog.com>
    Signed-off-by: Alisa Roman <alisa.roman@analog.com>
    Reviewed-by: Nuno Sa <nuno.sa@analog.com>
    Link: https://lore.kernel.org/r/20230330102100.17590-1-paul@crapouillou.net
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c983ed4315e4e1517d3dbdc82497106fc85682b4
Author: Rasmus Villemoes <linux@rasmusvillemoes.dk>
```

```
Date:   Wed May 3 11:58:17 2023 +0200

    iio: addac: ad74413: fix resistance input processing

    commit 24febc99ca725dcf42d57168a2f4e8a75a5ade92 upstream.

    On success, ad74413r_get_single_adc_result() returns IIO_VAL_INT aka
    1. So currently, the IIO_CHAN_INFO_PROCESSED case is effectively
    equivalent to the IIO_CHAN_INFO_RAW case, and we never call
    ad74413r_adc_to_resistance_result() to convert the adc measurement to
    ohms.

    Check ret for being negative rather than non-zero.

    Fixes: fea251b6a5dbd (iio: addac: add AD74413R driver)
    Signed-off-by: Rasmus Villemoes <linux@rasmusvillemoes.dk>
    Reviewed-by: Nuno Sa <nuno.sa@analog.com>
    Link: https://lore.kernel.org/r/20230503095817.452551-1-linux@rasmusvillemoes.dk
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit cf0880de552a95234a994e781139c3308b0b69ae
Author: Marek Vasut <marex@denx.de>
Date:   Thu May 11 02:43:30 2023 +0200

    iio: dac: mcp4725: Fix i2c_master_send() return value handling

    commit 09d3bec7009186bdba77039df01e5834788b3f95 upstream.

    The i2c_master_send() returns number of sent bytes on success,
    or negative on error. The suspend/resume callbacks expect zero
    on success and non-zero on error. Adapt the return value of the
    i2c_master_send() to the expectation of the suspend and resume
    callbacks, including proper validation of the return value.

    Fixes: cf35ad61aca2 ("iio: add mcp4725 I2C DAC driver")
    Signed-off-by: Marek Vasut <marex@denx.de>
    Reviewed-by: Uwe Kleine-König <u.kleine-koenig@pengutronix.de>
    Link: https://lore.kernel.org/r/20230511004330.206942-1-marex@denx.de
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 4c13d06d16f0baf5c7888024a30ecda22658e994
Author: Masahiro Honda <honda@mechatrax.com>
Date:   Thu May 18 20:08:16 2023 +0900

    iio: adc: ad_sigma_delta: Fix IRQ issue by setting IRQ_DISABLE_UNLAZY flag

    commit 626d312028bec44209d0ecd5beaa9b1aa8945f7d upstream.

    The Sigma-Delta ADCs supported by this driver can use SDO as an interrupt
    line to indicate the completion of a conversion. However, some devices
    cannot properly detect the completion of a conversion by an interrupt.
    This is for the reason mentioned in the following commit.

    commit e9849777d0e2 ("genirq: Add flag to force mask in
                          disable_irq[_nosync]()")

    A read operation is performed by an extra interrupt before the completion
    of a conversion. At this time, the value read from the ADC data register
    is the same as the previous conversion result. This patch fixes the issue
    by setting IRQ_DISABLE_UNLAZY flag.
```

```
    Fixes: 0c6ef985a1fd ("iio: adc: ad7791: fix IRQ flags")
    Fixes: 1a913270e57a ("iio: adc: ad7793: Fix IRQ flag")
    Fixes: e081102f3077 ("iio: adc: ad7780: Fix IRQ flag")
    Fixes: 89a86da5cb8e ("iio: adc: ad7192: Add IRQ flag")
    Fixes: 79ef91493f54 ("iio: adc: ad7124: Set IRQ type to falling")
    Signed-off-by: Masahiro Honda <honda@mechatrax.com>
    Link: https://lore.kernel.org/r/20230518110816.248-1-honda@mechatrax.com
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c6a3fcf98553ac6d59b7c089fdaccb3a68d30449
Author: Sean Nyekjaer <sean@geanix.com>
Date:   Wed May 3 18:20:29 2023 +0200

    iio: adc: stm32-adc: skip adc-channels setup if none is present

    commit 3e27ef0ced49f8ae7883c25fadf76a2086e99025 upstream.

    If only adc differential channels are defined driver will fail with
    stm32-adc: probe of 48003000.adc:adc@0 failed with error -22

    Fix this by skipping the initialization if no channels are defined.

    This applies only to the legacy way of initializing adc channels.

    Fixes: d7705f35448a ("iio: adc: stm32-adc: convert to device properties")
    Signed-off-by: Sean Nyekjaer <sean@geanix.com>
    Reviewed-by: Olivier Moysan <olivier.moysan@foss.st.com>
    Link: https://lore.kernel.org/r/20230503162029.3654093-2-sean@geanix.com
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1628d08fa07da617f09d996476b3e814cf1221cf
Author: Matti Vaittinen <mazziesaccount@gmail.com>
Date:   Fri May 12 10:53:41 2023 +0300

    iio: accel: kx022a fix irq getting

    commit 56cd3d1c5c5b073a1a444eafdcf97d4d866d351a upstream.

    The fwnode_irq_get_byname() was returning 0 at device-tree mapping
    error. If this occurred, the KX022A driver did abort the probe but
    errorneously directly returned the return value from
    fwnode_irq_get_byname() from probe. In case of a device-tree mapping
    error this indicated success.

    The fwnode_irq_get_byname() has since been fixed to not return zero on
    error so the check for fwnode_irq_get_byname() can be relaxed to only
    treat negative values as errors. This will also do decent fix even when
    backported to branches where fwnode_irq_get_byname() can still return
    zero on error because KX022A probe should later fail at IRQ requesting
    and a prober error handling should follow.

    Relax the return value check for fwnode_irq_get_byname() to only treat
    negative values as errors.

    Reported-by: kernel test robot <lkp@intel.com>
    Reported-by: Dan Carpenter <error27@gmail.com>
    Closes: https://lore.kernel.org/r/202305110245.MFxC9bUj-lkp@intel.com/
    Link: https://lore.kernel.org/r/202305110245.MFxC9bUj-lkp@intel.com/
    Signed-off-by: Matti Vaittinen <mazziesaccount@gmail.com>
    Fixes: 7c1d1677b322 ("iio: accel: Support Kionix/ROHM KX022A accelerometer")
    Link:
```

```
https://lore.kernel.org/r/b45b4b638db109c6078d243252df3a7b0485f7d5.1683875389.git.mazziesacc
ount@gmail.com
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 261be970278768f1ec19e046b24a3756dc11b66a
Author: Frank Li <Frank.Li@nxp.com>
Date:   Mon May 1 10:36:04 2023 -0400

    iio: light: vcnl4035: fixed chip ID check

    commit a551c26e8e568fad42120843521529241b9bceec upstream.

    VCNL4035 register(0xE) ID_L and ID_M define as:

     ID_L: 0x80
     ID_H: 7:6 (0:0)
           5:4 (0:0) slave address = 0x60 (7-bit)
               (0:1) slave address = 0x51 (7-bit)
               (1:0) slave address = 0x40 (7-bit)
               (1:0) slave address = 0x41 (7-bit)
           3:0 Version code default (0:0:0:0)

    So just check ID_L.

    Fixes: 55707294c4eb ("iio: light: Add support for vishay vcnl4035")
    Signed-off-by: Frank Li <Frank.Li@nxp.com>
    Link: https://lore.kernel.org/r/20230501143605.1615549-1-Frank.Li@nxp.com
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 865a38c61e58cd7d1aaa7be2b1e688eee706f2a5
Author: Geert Uytterhoeven <geert+renesas@glider.be>
Date:   Tue May 9 14:34:22 2023 +0200

    dt-bindings: iio: adc: renesas,rcar-gyroadc: Fix adi,ad7476 compatible value

    commit 55720d242052e860b9fde445e302e0425722e7f1 upstream.

    The conversion to json-schema accidentally dropped the "ad" part prefix
    from the compatible value.

    Fixes: 8c41245872e2 ("dt-bindings:iio:adc:renesas,rcar-gyroadc: txt to yaml
conversion.")
    Signed-off-by: Geert Uytterhoeven <geert+renesas@glider.be>
    Reviewed-by: Marek Vasut <marek.vasut+renesas@mailbox.org>
    Reviewed-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Reviewed-by: Wolfram Sang <wsa+renesas@sang-engineering.com>
    Link:
https://lore.kernel.org/r/6b328a3f52657c20759f3a5bb2fe033d47644ba8.1683635404.git.geert+rene
sas@glider.be
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c65b8d6736f636bae1641fa91ad7a594acb94a88
Author: Jean-Baptiste Maneyrol <jean-baptiste.maneyrol@tdk.com>
Date:   Tue May 9 15:22:02 2023 +0000

    iio: imu: inv_icm42600: fix timestamp reset

    commit bbaae0c79ebd49f61ad942a8bf9e12bfc7f821bb upstream.
```

```
    Timestamp reset is not done in the correct place. It must be done
    before enabling buffer. The reason is that interrupt timestamping
    is always happening when the chip is on, even if the
    corresponding sensor is off. When the sensor restarts, timestamp
    is wrong if you don't do a reset first.

    Fixes: ec74ae9fd37c ("iio: imu: inv_icm42600: add accurate timestamping")
    Signed-off-by: Jean-Baptiste Maneyrol <jean-baptiste.maneyrol@tdk.com>
    Cc: <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20230509152202.245444-1-inv.git-commit@tdk.com
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 795a6a3b9bebf545339912e8c3c4804474cdf78c
Author: Nikita Zhandarovich <n.zhandarovich@fintech.ru>
Date:   Mon Apr 17 09:01:48 2023 -0700

    HID: wacom: avoid integer overflow in wacom_intuos_inout()

    commit bd249b91977b768ea02bf84d04625d2690ad2b98 upstream.

    If high bit is set to 1 in ((data[3] & 0x0f << 28), after all arithmetic
    operations and integer promotions are done, high bits in
    wacom->serial[idx] will be filled with 1s as well.
    Avoid this, albeit unlikely, issue by specifying left operand's __u64
    type for the right operand.

    Found by Linux Verification Center (linuxtesting.org) with static
    analysis tool SVACE.

    Fixes: 3bea733ab212 ("USB: wacom tablet driver reorganization")
    Signed-off-by: Nikita Zhandarovich <n.zhandarovich@fintech.ru>
    Reviewed-by: Ping Cheng <ping.cheng@wacom.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Jiri Kosina <jkosina@suse.cz>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 6ff743afaf33c0cb57a70f988a2dec9d95e25f76
Author: Sung-Chi Li <lschyi@chromium.org>
Date:   Mon Apr 24 10:37:36 2023 +0800

    HID: google: add jewel USB id

    commit ed84c4517a5bc536e8572a01dfa11bc22a280d06 upstream.

    Add 1 additional hammer-like device.

    Signed-off-by: Sung-Chi Li <lschyi@chromium.org>
    Signed-off-by: Jiri Kosina <jkosina@suse.cz>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit d99168a26329470249013246c1f2d893887e3f22
Author: ChiaEn Wu <chiaen_wu@richtek.com>
Date:   Mon Apr 10 18:34:22 2023 +0800

    iio: adc: mt6370: Fix ibus and ibat scaling value of some specific vendor ID chips

    commit 00ffdd6fa90298522d45ca0c348b23485584dcdc upstream.

    The scale value of ibus and ibat on the datasheet is incorrect due to the
    customer report after the experimentation with some specific vendor ID
    chips.

    Fixes: c1404d1b659f ("iio: adc: mt6370: Add MediaTek MT6370 support")
    Signed-off-by: ChiaEn Wu <chiaen_wu@richtek.com>
```

```
    Reviewed-by: Alexandre Mergnat <amergnat@baylibre.com>
    Link: https://lore.kernel.org/r/1681122862-1994-1-git-send-email-chiaen_wu@richtek.com
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 029e23f9750de90a162207f5fecbb9f5582e7504
Author: Lars-Peter Clausen <lars@metafoo.de>
Date:   Fri Apr 14 08:07:02 2023 -0700

    iio: ad4130: Make sure clock provider gets removed

    commit 28f73ded19d403697f87473c9b85a27eb8ed9cf2 upstream.

    The ad4130 driver registers a clock provider, but never removes it. This
    leaves a stale clock provider behind that references freed clocks when the
    device is unbound.

    Register a managed action to remove the clock provider when the device is
    removed.

    Fixes: 62094060cf3a ("iio: adc: ad4130: add AD4130 driver")
    Signed-off-by: Lars-Peter Clausen <lars@metafoo.de>
    Link: https://lore.kernel.org/r/20230414150702.518441-1-lars@metafoo.de
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 5b3d3fd0108929bf3c1c5a5846b8a5dd87002098
Author: Lars-Peter Clausen <lars@metafoo.de>
Date:   Thu Apr 13 18:37:52 2023 -0700

    iio: tmag5273: Fix runtime PM leak on measurement error

    commit 265c82ea8b172129cb6d4eff41af856c3aff6168 upstream.

    The tmag5273 gets a runtime PM reference before reading a measurement and
    releases it when done. But if the measurement fails the tmag5273_read_raw()
    function exits before releasing the reference.

    Make sure that this error path also releases the runtime PM reference.

    Fixes: 866a1389174b ("iio: magnetometer: add ti tmag5273 driver")
    Signed-off-by: Lars-Peter Clausen <lars@metafoo.de>
    Acked-by: Gerald Loacker <gerald.loacker@wolfvision.net>
    Reviewed-by: Nuno Sa <nuno.sa@analog.com>
    Link: https://lore.kernel.org/r/20230414013752.498767-1-lars@metafoo.de
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 10385ddf6667fe433ecc52809e078d98bce606f7
Author: Jiakai Luo <jkluo@hust.edu.cn>
Date:   Sat Apr 22 06:34:06 2023 -0700

    iio: adc: mxs-lradc: fix the order of two cleanup operations

    commit 27b2ed5b6d53cd62fc61c3f259ae52f5cac23b66 upstream.

    Smatch reports:
    drivers/iio/adc/mxs-lradc-adc.c:766 mxs_lradc_adc_probe() warn:
    missing unwind goto?

    the order of three init operation:
    1.mxs_lradc_adc_trigger_init
```

```
    2.iio_triggered_buffer_setup
    3.mxs_lradc_adc_hw_init

    thus, the order of three cleanup operation should be:
    1.mxs_lradc_adc_hw_stop
    2.iio_triggered_buffer_cleanup
    3.mxs_lradc_adc_trigger_remove

    we exchange the order of two cleanup operations,
    introducing the following differences:
    1.if mxs_lradc_adc_trigger_init fails, returns directly;
    2.if trigger_init succeeds but iio_triggered_buffer_setup fails,
    goto err_trig and remove the trigger.

    In addition, we also reorder the unwind that goes on in the
    remove() callback to match the new ordering.

    Fixes: 6dd112b9f85e ("iio: adc: mxs-lradc: Add support for ADC driver")
    Signed-off-by: Jiakai Luo <jkluo@hust.edu.cn>
    Reviewed-by: Dongliang Mu <dzm91@hust.edu.cn>
    Link: https://lore.kernel.org/r/20230422133407.72908-1-jkluo@hust.edu.cn
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 4405f1023686719c727c4594345be0ee99b29c31
Author: Hans de Goede <hdegoede@redhat.com>
Date:   Sun Apr 16 23:24:09 2023 +0200

    iio: accel: st_accel: Fix invalid mount_matrix on devices without ACPI _ONT method

    commit 79b8ded9d9c595db9bd5b2f62f5f738b36de1e22 upstream.

    When apply_acpi_orientation() fails, st_accel_common_probe() will fall back
    to iio_read_mount_matrix(), which checks for a mount-matrix device property
    and if that is not set falls back to the identity matrix.

    But when a sensor has no ACPI companion fwnode, or when the ACPI fwnode
    does not have a "_ONT" method apply_acpi_orientation() was returning 0,
    causing iio_read_mount_matrix() to never get called resulting in an
    invalid mount_matrix:

    [root@fedora ~]# cat /sys/bus/iio/devices/iio\:device0/mount_matrix
    (null), (null), (null); (null), (null), (null); (null), (null), (null)

    Fix this by making apply_acpi_orientation() always return an error when
    it did not set the mount_matrix.

    Fixes: 3d8ad94bb175 ("iio: accel: st_sensors: Support generic mounting matrix")
    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Reviewed-by: Linus Walleij <linus.walleij@linaro.org>
    Tested-by: Marius Hoch <mail@mariushoch.de>
    Link: https://lore.kernel.org/r/20230416212409.310936-1-hdegoede@redhat.com
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit bb533af39ae139fd1bb5ea2816a6f64f6405dc13
Author: Aric Cyr <aric.cyr@amd.com>
Date:   Sat Feb 11 10:03:22 2023 -0500

    drm/amd/display: Only wait for blank completion if OTG active

    [ Upstream commit 82a10aff9428f1d190de55ef7971fdb84303cc7a ]
```

```
    [why]
    If OTG is not active, waiting for blank completion will always fail and
    timeout resulting in unnecessary driver delays.

    [how]
    Check that OTG is enabled before waiting for blank.

    Reviewed-by: Alvin Lee <Alvin.Lee2@amd.com>
    Acked-by: Qingqing Zhuo <qingqing.zhuo@amd.com>
    Signed-off-by: Aric Cyr <aric.cyr@amd.com>
    Tested-by: Daniel Wheeler <daniel.wheeler@amd.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4188d68dccc2134e4a4a7db81103baf7de0ccdb7
Author: Masami Hiramatsu (Google) <mhiramat@kernel.org>
Date:   Sun Mar 19 11:53:32 2023 +0900

    selftests/ftrace: Choose target function for filter test from samples

    [ Upstream commit eb50d0f250e96ede9192d936d220cd97adc93b89 ]

    Since the event-filter-function.tc expects the 'exit_mmap()' directly
    calls 'kmem_cache_free()', this is vulnerable to code modifications.

    Choose the target function for the filter test from the sample
    event data so that it can keep test running correctly even if the caller
    function name will be changed.

    Link: https://lore.kernel.org/linux-trace-
kernel/167919441260.1922645.18355804179347364057.stgit@mhiramat.roam.corp.google.com/

    Link: https://lore.kernel.org/all/CA+G9fYtF-XEKi9YNGgR=Kf==7iRb2FrmEC7qtwAeQbfyah-
UhA@mail.gmail.com/
    Reported-by: Linux Kernel Functional Testing <lkft@linaro.org>
    Fixes: 7f09d639b8c4 ("tracing/selftests: Add test for event filtering on function name")
    Signed-off-by: Masami Hiramatsu (Google) <mhiramat@kernel.org>
    Acked-by: Steven Rostedt (Google) <rostedt@goodmis.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5a698e42aca0bac99629f3a6f07b2f38c83ad85e
Author: Laurent Pinchart <laurent.pinchart@ideasonboard.com>
Date:   Thu Apr 20 10:45:59 2023 +0100

    media: uvcvideo: Don't expose unsupported formats to userspace

    [ Upstream commit 81f3affa19d6ab0c32aef46b053838219eef7e71 ]

    When the uvcvideo driver encounters a format descriptor with an unknown
    format GUID, it creates a corresponding struct uvc_format instance with
    the fcc field set to 0. Since commit 50459f103edf ("media: uvcvideo:
    Remove format descriptions"), the driver relies on the V4L2 core to
    provide the format description string, which the V4L2 core can't do
    without a valid 4CC. This triggers a WARN_ON.

    As a format with a zero 4CC can't be selected, it is unusable for
    applications. Ignore the format completely without creating a uvc_format
    instance, which fixes the warning.

    Link: https://bugzilla.kernel.org/show_bug.cgi?id=217252
    Link: https://bugzilla.redhat.com/show_bug.cgi?id=2180107

    Fixes: 50459f103edf ("media: uvcvideo: Remove format descriptions")
    Signed-off-by: Laurent Pinchart <laurent.pinchart@ideasonboard.com>
    Reviewed-by: Ricardo Ribalda <ribalda@chromium.org>
```

```
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1c1bbd477ed644ac7276ee6fce4af09b30892878
Author: Francesco Dolcini <francesco.dolcini@toradex.com>
Date:   Wed May 31 13:10:38 2023 +0200

    dt-bindings: serial: 8250_omap: add rs485-rts-active-high

    [ Upstream commit 403e97d6ab2cb6fd0ac1ff968cd7b691771f1613 ]

    Add rs485-rts-active-high property, this was removed by mistake.
    In general we just use rs485-rts-active-low property, however the OMAP
    UART for legacy reason uses the -high one.

    Fixes: 767d3467eb60 ("dt-bindings: serial: 8250_omap: drop rs485 properties")
    Closes: https://lore.kernel.org/all/ZGefR4mTHHo1iQ7H@francesco-nb.int.toradex.com/
    Signed-off-by: Francesco Dolcini <francesco.dolcini@toradex.com>
    Reviewed-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Link: https://lore.kernel.org/r/20230531111038.6302-1-francesco@dolcini.it
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c7c37c2994bba9af80a79f51dbaf0ea76db0ee54
Author: K Prateek Nayak <kprateek.nayak@amd.com>
Date:   Mon May 8 14:11:15 2023 +0530

    drivers: base: cacheinfo: Update cpu_map_populated during CPU Hotplug

    [ Upstream commit c26fabe73330d983c7ce822c6b6ec0879b4da61f ]

    Until commit 5c2712387d48 ("cacheinfo: Fix LLC is not exported through
    sysfs"), cacheinfo called populate_cache_leaves() for CPU coming online
    which let the arch specific functions handle (at least on x86)
    populating the shared_cpu_map. However, with the changes in the
    aforementioned commit, populate_cache_leaves() is not called when a CPU
    comes online as a result of hotplug since last_level_cache_is_valid()
    returns true as the cacheinfo data is not discarded. The CPU coming
    online is not present in shared_cpu_map, however, it will not be added
    since the cpu_cacheinfo->cpu_map_populated flag is set (it is set in
    populate_cache_leaves() when cacheinfo is first populated for x86)

    This can lead to inconsistencies in the shared_cpu_map when an offlined
    CPU comes online again. Example below depicts the inconsistency in the
    shared_cpu_list in cacheinfo when CPU8 is offlined and onlined again on
    a 3rd Generation EPYC processor:

      # for i in /sys/devices/system/cpu/cpu8/cache/index*/shared_cpu_list; do echo -n "$i:
"; cat $i; done
        /sys/devices/system/cpu/cpu8/cache/index0/shared_cpu_list: 8,136
        /sys/devices/system/cpu/cpu8/cache/index1/shared_cpu_list: 8,136
        /sys/devices/system/cpu/cpu8/cache/index2/shared_cpu_list: 8,136
        /sys/devices/system/cpu/cpu8/cache/index3/shared_cpu_list: 8-15,136-143

      # echo 0 > /sys/devices/system/cpu/cpu8/online
      # echo 1 > /sys/devices/system/cpu/cpu8/online

      # for i in /sys/devices/system/cpu/cpu8/cache/index*/shared_cpu_list; do echo -n "$i:
"; cat $i; done
        /sys/devices/system/cpu/cpu8/cache/index0/shared_cpu_list: 8
        /sys/devices/system/cpu/cpu8/cache/index1/shared_cpu_list: 8
        /sys/devices/system/cpu/cpu8/cache/index2/shared_cpu_list: 8
        /sys/devices/system/cpu/cpu8/cache/index3/shared_cpu_list: 8

      # cat /sys/devices/system/cpu/cpu136/cache/index0/shared_cpu_list
```

```
        136

      # cat /sys/devices/system/cpu/cpu136/cache/index3/shared_cpu_list
        9-15,136-143

    Clear the flag when the CPU is removed from shared_cpu_map when
    cache_shared_cpu_map_remove() is called during CPU hotplug. This will
    allow cache_shared_cpu_map_setup() to add the CPU coming back online in
    the shared_cpu_map. Set the flag again when the shared_cpu_map is setup.
    Following are results of performing the same test as described above with
    the changes:

      # for i in /sys/devices/system/cpu/cpu8/cache/index*/shared_cpu_list; do echo -n "$i:
"; cat $i; done
        /sys/devices/system/cpu/cpu8/cache/index0/shared_cpu_list: 8,136
        /sys/devices/system/cpu/cpu8/cache/index1/shared_cpu_list: 8,136
        /sys/devices/system/cpu/cpu8/cache/index2/shared_cpu_list: 8,136
        /sys/devices/system/cpu/cpu8/cache/index3/shared_cpu_list: 8-15,136-143

      # echo 0 > /sys/devices/system/cpu/cpu8/online
      # echo 1 > /sys/devices/system/cpu/cpu8/online

      # for i in /sys/devices/system/cpu/cpu8/cache/index*/shared_cpu_list; do echo -n "$i:
"; cat $i; done
        /sys/devices/system/cpu/cpu8/cache/index0/shared_cpu_list: 8,136
        /sys/devices/system/cpu/cpu8/cache/index1/shared_cpu_list: 8,136
        /sys/devices/system/cpu/cpu8/cache/index2/shared_cpu_list: 8,136
        /sys/devices/system/cpu/cpu8/cache/index3/shared_cpu_list: 8-15,136-143

      # cat /sys/devices/system/cpu/cpu136/cache/index0/shared_cpu_list
        8,136

      # cat /sys/devices/system/cpu/cpu136/cache/index3/shared_cpu_list
        8-15,136-143

    Fixes: 5c2712387d48 ("cacheinfo: Fix LLC is not exported through sysfs")
    Signed-off-by: K Prateek Nayak <kprateek.nayak@amd.com>
    Reviewed-by: Yicong Yang <yangyicong@hisilicon.com>
    Reviewed-by: Sudeep Holla <sudeep.holla@arm.com>
    Link: https://lore.kernel.org/r/20230508084115.1157-3-kprateek.nayak@amd.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0fed7242bec2d8b45ad3dbc222a4427d5f68583c
Author: K Prateek Nayak <kprateek.nayak@amd.com>
Date:   Mon May 8 14:11:14 2023 +0530

    drivers: base: cacheinfo: Fix shared_cpu_map changes in event of CPU hotplug

    [ Upstream commit 126310c9f669c9a8c875a3e5c2292299ca90225d ]

    While building the shared_cpu_map, check if the cache level and cache
    type matches. On certain systems that build the cache topology based on
    the instance ID, there are cases where the same ID may repeat across
    multiple cache levels, leading inaccurate topology.

    In event of CPU offlining, the cache_shared_cpu_map_remove() does not
    consider if IDs at same level are being compared. As a result, when same
    IDs repeat across different cache levels, the CPU going offline is not
    removed from all the shared_cpu_map.

    Below is the output of cache topology of CPU8 and it's SMT sibling after
    CPU8 is offlined on a dual socket 3rd Generation AMD EPYC processor
    (2 x 64C/128T) running kernel release v6.3:
```

```
       # for i in /sys/devices/system/cpu/cpu8/cache/index*/shared_cpu_list; do echo -n "$i:
 "; cat $i; done
         /sys/devices/system/cpu/cpu8/cache/index0/shared_cpu_list: 8,136
         /sys/devices/system/cpu/cpu8/cache/index1/shared_cpu_list: 8,136
         /sys/devices/system/cpu/cpu8/cache/index2/shared_cpu_list: 8,136
         /sys/devices/system/cpu/cpu8/cache/index3/shared_cpu_list: 8-15,136-143

       # echo 0 > /sys/devices/system/cpu/cpu8/online

       # for i in /sys/devices/system/cpu/cpu136/cache/index*/shared_cpu_list; do echo -n
 "$i: "; cat $i; done
         /sys/devices/system/cpu/cpu136/cache/index0/shared_cpu_list: 136
         /sys/devices/system/cpu/cpu136/cache/index1/shared_cpu_list: 8,136
         /sys/devices/system/cpu/cpu136/cache/index2/shared_cpu_list: 8,136
         /sys/devices/system/cpu/cpu136/cache/index3/shared_cpu_list: 9-15,136-143

     CPU8 is removed from index0 (L1i) but remains in the shared_cpu_list of
     index1 (L1d) and index2 (L2). Since L1i, L1d, and L2 are shared by the
     SMT siblings, and they have the same cache instance ID, CPU 2 is only
     removed from the first index with matching ID which is index1 (L1i) in
     this case. With this fix, the results are as expected when performing
     the same experiment on the same system:

       # for i in /sys/devices/system/cpu/cpu8/cache/index*/shared_cpu_list; do echo -n "$i:
 "; cat $i; done
         /sys/devices/system/cpu/cpu8/cache/index0/shared_cpu_list: 8,136
         /sys/devices/system/cpu/cpu8/cache/index1/shared_cpu_list: 8,136
         /sys/devices/system/cpu/cpu8/cache/index2/shared_cpu_list: 8,136
         /sys/devices/system/cpu/cpu8/cache/index3/shared_cpu_list: 8-15,136-143

       # echo 0 > /sys/devices/system/cpu/cpu8/online

       # for i in /sys/devices/system/cpu/cpu136/cache/index*/shared_cpu_list; do echo -n
 "$i: "; cat $i; done
         /sys/devices/system/cpu/cpu136/cache/index0/shared_cpu_list: 136
         /sys/devices/system/cpu/cpu136/cache/index1/shared_cpu_list: 136
         /sys/devices/system/cpu/cpu136/cache/index2/shared_cpu_list: 136
         /sys/devices/system/cpu/cpu136/cache/index3/shared_cpu_list: 9-15,136-143

     When rebuilding topology, the same problem appears as
     cache_shared_cpu_map_setup() implements a similar logic. Consider the
     same 3rd Generation EPYC processor: CPUs in Core 1, that share the L1
     and L2 caches, have L1 and L2 instance ID as 1. For all the CPUs on
     the second chiplet, the L3 ID is also 1 leading to grouping on CPUs from
     Core 1 (1, 17) and the entire second chiplet (8-15, 24-31) as CPUs
     sharing one cache domain. This went undetected since x86 processors
     depended on arch specific populate_cache_leaves() method to repopulate
     the shared_cpus_map when CPU came back online until kernel release
     v6.3-rc5.

     Fixes: 198102c9103f ("cacheinfo: Fix shared_cpu_map to handle shared caches at different
 levels")
     Signed-off-by: K Prateek Nayak <kprateek.nayak@amd.com>
     Reviewed-by: Sudeep Holla <sudeep.holla@arm.com>
     Link: https://lore.kernel.org/r/20230508084115.1157-2-kprateek.nayak@amd.com
     Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit c51b76e7b5c9a66206ba000f4b971184ee338cee
 Author: Dan Carpenter <dan.carpenter@linaro.org>
 Date:   Fri May 5 12:22:09 2023 +0300

     mailbox: mailbox-test: fix a locking issue in mbox_test_message_write()

     [ Upstream commit 8fe72b76db79d694858e872370df49676bc3be8c ]
```

```
    There was a bug where this code forgot to unlock the tdev->mutex if the
    kzalloc() failed.  Fix this issue, by moving the allocation outside the
    lock.

    Fixes: 2d1e952a2b8e ("mailbox: mailbox-test: Fix potential double-free in
mbox_test_message_write()")
    Signed-off-by: Dan Carpenter <dan.carpenter@linaro.org>
    Reviewed-by: Lee Jones <lee@kernel.org>
    Signed-off-by: Jassi Brar <jaswinder.singh@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 845007332b838029e8569eab6cf90d60ed8fb51c
Author: Pin-yen Lin <treapking@chromium.org>
Date:   Sat Apr 22 11:39:05 2023 +0100

    media: mediatek: vcodec: Only apply 4K frame sizes on decoder formats

    [ Upstream commit ed17f89e9502f03af493e130620a9bb74c07cf28 ]

    When VCODEC_CAPABILITY_4K_DISABLED is not set in dec_capability, skip
    formats that are not MTK_FMT_DEC so only decoder formats is updated in
    mtk_init_vdec_params.

    Fixes: e25528e1dbe5 ("media: mediatek: vcodec: Use 4K frame size when supported by
stateful decoder")
    Signed-off-by: Pin-yen Lin <treapking@chromium.org>
    Reviewed-by: Chen-Yu Tsai <wenst@chromium.org>
    Reviewed-by: Yunfei Dong <yunfei.dong@mediatek.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6322c939b84f4ec7b97c6b3e014f9bfe71da7ab7
Author: Fuad Tabba <tabba@google.com>
Date:   Mon May 22 11:32:58 2023 +0100

    KVM: arm64: Reload PTE after invoking walker callback on preorder traversal

    [ Upstream commit a9f0e3d5a089d0844abb679a5e99f15010d53e25 ]

    The preorder callback on the kvm_pgtable_stage2_map() path can replace
    a table with a block, then recursively free the detached table. The
    higher-level walking logic stashes the old page table entry and
    then walks the freed table, invoking the leaf callback and
    potentially freeing pgtable pages prematurely.

    In normal operation, the call to tear down the detached stage-2
    is indirected and uses an RCU callback to trigger the freeing.
    RCU is not available to pKVM, which is where this bug is
    triggered.

    Change the behavior of the walker to reload the page table entry
    after invoking the walker callback on preorder traversal, as it
    does for leaf entries.

    Tested on Pixel 6.

    Fixes: 5c359cca1faf ("KVM: arm64: Tear down unlinked stage-2 subtree after break-before-
make")
    Suggested-by: Oliver Upton <oliver.upton@linux.dev>
    Signed-off-by: Fuad Tabba <tabba@google.com>
    Signed-off-by: Marc Zyngier <maz@kernel.org>
    Link: https://lore.kernel.org/r/20230522103258.402272-1-tabba@google.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 0cf760b5c284cddbb68a079d7b1db6a95c379538
Author: Like Xu <likexu@tencent.com>
Date:   Wed May 17 21:38:08 2023 +0800

    perf/x86/intel: Save/restore cpuc->active_pebs_data_cfg when using guest PEBS

    [ Upstream commit 3c845304d2d723f20d5b91fef5d133ff94825d76 ]

    After commit b752ea0c28e3 ("perf/x86/intel/ds: Flush PEBS DS when changing
    PEBS_DATA_CFG"), the cpuc->pebs_data_cfg may save some bits that are not
    supported by real hardware, such as PEBS_UPDATE_DS_SW. This would cause
    the VMX hardware MSR switching mechanism to save/restore invalid values
    for PEBS_DATA_CFG MSR, thus crashing the host when PEBS is used for guest.
    Fix it by using the active host value from cpuc->active_pebs_data_cfg.

    Fixes: b752ea0c28e3 ("perf/x86/intel/ds: Flush PEBS DS when changing PEBS_DATA_CFG")
    Signed-off-by: Like Xu <likexu@tencent.com>
    Signed-off-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Reviewed-by: Kan Liang <kan.liang@linux.intel.com>
    Link: https://lore.kernel.org/r/20230517133808.67885-1-likexu@tencent.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 33f1ed7f07b03742a11b64399fb591fda9e90779
Author: Gleb Chesnokov <gleb.chesnokov@scst.dev>
Date:   Wed May 17 11:22:35 2023 +0300

    scsi: qla2xxx: Fix NULL pointer dereference in target mode

    [ Upstream commit d54820b22e404b06b2b65877ff802cc7b31688bc ]

    When target mode is enabled, the pci_irq_get_affinity() function may return
    a NULL value in qla_mapq_init_qp_cpu_map() due to the qla24xx_enable_msix()
    code that handles IRQ settings for target mode. This leads to a crash due
    to a NULL pointer dereference.

    This patch fixes the issue by adding a check for the NULL value returned by
    pci_irq_get_affinity() and introducing a 'cpu_mapped' boolean flag to the
    qla_qpair structure, ensuring that the qpair's CPU affinity is updated when
    it has not been mapped to a CPU.

    Fixes: 1d201c81d4cc ("scsi: qla2xxx: Select qpair depending on which CPU post_cmd() gets
called")
    Signed-off-by: Gleb Chesnokov <gleb.chesnokov@scst.dev>
    Link: https://lore.kernel.org/r/56b416f2-4e0f-b6cf-d6d5-b7c372e3c6a2@scst.dev
    Reviewed-by: Himanshu Madhani <himanshu.madhani@oracle.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c3cd33abe63f0ea32c3966ae67a7efc48e86c3e0
Author: Will Deacon <will@kernel.org>
Date:   Thu May 18 10:58:44 2023 +0100

    KVM: arm64: Prevent unconditional donation of unmapped regions from the host

    [ Upstream commit 09cce60bddd6461a93a5bf434265a47827d1bc6f ]

    Since host stage-2 mappings are created lazily, we cannot rely solely on
    the pte in order to recover the target physical address when checking a
    host-initiated memory transition as this permits donation of unmapped
    regions corresponding to MMIO or "no-map" memory.

    Instead of inspecting the pte, move the addr_is_allowed_memory() check
    into the host callback function where it is passed the physical address
    directly from the walker.
```

```
    Cc: Quentin Perret <qperret@google.com>
    Fixes: e82edcc75c4e ("KVM: arm64: Implement do_share() helper for sharing memory")
    Signed-off-by: Will Deacon <will@kernel.org>
    Signed-off-by: Marc Zyngier <maz@kernel.org>
    Link: https://lore.kernel.org/r/20230518095844.1178-1-will@kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b47ce8899bfab95f6dfd4b4686b3588f3b9114d5
Author: Jean-Philippe Brucker <jean-philippe@linaro.org>
Date:   Thu May 18 11:09:17 2023 +0100

    KVM: arm64: vgic: Fix locking comment

    [ Upstream commit c38b8400aef99d63be2b1ff131bb993465dcafe1 ]

    It is now config_lock that must be held, not kvm lock. Replace the
    comment with a lockdep annotation.

    Fixes: f00327731131 ("KVM: arm64: Use config_lock to protect vgic state")
    Signed-off-by: Jean-Philippe Brucker <jean-philippe@linaro.org>
    Reviewed-by: Oliver Upton <oliver.upton@linux.dev>
    Signed-off-by: Marc Zyngier <maz@kernel.org>
    Link: https://lore.kernel.org/r/20230518100914.2837292-4-jean-philippe@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit bcfe1c9649a2add51a5615ec18b360449a0ceaf4
Author: Jean-Philippe Brucker <jean-philippe@linaro.org>
Date:   Thu May 18 11:09:16 2023 +0100

    KVM: arm64: vgic: Wrap vgic_its_create() with config_lock

    [ Upstream commit 9cf2f840c439b6b23bd99f584f2917ca425ae406 ]

    vgic_its_create() changes the vgic state without holding the
    config_lock, which triggers a lockdep warning in vgic_v4_init():

    [  358.667941] WARNING: CPU: 3 PID: 178 at arch/arm64/kvm/vgic/vgic-v4.c:245
vgic_v4_init+0x15c/0x7a8
    ...
    [  358.707410]  vgic_v4_init+0x15c/0x7a8
    [  358.708550]  vgic_its_create+0x37c/0x4a4
    [  358.709640]  kvm_vm_ioctl+0x1518/0x2d80
    [  358.710688]  __arm64_sys_ioctl+0x7ac/0x1ba8
    [  358.711960]  invoke_syscall.constprop.0+0x70/0x1e0
    [  358.713245]  do_el0_svc+0xe4/0x2d4
    [  358.714289]  el0_svc+0x44/0x8c
    [  358.715329]  el0t_64_sync_handler+0xf4/0x120
    [  358.716615]  el0t_64_sync+0x190/0x194

    Wrap the whole of vgic_its_create() with config_lock since, in addition
    to calling vgic_v4_init(), it also modifies the global kvm->arch.vgic
    state.

    Fixes: f00327731131 ("KVM: arm64: Use config_lock to protect vgic state")
    Signed-off-by: Jean-Philippe Brucker <jean-philippe@linaro.org>
    Reviewed-by: Oliver Upton <oliver.upton@linux.dev>
    Signed-off-by: Marc Zyngier <maz@kernel.org>
    Link: https://lore.kernel.org/r/20230518100914.2837292-3-jean-philippe@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit db0d5f3fd1c63fef37cd98a9cad82f3c82b480f4
Author: Jean-Philippe Brucker <jean-philippe@linaro.org>
Date:   Thu May 18 11:09:15 2023 +0100
```

```
KVM: arm64: vgic: Fix a circular locking issue

[ Upstream commit 59112e9c390be595224e427827475a6cd3726021 ]

Lockdep reports a circular lock dependency between the srcu and the
config_lock:

[  262.179917] -> #1 (&kvm->srcu){.+.+}-{0:0}:
[  262.182010]        __synchronize_srcu+0xb0/0x224
[  262.183422]        synchronize_srcu_expedited+0x24/0x34
[  262.184554]        kvm_io_bus_register_dev+0x324/0x50c
[  262.185650]        vgic_register_redist_iodev+0x254/0x398
[  262.186740]        vgic_v3_set_redist_base+0x3b0/0x724
[  262.188087]        kvm_vgic_addr+0x364/0x600
[  262.189189]        vgic_set_common_attr+0x90/0x544
[  262.190278]        vgic_v3_set_attr+0x74/0x9c
[  262.191432]        kvm_device_ioctl+0x2a0/0x4e4
[  262.192515]        __arm64_sys_ioctl+0x7ac/0x1ba8
[  262.193612]        invoke_syscall.constprop.0+0x70/0x1e0
[  262.195006]        do_el0_svc+0xe4/0x2d4
[  262.195929]        el0_svc+0x44/0x8c
[  262.196917]        el0t_64_sync_handler+0xf4/0x120
[  262.198238]        el0t_64_sync+0x190/0x194
[  262.199224]
[  262.199224] -> #0 (&kvm->arch.config_lock){+.+.}-{3:3}:
[  262.201094]        __lock_acquire+0x2b70/0x626c
[  262.202245]        lock_acquire+0x454/0x778
[  262.203132]        __mutex_lock+0x190/0x8b4
[  262.204023]        mutex_lock_nested+0x24/0x30
[  262.205100]        vgic_mmio_write_v3_misc+0x5c/0x2a0
[  262.206178]        dispatch_mmio_write+0xd8/0x258
[  262.207498]        __kvm_io_bus_write+0x1e0/0x350
[  262.208582]        kvm_io_bus_write+0xe0/0x1cc
[  262.209653]        io_mem_abort+0x2ac/0x6d8
[  262.210569]        kvm_handle_guest_abort+0x9b8/0x1f88
[  262.211937]        handle_exit+0xc4/0x39c
[  262.212971]        kvm_arch_vcpu_ioctl_run+0x90c/0x1c04
[  262.214154]        kvm_vcpu_ioctl+0x450/0x12f8
[  262.215233]        __arm64_sys_ioctl+0x7ac/0x1ba8
[  262.216402]        invoke_syscall.constprop.0+0x70/0x1e0
[  262.217774]        do_el0_svc+0xe4/0x2d4
[  262.218758]        el0_svc+0x44/0x8c
[  262.219941]        el0t_64_sync_handler+0xf4/0x120
[  262.221110]        el0t_64_sync+0x190/0x194

Note that the current report, which can be triggered by the vgic_irq
kselftest, is a triple chain that includes slots_lock, but after
inverting the slots_lock/config_lock dependency, the actual problem
reported above remains.

In several places, the vgic code calls kvm_io_bus_register_dev(), which
synchronizes the srcu, while holding config_lock (#1). And the MMIO
handler takes the config_lock while holding the srcu read lock (#0).

Break dependency #1, by registering the distributor and redistributors
without holding config_lock. The ITS also uses kvm_io_bus_register_dev()
but already relies on slots_lock to serialize calls.

The distributor iodev is created on the first KVM_RUN call. Multiple
threads will race for vgic initialization, and only the first one will
see !vgic_ready() under the lock. To serialize those threads, rely on
slots_lock rather than config_lock.

Redistributors are created earlier, through KVM_DEV_ARM_VGIC_GRP_ADDR
ioctls and vCPU creation. Similarly, serialize the iodev creation with
```

```
    slots_lock, and the rest with config_lock.

    Fixes: f00327731131 ("KVM: arm64: Use config_lock to protect vgic state")
    Signed-off-by: Jean-Philippe Brucker <jean-philippe@linaro.org>
    Reviewed-by: Oliver Upton <oliver.upton@linux.dev>
    Signed-off-by: Marc Zyngier <maz@kernel.org>
    Link: https://lore.kernel.org/r/20230518100914.2837292-2-jean-philippe@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 635383e95ee21260a4ca0a83d1637146ec73faf4
Author: Dan Carpenter <error27@gmail.com>
Date:   Tue Feb 14 18:47:30 2023 +0300

    iio: adc: imx93: fix a signedness bug in imx93_adc_read_raw()

    [ Upstream commit 20f291b88ecf23f674ee2ed980a4d93b7f16a06f ]

    The problem is these lines:

            ret = vref_uv = regulator_get_voltage(adc->vref);
            if (ret < 0)

    The "ret" variable is type long and "vref_uv" is u32 so that means
    the condition can never be true on a 64bit system.  A negative error
    code from regulator_get_voltage() would be cast to a high positive
    u32 value and then remain a high positive value when cast to a long.

    The "ret" variable only ever stores ints so it should be declared as
    an int.  We can delete the "vref_uv" variable and use "ret" directly.

    Fixes: 7d02296ac8b8 ("iio: adc: add imx93 adc support")
    Signed-off-by: Dan Carpenter <error27@gmail.com>
    Reviewed-by: Haibo Chen <haibo.chen@nxp.com>
    Link: https://lore.kernel.org/r/Y+utEvjfjQRQo2QB@kili
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 53c03189cafd19219ce16ce4eaa01f1f30e4e552
Author: Loic Poulain <loic.poulain@linaro.org>
Date:   Wed May 10 09:42:23 2023 +0200

    block: Deny writable memory mapping if block is read-only

    [ Upstream commit 69baa3a623fd2e58624f24f2f23d46f87b817c93 ]

    User should not be able to write block device if it is read-only at
    block level (e.g force_ro attribute). This is ensured in the regular
    fops write operation (blkdev_write_iter) but not when writing via
    user mapping (mmap), allowing user to actually write a read-only
    block device via a PROT_WRITE mapping.

    Example: This can lead to integrity issue of eMMC boot partition
    (e.g mmcblk0boot0) which is read-only by default.

    To fix this issue, simply deny shared writable mapping if the block
    is readonly.

    Note: Block remains writable if switch to read-only is performed
    after the initial mapping, but this is expected behavior according
    to commit a32e236eb93e ("Partially revert "block: fail op_is_write()
    requests to read-only partitions"")'.

    Signed-off-by: Loic Poulain <loic.poulain@linaro.org>
    Reviewed-by: Christoph Hellwig <hch@lst.de>
    Link: https://lore.kernel.org/r/20230510074223.991297-1-loic.poulain@linaro.org
```

```
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8c8095907cd391325b4a75d90e5031d519c8d3a0
Author: Daniel Smith <dansmith@ds.gy>
Date:   Wed May 17 14:32:32 2023 -0700

    nvme-pci: Add quirk for Teamgroup MP33 SSD

    [ Upstream commit 0649728123cf6a5518e154b4e1735fc85ea4f55c ]

    Add a quirk for Teamgroup MP33 that reports duplicate ids for disk.

    Signed-off-by: Daniel Smith <dansmith@ds.gy>
    [kch: patch formatting]
    Signed-off-by: Chaitanya Kulkarni <kch@nvidia.com>
    Tested-by: Daniel Smith <dansmith@ds.gy>
    Signed-off-by: Keith Busch <kbusch@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 936048a565bcdf747770eaf0ca6ebe4df374a119
Author: Ming Lei <ming.lei@redhat.com>
Date:   Wed May 17 21:34:08 2023 +0800

    ublk: fix AB-BA lockdep warning

    [ Upstream commit ac5902f84bb546c64aea02c439c2579cbf40318f ]

    When handling UBLK_IO_FETCH_REQ, ctx->uring_lock is grabbed first, then
    ub->mutex is acquired.

    When handling UBLK_CMD_STOP_DEV or UBLK_CMD_DEL_DEV, ub->mutex is
    grabbed first, then calling io_uring_cmd_done() for canceling uring
    command, in which ctx->uring_lock may be required.

    Real deadlock only happens when all the above commands are issued from
    same uring context, and in reality different uring contexts are often used
    for handing control command and IO command.

    Fix the issue by using io_uring_cmd_complete_in_task() to cancel command
    in ublk_cancel_dev(ublk_cancel_queue).

    Reported-by: Shinichiro Kawasaki <shinichiro.kawasaki@wdc.com>
    Closes: https://lore.kernel.org/linux-
block/becol2g7sawl4rsjq2dztsbc7mqypfqko6wzsyoyazqydoasml@rcxarzwidrhk
    Cc: Ziyang Zhang <ZiyangZhang@linux.alibaba.com>
    Signed-off-by: Ming Lei <ming.lei@redhat.com>
    Tested-by: Shinichiro Kawasaki <shinichiro.kawasaki@wdc.com>
    Link: https://lore.kernel.org/r/20230517133408.210944-1-ming.lei@redhat.com
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2e02dcd0c3ec878473165177c9ec9feecec0aa70
Author: Guchun Chen <guchun.chen@amd.com>
Date:   Tue May 9 16:15:27 2023 +0800

    drm/amdgpu: skip disabling fence driver src_irqs when device is unplugged

    [ Upstream commit c1a322a7a4a96cd0a3dde32ce37af437a78bf8cd ]

    When performing device unbind or halt, we have disabled all irqs at the
    very begining like amdgpu_pci_remove or amdgpu_device_halt. So
    amdgpu_irq_put for irqs stored in fence driver should not be called
    any more, otherwise, below calltrace will arrive.
```

```
      [  139.114088] WARNING: CPU: 2 PID: 1550 at drivers/gpu/drm/amd/amdgpu/amdgpu_irq.c:616
  amdgpu_irq_put+0xf6/0x110 [amdgpu]
      [  139.114655] Call Trace:
      [  139.114655]  <TASK>
      [  139.114657]  amdgpu_fence_driver_hw_fini+0x93/0x130 [amdgpu]
      [  139.114836]  amdgpu_device_fini_hw+0xb6/0x350 [amdgpu]
      [  139.114955]  amdgpu_driver_unload_kms+0x51/0x70 [amdgpu]
      [  139.115075]  amdgpu_pci_remove+0x63/0x160 [amdgpu]
      [  139.115193]  ? __pm_runtime_resume+0x64/0x90
      [  139.115195]  pci_device_remove+0x3a/0xb0
      [  139.115197]  device_remove+0x43/0x70
      [  139.115198]  device_release_driver_internal+0xbd/0x140

      Signed-off-by: Guchun Chen <guchun.chen@amd.com>
      Acked-by: Alex Deucher <alexander.deucher@amd.com>
      Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit a7a50fd517cb9e0e0c409664540f8bdcf3b20cff
  Author: Xiubo Li <xiubli@redhat.com>
  Date:   Mon May 8 14:45:01 2023 +0800

      ceph: silence smatch warning in reconnect_caps_cb()

      [ Upstream commit 9aaa7eb018661b2da221362d9bacb096bd596f52 ]

      Smatch static checker warning:

        fs/ceph/mds_client.c:3968 reconnect_caps_cb()
        warn: missing error code here? '__get_cap_for_mds()' failed. 'err' = '0'

      [ idryomov: Dan says that Smatch considers it intentional only if the
        "ret = 0;" assignment is within 4 or 5 lines of the goto. ]

      Reported-by: Dan Carpenter <dan.carpenter@linaro.org>
      Signed-off-by: Xiubo Li <xiubli@redhat.com>
      Reviewed-by: Ilya Dryomov <idryomov@gmail.com>
      Signed-off-by: Ilya Dryomov <idryomov@gmail.com>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 836e2583cdf25735beb2bf8b3066e6a84a8cb4fe
  Author: Arnd Bergmann <arnd@arndb.de>
  Date:   Tue May 16 21:45:34 2023 +0200

      atm: hide unused procfs functions

      [ Upstream commit fb1b7be9b16c1f4626969ba4e95a97da2a452b41 ]

      When CONFIG_PROC_FS is disabled, the function declarations for some
      procfs functions are hidden, but the definitions are still build,
      as shown by this compiler warning:

      net/atm/resources.c:403:7: error: no previous prototype for 'atm_dev_seq_start' [-
  Werror=missing-prototypes]
      net/atm/resources.c:409:6: error: no previous prototype for 'atm_dev_seq_stop' [-
  Werror=missing-prototypes]
      net/atm/resources.c:414:7: error: no previous prototype for 'atm_dev_seq_next' [-
  Werror=missing-prototypes]

      Add another #ifdef to leave these out of the build.

      Signed-off-by: Arnd Bergmann <arnd@arndb.de>
      Link: https://lore.kernel.org/r/20230516194625.549249-2-arnd@kernel.org
      Signed-off-by: Jakub Kicinski <kuba@kernel.org>
      Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit bfe752b1372ca53ed7a3523a0093cca3e0098aac
Author: Rob Clark <robdclark@chromium.org>
Date:   Tue May 16 15:20:37 2023 -0700

    drm/msm: Be more shouty if per-process pgtables aren't working

    [ Upstream commit 5c054db54c43a5fcb5cc81012361f5e3fac37637 ]

    Otherwise it is not always obvious if a dt or iommu change is causing us
    to fall back to global pgtable.

    Signed-off-by: Rob Clark <robdclark@chromium.org>
    Reviewed-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Patchwork: https://patchwork.freedesktop.org/patch/537359/
    Link: https://lore.kernel.org/r/20230516222039.907690-2-robdclark@gmail.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e3a60abab72220a502ac88b56287058adbf0e23f
Author: Arnd Bergmann <arnd@arndb.de>
Date:   Tue May 16 21:50:42 2023 +0200

    ALSA: oss: avoid missing-prototype warnings

    [ Upstream commit 040b5a046a9e18098580d3ccd029e2318fca7859 ]

    Two functions are defined and used in pcm_oss.c but also optionally
    used from io.c, with an optional prototype. If CONFIG_SND_PCM_OSS_PLUGINS
    is disabled, this causes a warning as the functions are not static
    and have no prototype:

    sound/core/oss/pcm_oss.c:1235:19: error: no previous prototype for 'snd_pcm_oss_write3'
[-Werror=missing-prototypes]
    sound/core/oss/pcm_oss.c:1266:19: error: no previous prototype for 'snd_pcm_oss_read3'
[-Werror=missing-prototypes]

    Avoid this by making the prototypes unconditional.

    Signed-off-by: Arnd Bergmann <arnd@arndb.de>
    Link: https://lore.kernel.org/r/20230516195046.550584-2-arnd@kernel.org
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e3d79676be2e69e5504a9c8895db30116b76b85b
Author: Maurizio Lombardi <mlombard@redhat.com>
Date:   Thu May 11 13:07:41 2023 +0200

    nvme: do not let the user delete a ctrl before a complete initialization

    [ Upstream commit 2eb94dd56a4a4e3fe286def3e2ba207804a37345 ]

    If a userspace application performes a "delete_controller" command
    early during the ctrl initialization, the delete operation
    may race against the init code and the kernel will crash.

    nvme nvme5: Connect command failed: host path error
    nvme nvme5: failed to connect queue: 0 ret=880
    PF: supervisor write access in kernel mode
    PF: error_code(0x0002) - not-present page
     blk_mq_quiesce_queue+0x18/0x90
     nvme_tcp_delete_ctrl+0x24/0x40 [nvme_tcp]
     nvme_do_delete_ctrl+0x7f/0x8b [nvme_core]
     nvme_sysfs_delete.cold+0x8/0xd [nvme_core]
     kernfs_fop_write_iter+0x124/0x1b0
     new_sync_write+0xff/0x190
```

```
       vfs_write+0x1ef/0x280

    Fix the crash by checking the NVME_CTRL_STARTED_ONCE bit;
    if it's not set it means that the nvme controller is still
    in the process of getting initialized and the kernel
    will return an -EBUSY error to userspace.
    Set the NVME_CTRL_STARTED_ONCE later in the nvme_start_ctrl()
    function, after the controller start operation is completed.

    Signed-off-by: Maurizio Lombardi <mlombard@redhat.com>
    Reviewed-by: Sagi Grimberg <sagi@grimberg.me>
    Reviewed-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Keith Busch <kbusch@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4727533dc80effa81faf233ed0c29beebf6f7e14
Author: Christoph Hellwig <hch@lst.de>
Date:   Wed May 17 09:53:45 2023 +0200

    nvme-multipath: don't call blk_mark_disk_dead in nvme_mpath_remove_disk

    [ Upstream commit 1743e5f6000901a11f4e1cd741bfa9136f3ec9b1 ]

    nvme_mpath_remove_disk is called after del_gendisk, at which point a
    blk_mark_disk_dead call doesn't make any sense.

    Signed-off-by: Christoph Hellwig <hch@lst.de>
    Reviewed-by: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Keith Busch <kbusch@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5aa1dbfd789fa8aa54284fe72fda651d0543e50c
Author: Tom Rix <trix@redhat.com>
Date:   Sun May 14 10:00:10 2023 -0400

    netfilter: conntrack: define variables exp_nat_nla_policy and any_addr with
CONFIG_NF_NAT

    [ Upstream commit 224a876e37543eee111bf9b6aa4935080e619335 ]

    gcc with W=1 and ! CONFIG_NF_NAT
    net/netfilter/nf_conntrack_netlink.c:3463:32: error:
      ‘exp_nat_nla_policy’ defined but not used [-Werror=unused-const-variable=]
     3463 | static const struct nla_policy exp_nat_nla_policy[CTA_EXPECT_NAT_MAX+1] = {
          |                                ^~~~~~~~~~~~~~~~~~
    net/netfilter/nf_conntrack_netlink.c:2979:33: error:
      ‘any_addr’ defined but not used [-Werror=unused-const-variable=]
     2979 | static const union nf_inet_addr any_addr;
          |                                 ^~~~~~~~

    These variables use is controlled by CONFIG_NF_NAT, so should their definitions.

    Signed-off-by: Tom Rix <trix@redhat.com>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Signed-off-by: Florian Westphal <fw@strlen.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 439ddc8a1abbd38b8dde22fcd7749ac95d34fce2
Author: Kai-Heng Feng <kai.heng.feng@canonical.com>
Date:   Wed May 17 13:24:51 2023 +0800

    net: wwan: t7xx: Ensure init is completed before system sleep

    [ Upstream commit ab87603b251134441a67385ecc9d3371be17b7a7 ]
```

```
    When the system attempts to sleep while mtk_t7xx is not ready, the driver
    cannot put the device to sleep:
    [   12.472918] mtk_t7xx 0000:57:00.0: [PM] Exiting suspend, modem in invalid state
    [   12.472936] mtk_t7xx 0000:57:00.0: PM: pci_pm_suspend(): t7xx_pci_pm_suspend+0x0/0x20
[mtk_t7xx] returns -14
    [   12.473678] mtk_t7xx 0000:57:00.0: PM: dpm_run_callback(): pci_pm_suspend+0x0/0x1b0
returns -14
    [   12.473711] mtk_t7xx 0000:57:00.0: PM: failed to suspend async: error -14
    [   12.764776] PM: Some devices failed to suspend, or early wake event detected

    Mediatek confirmed the device can take a rather long time to complete
    its initialization, so wait for up to 20 seconds until init is done.

    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 bc77ca0bc949e8d997d7fa43d56885e55f87107d
Author: Arnd Bergmann <arnd@arndb.de>
Date:   Tue May 16 20:34:22 2023 +0200

    wifi: b43: fix incorrect __packed annotation

    [ Upstream commit 212457ccbd60dba34f965e4ffbe62f0e4f970538 ]

    clang warns about an unpacked structure inside of a packed one:

    drivers/net/wireless/broadcom/b43/b43.h:654:4: error: field data within 'struct b43_iv'
is less aligned than 'union (unnamed union at /home/arnd/arm-
soc/drivers/net/wireless/broadcom/b43/b43.h:651:2)' and is usually due to 'struct b43_iv'
being packed, which can lead to unaligned accesses [-Werror,-Wunaligned-access]

    The problem here is that the anonymous union has the default alignment
    from its members, apparently because the original author mixed up the
    placement of the __packed attribute by placing it next to the struct
    member rather than the union definition. As the struct itself is
    also marked as __packed, there is no need to mark its members, so just
    move the annotation to the inner type instead.

    As Michael noted, the same problem is present in b43legacy, so
    change both at the same time.

    Acked-by: Michael Büsch <m@bues.ch>
    Reported-by: kernel test robot <lkp@intel.com>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Tested-by: Larry Finger <Larry.Finger@lwfinger.net>
    Link: https://lore.kernel.org/oe-kbuild-all/202305160749.ay1HAoyP-lkp@intel.com/
    Signed-off-by: Arnd Bergmann <arnd@arndb.de>
    Signed-off-by: Kalle Valo <kvalo@kernel.org>
    Link: https://lore.kernel.org/r/20230516183442.536589-1-arnd@kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e07d70177df7a117455c93e68b15004646d299ca
Author: Wenchao Hao <haowenchao2@huawei.com>
Date:   Mon May 15 15:01:56 2023 +0800

    scsi: core: Decrease scsi_device's iorequest_cnt if dispatch failed

    [ Upstream commit 09e797c8641f6ad435c33ae24c223351197ea29a ]

    If scsi_dispatch_cmd() failed, the SCSI command was not sent to the target,
    scsi_queue_rq() would return BLK_STS_RESOURCE and the related request would
    be requeued. The timeout of this request would not fire, no one would
    increase iodone_cnt.
```

```
    The above flow would result the iodone_cnt smaller than iorequest_cnt.  So
    decrease the iorequest_cnt if dispatch failed to workaround the issue.

    Signed-off-by: Wenchao Hao <haowenchao2@huawei.com>
    Reported-by: Ming Lei <ming.lei@redhat.com>
    Closes: https://lore.kernel.org/r/ZF+zB+bB7iqe0wGd@ovpn-8-17.pek2.redhat.com
    Link: https://lore.kernel.org/r/20230515070156.1790181-3-haowenchao2@huawei.com
    Reviewed-by: Ming Lei <ming.lei@redhat.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b1e65473a717725045aad378cc8d824b3be4351b
Author: Po-Wen Kao <powen.kao@mediatek.com>
Date:   Thu May 4 23:44:51 2023 +0800

    scsi: ufs: core: Fix MCQ nr_hw_queues

    [ Upstream commit 72a81bb0b6fc9b759ac0fdaca3ec5884a8b2f304 ]

    Since MAXQ is 0-based value, add one to obtain number of hardware queues.

    Signed-off-by: Po-Wen Kao <powen.kao@mediatek.com>
    Link: https://lore.kernel.org/r/20230504154454.26654-4-powen.kao@mediatek.com
    Reviewed-by: Bean Huo <beanhuo@micron.com>
    Reviewed-by: Manivannan Sadhasivam <mani@kernel.org>
    Reviewed-by: Bart Van Assche <bvanassche@acm.org>
    Reviewed-by: Stanley Chu <stanley.chu@mediatek.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c953c5377cf3153fca31cd007755659f4d186c0e
Author: Po-Wen Kao <powen.kao@mediatek.com>
Date:   Thu May 4 23:44:50 2023 +0800

    scsi: ufs: core: Rename symbol sizeof_utp_transfer_cmd_desc()

    [ Upstream commit 06caeb536b2b21668efd2d6fa97c09461957b3a7 ]

    Naming the functions after standard operators like sizeof() may cause
    confusion. Rename it to ufshcd_get_ucd_size().

    Signed-off-by: Po-Wen Kao <powen.kao@mediatek.com>
    Link: https://lore.kernel.org/r/20230504154454.26654-3-powen.kao@mediatek.com
    Suggested-by: Manivannan Sadhasivam <mani@kernel.org>
    Reviewed-by: Stanley Chu <stanley.chu@mediatek.com>
    Reviewed-by: Ziqi Chen <quic_ziqichen@quicinc.com>
    Reviewed-by: Bart Van Assche <bvanassche@acm.org>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 38a898c2e6c9d74882096793640e252af97ea3ab
Author: Po-Wen Kao <powen.kao@mediatek.com>
Date:   Thu May 4 23:44:49 2023 +0800

    scsi: ufs: core: Fix MCQ tag calculation

    [ Upstream commit 5149452ca66289ef33d13897ee845a2f6f5b680f ]

    The transfer command descriptor is allocated in ufshcd_memory_alloc() and
    referenced by the transfer request descriptor with stride size
    sizeof_utp_transfer_cmd_desc() instead of sizeof(struct
    utp_transfer_cmd_desc).

    Consequently, computing tag by address offset should also refer to the
    same stride.
```

```
    Signed-off-by: Po-Wen Kao <powen.kao@mediatek.com>
    Link: https://lore.kernel.org/r/20230504154454.26654-2-powen.kao@mediatek.com
    Reviewed-by: Bart Van Assche <bvanassche@acm.org>
    Reviewed-by: Manivannan Sadhasivam <mani@kernel.org>
    Reviewed-by: Stanley Chu <stanley.chu@mediatek.com>
    Reviewed-by: Ziqi Chen <quic_ziqichen@quicinc.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 26213aebd928712e97ec298f854884292e08f820
Author: Ariel Malamud <ariel.malamud@intel.com>
Date:   Sun May 14 12:15:55 2023 +0300

    wifi: iwlwifi: mvm: Add locking to the rate read flow

    [ Upstream commit a8938bc881d2a03f9b77f19fae924fe798a01285 ]

    The rs_drv_get_rate flow reads the lq_sta to return the optimal rate
    for tx frames. This read flow is not protected thereby leaving
    a small window, a few instructions wide, open to contention by an
    asynchronous rate update. Indeed this race condition was hit and the
    update occurred in the middle of the read.

    Fix this by locking the lq_sta struct during read.

    Signed-off-by: Ariel Malamud <ariel.malamud@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link:
https://lore.kernel.org/r/20230514120631.b52c9ed5c379.I15290b78e0d966c1b68278263776ca9de841d
5fe@changeid
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8adacbeafe6a5b7460c787ecf3a6ea91c0b6131a
Author: Johannes Berg <johannes.berg@intel.com>
Date:   Thu May 4 16:45:04 2023 +0300

    wifi: mac80211: recalc chanctx mindef before assigning

    [ Upstream commit 04312de4ced4b152749614e8179f3978a20a992f ]

    When we allocate a new channel context, or find an existing one
    that is compatible, we currently assign it to a link before its
    mindef is updated. This leads to strange situations, especially
    in link switching where you switch to an 80 MHz link and expect
    it to be active immediately, but the mindef is still configured
    to 20 MHz while assigning.  Also, it's strange that the chandef
    passed to the assign method's argument is wider than the one in
    the context.

    Fix this by calculating the mindef with the new link considered
    before calling the driver.

    In particular, this fixes an iwlwifi problem during link switch
    where the firmware would assert because the (link) station that
    was added for the AP is configured to transmit at a bandwidth
    that's wider than the channel context that it's configured on.

    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link: https://lore.kernel.org/r/20230504134511.828474-5-gregory.greenman@intel.com
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 6b07ac04c30e7f9d613656115c1a77b269a9451d
Author: Johannes Berg <johannes.berg@intel.com>
Date:   Thu May 4 16:45:03 2023 +0300

    wifi: mac80211: consider reserved chanctx for mindef

    [ Upstream commit b72a455a2409fd94d6d9b4eb51d659a88213243b ]

    When a chanctx is reserved for a new vif and we recalculate
    the minimal definition for it, we need to consider the new
    interface it's being reserved for before we assign it, so it
    can be used directly with the correct min channel width.

    Fix the code to - optionally - consider that, and use that
    option just before doing the reassignment.

    Also, when considering channel context reservations, we
    should only consider the one link we're currently working with.
    Change the boolean argument to a link pointer to do that.

    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link: https://lore.kernel.org/r/20230504134511.828474-4-gregory.greenman@intel.com
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a6016d4b363c15c31daf92bbf6b5fd6e1ba23e9e
Author: Johannes Berg <johannes.berg@intel.com>
Date:   Thu May 4 16:45:02 2023 +0300

    wifi: mac80211: simplify chanctx allocation

    [ Upstream commit 860e1b43da94551cd1e73adc36b3c64cc3e5dc01 ]

    There's no need to call ieee80211_recalc_chanctx_min_def()
    since it cannot and won't call the driver anyway; just use
    _ieee80211_recalc_chanctx_min_def() instead.

    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link: https://lore.kernel.org/r/20230504134511.828474-3-gregory.greenman@intel.com
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 19f8ea5ec150525f566e14acad4c8377ee5a45a2
Author: Linus Walleij <linus.walleij@linaro.org>
Date:   Wed May 10 08:48:11 2023 +0200

    arm64: vdso: Pass (void *) to virt_to_page()

    [ Upstream commit b0abde80620f42d1ceb3de5e4c1a49cdd5628229 ]

    Like the other calls in this function virt_to_page() expects
    a pointer, not an integer.

    However since many architectures implement virt_to_pfn() as
    a macro, this function becomes polymorphic and accepts both a
    (unsigned long) and a (void *).

    Fix this up with an explicit cast.

    Signed-off-by: Linus Walleij <linus.walleij@linaro.org>
    Link: http://lists.infradead.org/pipermail/linux-arm-kernel/2023-May/832583.html
    Signed-off-by: Will Deacon <will@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 1b29614f01d4bfe35a4089353ee2cc994fdf9365
Author: Min-Hua Chen <minhuadotchen@gmail.com>
Date:   Tue May 2 23:19:06 2023 +0800

    arm64/mm: mark private VM_FAULT_X defines as vm_fault_t

    [ Upstream commit d91d580878064b880f3574ac35b98d8b70ee8620 ]

    This patch fixes several sparse warnings for fault.c:

    arch/arm64/mm/fault.c:493:24: sparse: warning: incorrect type in return expression
(different base types)
    arch/arm64/mm/fault.c:493:24: sparse:    expected restricted vm_fault_t
    arch/arm64/mm/fault.c:493:24: sparse:    got int
    arch/arm64/mm/fault.c:501:32: sparse: warning: incorrect type in return expression
(different base types)
    arch/arm64/mm/fault.c:501:32: sparse:    expected restricted vm_fault_t
    arch/arm64/mm/fault.c:501:32: sparse:    got int
    arch/arm64/mm/fault.c:503:32: sparse: warning: incorrect type in return expression
(different base types)
    arch/arm64/mm/fault.c:503:32: sparse:    expected restricted vm_fault_t
    arch/arm64/mm/fault.c:503:32: sparse:    got int
    arch/arm64/mm/fault.c:511:24: sparse: warning: incorrect type in return expression
(different base types)
    arch/arm64/mm/fault.c:511:24: sparse:    expected restricted vm_fault_t
    arch/arm64/mm/fault.c:511:24: sparse:    got int
    arch/arm64/mm/fault.c:670:13: sparse: warning: restricted vm_fault_t degrades to integer
    arch/arm64/mm/fault.c:670:13: sparse: warning: restricted vm_fault_t degrades to integer
    arch/arm64/mm/fault.c:713:39: sparse: warning: restricted vm_fault_t degrades to integer

    Reported-by: kernel test robot <lkp@intel.com>
    Signed-off-by: Min-Hua Chen <minhuadotchen@gmail.com>
    Link: https://lore.kernel.org/r/20230502151909.128810-1-minhuadotchen@gmail.com
    Signed-off-by: Will Deacon <will@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4b31fd53b0e7d2cfc94cef17368152f5f0a9cffb
Author: Dario Binacchi <dario.binacchi@amarulasolutions.com>
Date:   Thu Apr 27 22:45:38 2023 +0200

    ARM: dts: stm32: add pin map for CAN controller on stm32f7

    [ Upstream commit 011644249686f2675e142519cd59e81e04cfc231 ]

    Add pin configurations for using CAN controller on stm32f7.

    Signed-off-by: Dario Binacchi <dario.binacchi@amarulasolutions.com>
    Link: https://lore.kernel.org/all/20230427204540.3126234-4-
dario.binacchi@amarulasolutions.com
    Signed-off-by: Marc Kleine-Budde <mkl@pengutronix.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3132c71c24ed98501070b5128674516a22178bea
Author: Yun Lu <luyun@kylinos.cn>
Date:   Fri May 12 09:20:55 2023 +0800

    wifi: rtl8xxxu: fix authentication timeout due to incorrect RCR value

    [ Upstream commit 20429444e653ee8242dfbf815c0c37866beb371b ]

    When using rtl8192cu with rtl8xxxu driver to connect wifi, there is a
    probability of failure, which shows "authentication with ... timed out".
    Through debugging, it was found that the RCR register has been inexplicably
    modified to an incorrect value, resulting in the nic not being able to
```

```
    receive authenticated frames.

    To fix this problem, add regrcr in rtl8xxxu_priv struct, and store
    the RCR value every time the register is written, and use it the next
    time the register need to be modified.

    Signed-off-by: Yun Lu <luyun@kylinos.cn>
    Link: https://lore.kernel.org/all/20230427020512.1221062-1-luyun_611@163.com
    Signed-off-by: Kalle Valo <kvalo@kernel.org>
    Link: https://lore.kernel.org/r/20230512012055.2990472-1-luyun_611@163.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1a4909376e32687ff4fd87ec2e907608678c6c55
Author: Rubén Gómez <mrgommer@proton.me>
Date:   Mon May 8 18:03:07 2023 +0000

    ACPI: resource: Add IRQ override quirk for LG UltraPC 17U70P

    [ Upstream commit 71a485624c4cbb144169852d7bb8ca8c0667d7a3 ]

    Add an ACPI IRQ override quirk for LG UltraPC 17U70P to address the
    internal keyboard problem on it.

    Link: https://bugzilla.kernel.org/show_bug.cgi?id=213031
    Link: https://bugzilla.kernel.org/show_bug.cgi?id=216983
    Signed-off-by: Rubén Gómez Agudo <mrgommer@proton.me>
    [ rjw: Subject, changelog, white space damage fixes ]
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d1ad2eb081fd2f77e91dd6cdfb35d3f805ae5bbf
Author: Alexander Gordeev <agordeev@linux.ibm.com>
Date:   Sun May 7 18:09:02 2023 +0200

    s390/ipl: fix IPIB virtual vs physical address confusion

    [ Upstream commit 2facd5d3980f3a26c04fe6ec8689a1d019a5812c ]

    The pointer to IPL Parameter Information Block is stored
    in the absolute lowcore for later use by dump tools. That
    pointer is a virtual address, though it should be physical
    instead.

    Note, this does not fix a real issue, since virtual and
    physical addresses are currently the same.

    Suggested-by: Heiko Carstens <hca@linux.ibm.com>
    Reviewed-by: Heiko Carstens <hca@linux.ibm.com>
    Signed-off-by: Alexander Gordeev <agordeev@linux.ibm.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 06cf10f6e1f0cdb3a1548961eab0f7e965a554e2
Author: Alexander Gordeev <agordeev@linux.ibm.com>
Date:   Thu May 4 16:21:48 2023 +0200

    s390/topology: honour nr_cpu_ids when adding CPUs

    [ Upstream commit a33239be2d38ff5a44427db1707c08787508d34a ]

    When SMT thread CPUs are added to CPU masks the nr_cpu_ids
    limit is not checked and could be exceeded. This leads to
    a warning for example if CONFIG_DEBUG_PER_CPU_MAPS is set
    and the command line parameter nr_cpus is set to 1.

    Reviewed-by: Heiko Carstens <hca@linux.ibm.com>
```

```
    Signed-off-by: Alexander Gordeev <agordeev@linux.ibm.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5f877a429d173934010c55bd09c863ef2733d86b
Author: Holger Dengler <dengler@linux.ibm.com>
Date:   Thu Apr 20 14:34:10 2023 +0200

    s390/pkey: zeroize key blobs

    [ Upstream commit 844cf829e5f33e00b279230470c8c93b58b8c16f ]

    Key blobs for the IOCTLs PKEY_KBLOB2PROTK[23] may contain clear key
    material. Zeroize the copies of these keys in kernel memory after
    creating the protected key.

    Reviewed-by: Harald Freudenberger <freude@linux.ibm.com>
    Signed-off-by: Holger Dengler <dengler@linux.ibm.com>
    Signed-off-by: Alexander Gordeev <agordeev@linux.ibm.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 95055e6eb8319d5e929380bb7246362815890b75
Author: Kai Vehmanen <kai.vehmanen@linux.intel.com>
Date:   Fri May 12 13:46:38 2023 +0300

    ASoC: SOF: pm: save io region state in case of errors in resume

    [ Upstream commit 171b53be635ac15d4feafeb33946035649b1ca14 ]

    If there are failures in DSP runtime resume, the device state will not
    reach active and this makes it impossible e.g. to retrieve a possible
    DSP panic dump via "exception" debugfs node. If
    CONFIG_SND_SOC_SOF_DEBUG_ENABLE_DEBUGFS_CACHE=y is set, the data in
    cache is stale. If debugfs cache is not used, the region simply cannot
    be read.

    To allow debugging these scenarios, update the debugfs cache contents in
    resume error handler. User-space can then later retrieve DSP panic and
    other state via debugfs (requires SOF debugfs cache to be enabled in
    build).

    Reported-by: Curtis Malainey <cujomalainey@chromium.org
    Link: https://github.com/thesofproject/linux/issues/4274
    Signed-off-by: Kai Vehmanen <kai.vehmanen@linux.intel.com
    Reviewed-by: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com
    Reviewed-by: Ranjani Sridharan <ranjani.sridharan@linux.intel.com
    Reviewed-by: Curtis Malainey <cujomalainey@chromium.org
    Reviewed-by: Péter Ujfalusi <peter.ujfalusi@linux.intel.com
    Signed-off-by: Peter Ujfalusi <peter.ujfalusi@linux.intel.com
    Link: https://lore.kernel.org/r/20230512104638.21376-1-peter.ujfalusi@linux.intel.com
    Signed-off-by: Mark Brown <broonie@kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ee7632cee305c76cf743fcd07348adbfde6b5d05
Author: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com>
Date:   Fri May 12 13:33:15 2023 +0300

    ASoC: SOF: sof-client-probes: fix pm_runtime imbalance in error handling

    [ Upstream commit bc424273c74c1565c459c8f2a6ed95caee368d0a ]

    When an error occurs, we need to make sure the device can pm_runtime
    suspend instead of keeping it active.

    Signed-off-by: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com
    Reviewed-by: Daniel Baluta <daniel.baluta@nxp.com
```

```
    Reviewed-by: Ranjani Sridharan <ranjani.sridharan@linux.intel.com
    Signed-off-by: Peter Ujfalusi <peter.ujfalusi@linux.intel.com
    Link: https://lore.kernel.org/r/20230512103315.8921-4-peter.ujfalusi@linux.intel.com
    Signed-off-by: Mark Brown <broonie@kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1346ecb19947c90fa7536549c133566ccaec1b35
Author: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com>
Date:   Fri May 12 13:33:14 2023 +0300

    ASoC: SOF: pcm: fix pm_runtime imbalance in error handling

    [ Upstream commit da0fe8fd515a471d373acc3682bfb5522cca4d55 ]

    When an error occurs, we need to make sure the device can pm_runtime
    suspend instead of keeping it active.

    Signed-off-by: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com
    Reviewed-by: Daniel Baluta <daniel.baluta@nxp.com
    Reviewed-by: Ranjani Sridharan <ranjani.sridharan@linux.intel.com
    Signed-off-by: Peter Ujfalusi <peter.ujfalusi@linux.intel.com
    Link: https://lore.kernel.org/r/20230512103315.8921-3-peter.ujfalusi@linux.intel.com
    Signed-off-by: Mark Brown <broonie@kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 94ed039096667eb0c4a4201dd8bb62abb16cc10c
Author: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com>
Date:   Fri May 12 13:33:13 2023 +0300

    ASoC: SOF: debug: conditionally bump runtime_pm counter on exceptions

    [ Upstream commit 3de975862f985f1c9e225a0d13aa3d501373f7c3 ]

    When a firmware IPC error happens during a pm_runtime suspend, we
    ignore the error and suspend anyways. However, the code
    unconditionally increases the runtime_pm counter. This results in a
    confusing configuration where the code will suspend, resume but never
    suspend again due to the use of pm_runtime_get_noresume().

    The intent of the counter increase was to prevent entry in D3, but if
    that transition to D3 is already started it cannot be stopped. In
    addition, there's no point in that case in trying to prevent anything,
    the firmware error is handled and the next resume will re-initialize
    the firmware completely.

    This patch changes the logic to prevent suspend when the device is
    pm_runtime active and has a use_count > 0.

    Signed-off-by: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com
    Reviewed-by: Daniel Baluta <daniel.baluta@nxp.com
    Reviewed-by: Ranjani Sridharan <ranjani.sridharan@linux.intel.com
    Signed-off-by: Peter Ujfalusi <peter.ujfalusi@linux.intel.com
    Link: https://lore.kernel.org/r/20230512103315.8921-2-peter.ujfalusi@linux.intel.com
    Signed-off-by: Mark Brown <broonie@kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 59918dd7a6d1ad098550ca6fcf154c1ae6842bc7
Author: Hyunwoo Kim <v4bel@theori.io>
Date:   Mon Nov 21 06:33:08 2022 +0000

    media: dvb-core: Fix use-after-free due to race condition at dvb_ca_en50221

    [ Upstream commit 280a8ab81733da8bc442253c700a52c4c0886ffd ]

    If the device node of dvb_ca_en50221 is open() and the
```

```
    device is disconnected, a UAF may occur when calling
    close() on the device node.

    The root cause is that wake_up() and wait_event() for
    dvbdev->wait_queue are not implemented.

    So implement wait_event() function in dvb_ca_en50221_release()
    and add 'remove_mutex' which prevents race condition
    for 'ca->exit'.

    [mchehab: fix a checkpatch warning]

    Link: https://lore.kernel.org/linux-media/20221121063308.GA33821@ubuntu
    Signed-off-by: Hyunwoo Kim <v4bel@theori.io>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 47dc2e5f5fb45aff7f9c32f10412125ee13cb5ce
Author: Takashi Iwai <tiwai@suse.de>
Date:   Fri May 12 16:18:00 2023 +0100

    media: dvb-core: Fix kernel WARNING for blocking operation in wait_event*()

    [ Upstream commit b8c75e4a1b325ea0a9433fa8834be97b5836b946 ]

    Using a semaphore in the wait_event*() condition is no good idea.
    It hits a kernel WARN_ON() at prepare_to_wait_event() like:
      do not call blocking ops when !TASK_RUNNING; state=1 set at
      prepare_to_wait_event+0x6d/0x690

    For avoiding the potential deadlock, rewrite to an open-coded loop
    instead.  Unlike the loop in wait_event*(), this uses wait_woken()
    after the condition check, hence the task state stays consistent.

    CVE-2023-31084 was assigned to this bug.

    Link: https://lore.kernel.org/r/CA+UBctCu7fXn4q41O_3=id1+OdyQ85tZY1x+TkT-
6OVBL6KAUw@mail.gmail.com/

    Link: https://lore.kernel.org/linux-media/20230512151800.1874-1-tiwai@suse.de
    Reported-by: Yu Hao <yhao016@ucr.edu>
    Closes: https://nvd.nist.gov/vuln/detail/CVE-2023-31084
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cf1d83a221e7612c7daa0a043a090bca64986da2
Author: Hyunwoo Kim <imv4bel@gmail.com>
Date:   Thu Nov 17 04:59:24 2022 +0000

    media: dvb-core: Fix use-after-free due to race at dvb_register_device()

    [ Upstream commit 627bb528b086b4136315c25d6a447a98ea9448d3 ]

    dvb_register_device() dynamically allocates fops with kmemdup()
    to set the fops->owner.
    And these fops are registered in 'file->f_ops' using replace_fops()
    in the dvb_device_open() process, and kfree()d in dvb_free_device().

    However, it is not common to use dynamically allocated fops instead
    of 'static const' fops as an argument of replace_fops(),
    and UAF may occur.
    These UAFs can occur on any dvb type using dvb_register_device(),
    such as dvb_dvr, dvb_demux, dvb_frontend, dvb_net, etc.
```

```
    So, instead of kfree() the fops dynamically allocated in
    dvb_register_device() in dvb_free_device() called during the
    .disconnect() process, kfree() it collectively in exit_dvbdev()
    called when the dvbdev.c module is removed.

    Link: https://lore.kernel.org/linux-media/20221117045925.14297-4-imv4bel@gmail.com
    Signed-off-by: Hyunwoo Kim <imv4bel@gmail.com>
    Reported-by: kernel test robot <lkp@intel.com>
    Reported-by: Dan Carpenter <error27@gmail.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8bade849b15b3ecb62893f328b2cc4cdc65ac0c6
Author: Hyunwoo Kim <imv4bel@gmail.com>
Date:   Thu Nov 17 04:59:23 2022 +0000

    media: dvb-core: Fix use-after-free due on race condition at dvb_net

    [ Upstream commit 4172385b0c9ac366dcab78eda48c26814b87ed1a ]

    A race condition may occur between the .disconnect function, which
    is called when the device is disconnected, and the dvb_device_open()
    function, which is called when the device node is open()ed.
    This results in several types of UAFs.

    The root cause of this is that you use the dvb_device_open() function,
    which does not implement a conditional statement
    that checks 'dvbnet->exit'.

    So, add 'remove_mutex` to protect 'dvbnet->exit' and use
    locked_dvb_net_open() function to check 'dvbnet->exit'.

    [mchehab: fix a checkpatch warning]

    Link: https://lore.kernel.org/linux-media/20221117045925.14297-3-imv4bel@gmail.com
    Signed-off-by: Hyunwoo Kim <imv4bel@gmail.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8994830135b38bbf090956b00231edf531de2abc
Author: Hyunwoo Kim <imv4bel@gmail.com>
Date:   Thu Nov 17 04:59:22 2022 +0000

    media: dvb-core: Fix use-after-free on race condition at dvb_frontend

    [ Upstream commit 6769a0b7ee0c3b31e1b22c3fadff2bfb642de23f ]

    If the device node of dvb_frontend is open() and the device is
    disconnected, many kinds of UAFs may occur when calling close()
    on the device node.

    The root cause of this is that wake_up() for dvbdev->wait_queue
    is implemented in the dvb_frontend_release() function, but
    wait_event() is not implemented in the dvb_frontend_stop() function.

    So, implement wait_event() function in dvb_frontend_stop() and
    add 'remove_mutex' which prevents race condition for 'fe->exit'.

    [mchehab: fix a couple of checkpatch warnings and some mistakes at the error handling
logic]

    Link: https://lore.kernel.org/linux-media/20221117045925.14297-2-imv4bel@gmail.com
    Signed-off-by: Hyunwoo Kim <imv4bel@gmail.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit ec35bef6256ddc24114c7e6749c0baa1b467bcc4
Author: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
Date:   Sun Mar 12 13:13:18 2023 +0000

    media: mn88443x: fix !CONFIG_OF error by drop of_match_ptr from ID table

    [ Upstream commit ae11c0efaec32fb45130ee9886689f467232eebc ]

    The driver will match mostly by DT table (even thought there is regular
    ID table) so there is little benefit in of_match_ptr (this also allows
    ACPI matching via PRP0001, even though it might not be relevant here).
    This also fixes !CONFIG_OF error:

      drivers/media/dvb-frontends/mn88443x.c:782:34: error: ‘mn88443x_of_match’ defined but
not used [-Werror=unused-const-variable=]

    Link: https://lore.kernel.org/linux-media/20230312131318.351173-28-
krzysztof.kozlowski@linaro.org
    Signed-off-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dbef7d1ffea0ccc95446c5383e0be65babedf667
Author: Hyunwoo Kim <imv4bel@gmail.com>
Date:   Thu Nov 17 04:59:25 2022 +0000

    media: ttusb-dec: fix memory leak in ttusb_dec_exit_dvb()

    [ Upstream commit 517a281338322ff8293f988771c98aaa7205e457 ]

    Since dvb_frontend_detach() is not called in ttusb_dec_exit_dvb(),
    which is called when the device is disconnected, dvb_frontend_free()
    is not finally called.

    This causes a memory leak just by repeatedly plugging and
    unplugging the device.

    Fix this issue by adding dvb_frontend_detach() to ttusb_dec_exit_dvb().

    Link: https://lore.kernel.org/linux-media/20221117045925.14297-5-imv4bel@gmail.com
    Signed-off-by: Hyunwoo Kim <imv4bel@gmail.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4e8a33594cd5ff8d24a0dec22587bb0c8b8f65ef
Author: YongSu Yoo <yongsuyoo0215@gmail.com>
Date:   Thu Aug 18 13:50:27 2022 +0100

    media: dvb_ca_en50221: fix a size write bug

    [ Upstream commit a4315e5be7020aac9b24a8151caf4bb85224cd0e ]

    The function of "dvb_ca_en50221_write_data" at source/drivers/media
    /dvb-core/dvb_ca_en50221.c is used for two cases.
    The first case is for writing APDU data in the function of
    "dvb_ca_en50221_io_write" at source/drivers/media/dvb-core/
    dvb_ca_en50221.c.
    The second case is for writing the host link buf size on the
    Command Register in the function of "dvb_ca_en50221_link_init"
    at source/drivers/media/dvb-core/dvb_ca_en50221.c.
    In the second case, there exists a bug like following.
    In the function of the "dvb_ca_en50221_link_init",
    after a TV host calculates the host link buf_size,
    the TV host writes the calculated host link buf_size on the
```

```
    Size Register.
    Accroding to the en50221 Spec (the page 60 of
    https://dvb.org/wp-content/uploads/2020/02/En50221.V1.pdf),
    before this writing operation, the "SW(CMDREG_SW)" flag in the
    Command Register should be set. We can see this setting operation
    in the function of the "dvb_ca_en50221_link_init" like below.
    ...
            if ((ret = ca->pub->write_cam_control(ca->pub, slot,
    CTRLIF_COMMAND, IRQEN | CMDREG_SW)) != 0)
                    return ret;
    ...
    But, after that, the real writing operation is implemented using
    the function of the "dvb_ca_en50221_write_data" in the function of
    "dvb_ca_en50221_link_init", and the "dvb_ca_en50221_write_data"
    includes the function of "ca->pub->write_cam_control",
    and the function of the "ca->pub->write_cam_control" in the
    function of the "dvb_ca_en50221_wrte_data" does not include
    "CMDREG_SW" flag like below.
    ...
            if ((status = ca->pub->write_cam_control(ca->pub, slot,
    CTRLIF_COMMAND, IRQEN | CMDREG_HC)) != 0)
    ...
    In the above source code, we can see only the "IRQEN | CMDREG_HC",
    but we cannot see the "CMDREG_SW".
    The "CMDREG_SW" flag which was set in the function of the
    "dvb_ca_en50221_link_init" was rollbacked by the follwoing function
    of the "dvb_ca_en50221_write_data".
    This is a bug. and this bug causes that the calculated host link buf_size
    is not properly written in the CI module.
    Through this patch, we fix this bug.

    Link: https://lore.kernel.org/linux-media/20220818125027.1131-1-yongsuyoo0215@gmail.com
    Signed-off-by: YongSu Yoo <yongsuyoo0215@gmail.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7bd6525c2c37d8d3e8287be56e681fab952b8fb3
Author: Wei Chen <harperchen1110@gmail.com>
Date:   Wed Mar 15 13:45:18 2023 +0000

    media: netup_unidvb: fix irq init by register it at the end of probe

    [ Upstream commit e6ad6233592593079db5c8fa592c298e51bc1356 ]

    IRQ handler netup_spi_interrupt() takes spinlock spi->lock. The lock
    is initialized in netup_spi_init(). However, irq handler is registered
    before initializing the lock.

    Spinlock dma->lock and i2c->lock suffer from the same problem.

    Fix this by registering the irq at the end of probe.

    Link: https://lore.kernel.org/linux-media/20230315134518.1074497-1-
harperchen1110@gmail.com
    Signed-off-by: Wei Chen <harperchen1110@gmail.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 361ada7a7468bcf8fb1ad7120bc0c53e78b5be95
Author: Wei Chen <harperchen1110@gmail.com>
Date:   Tue Mar 28 13:44:16 2023 +0100

    media: dvb-usb: dw2102: fix uninit-value in su3000_read_mac_address

    [ Upstream commit a3fd1ef27aa686d871cefe207bd6168c4b0cd29e ]
```

```
    In su3000_read_mac_address, if i2c_transfer fails to execute two
    messages, array mac address will not be initialized. Without handling
    such error, later in function dvb_usb_adapter_dvb_init, proposed_mac
    is accessed before initialization.

    Fix this error by returning a negative value if message execution fails.

    Link: https://lore.kernel.org/linux-media/20230328124416.560889-1-
harperchen1110@gmail.com
    Signed-off-by: Wei Chen <harperchen1110@gmail.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c6dbca78d8e63f9c2857886433bf1c25fdfe428f
Author: Wei Chen <harperchen1110@gmail.com>
Date:   Mon Mar 13 09:50:08 2023 +0000

    media: dvb-usb: digitv: fix null-ptr-deref in digitv_i2c_xfer()

    [ Upstream commit 9ded5bd2a49ce3015b7c936743eec0a0e6e11f0c ]

    In digitv_i2c_xfer, msg is controlled by user. When msg[i].buf
    is null and msg[i].len is zero, former checks on msg[i].buf would be
    passed. Malicious data finally reach digitv_i2c_xfer. If accessing
    msg[i].buf[0] without sanity check, null ptr deref would happen. We add
    check on msg[i].len to prevent crash.

    Similar commit:
    commit 0ed554fd769a ("media: dvb-usb: az6027: fix null-ptr-deref in az6027_i2c_xfer()")

    Link: https://lore.kernel.org/linux-media/20230313095008.1039689-1-
harperchen1110@gmail.com
    Signed-off-by: Wei Chen <harperchen1110@gmail.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 94d062e14bc15df534eb1203ac749945132f04bd
Author: Zhang Shurong <zhang_shurong@foxmail.com>
Date:   Sun May 7 15:52:47 2023 +0100

    media: dvb-usb-v2: rtl28xxu: fix null-ptr-deref in rtl28xxu_i2c_xfer

    [ Upstream commit aa4a447b81b84f69c1a89ad899df157f386d7636 ]

    In rtl28xxu_i2c_xfer, msg is controlled by user. When msg[i].buf
    is null and msg[i].len is zero, former checks on msg[i].buf would be
    passed. Malicious data finally reach rtl28xxu_i2c_xfer. If accessing
    msg[i].buf[0] without sanity check, null ptr deref would happen.
    We add check on msg[i].len to prevent crash.

    Similar commit:
    commit 0ed554fd769a
    ("media: dvb-usb: az6027: fix null-ptr-deref in az6027_i2c_xfer()")

    Link: https://lore.kernel.org/linux-
media/tencent_3623572106754AC2F266B316798B0F6CCA05@qq.com
    Signed-off-by: Zhang Shurong <zhang_shurong@foxmail.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b18908216d4a75a9be54ce9e997cf7b52d524eac
Author: Wei Chen <harperchen1110@gmail.com>
Date:   Mon Mar 13 09:27:51 2023 +0000
```

```
    media: dvb-usb-v2: ce6230: fix null-ptr-deref in ce6230_i2c_master_xfer()

    [ Upstream commit dff919090155fb22679869e8469168f270dcd97f ]

    In ce6230_i2c_master_xfer, msg is controlled by user. When msg[i].buf
    is null and msg[i].len is zero, former checks on msg[i].buf would be
    passed. Malicious data finally reach ce6230_i2c_master_xfer. If accessing
    msg[i].buf[0] without sanity check, null ptr deref would happen. We add
    check on msg[i].len to prevent crash.

    Similar commit:
    commit 0ed554fd769a ("media: dvb-usb: az6027: fix null-ptr-deref in az6027_i2c_xfer()")

    Link: https://lore.kernel.org/linux-media/20230313092751.209496-1-
harperchen1110@gmail.com
    Signed-off-by: Wei Chen <harperchen1110@gmail.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4bb4fdab10a59e1a6a0dd3093769dbad81970681
Author: Wei Chen <harperchen1110@gmail.com>
Date:   Mon Mar 13 08:58:53 2023 +0000

    media: dvb-usb-v2: ec168: fix null-ptr-deref in ec168_i2c_xfer()

    [ Upstream commit a6dcefcc08eca1bf4e3d213c97c3cfb75f377935 ]

    In ec168_i2c_xfer, msg is controlled by user. When msg[i].buf is null
    and msg[i].len is zero, former checks on msg[i].buf would be passed.
    If accessing msg[i].buf[0] without sanity check, null pointer deref
    would happen. We add check on msg[i].len to prevent crash.

    Similar commit:
    commit 0ed554fd769a ("media: dvb-usb: az6027: fix null-ptr-deref in az6027_i2c_xfer()")

    Link: https://lore.kernel.org/linux-media/20230313085853.3252349-1-
harperchen1110@gmail.com
    Signed-off-by: Wei Chen <harperchen1110@gmail.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 722993741c696ebe4855a403c98408d720be2386
Author: Wei Chen <harperchen1110@gmail.com>
Date:   Fri Mar 10 16:56:04 2023 +0000

    media: dvb-usb: az6027: fix three null-ptr-deref in az6027_i2c_xfer()

    [ Upstream commit 858e97d7956d17a2cb56a9413468704a4d5abfe1 ]

    In az6027_i2c_xfer, msg is controlled by user. When msg[i].buf is null,
    commit 0ed554fd769a ("media: dvb-usb: az6027: fix null-ptr-deref in
    az6027_i2c_xfer()") fix the null-ptr-deref bug when msg[i].addr is 0x99.
    However, null-ptr-deref also happens when msg[i].addr is 0xd0 and 0xc0.
    We add check on msg[i].len to prevent null-ptr-deref.

    Link: https://lore.kernel.org/linux-media/20230310165604.3093483-1-
harperchen1110@gmail.com
    Signed-off-by: Wei Chen <harperchen1110@gmail.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit bf3ec0f5cd1efbc79b7adab290915b97b2bead0b
Author: YongSu Yoo <yongsuyoo0215@gmail.com>
Date:   Sun Mar 5 21:25:19 2023 +0000
```

```
    media: dvb_demux: fix a bug for the continuity counter

    [ Upstream commit 7efb10d8dc70ea3000cc70dca53407c52488acd1 ]

    In dvb_demux.c, some logics exist which compare the expected
    continuity counter and the real continuity counter. If they
    are not matched each other, both of the expected continuity
    counter and the real continuity counter should be printed.
    But there exists a bug that the expected continuity counter
    is not correctly printed. The expected continuity counter is
    replaced with the real countinuity counter + 1 so that
    the epected continuity counter is not correclty printed.
    This is wrong. This bug is fixed.

    Link: https://lore.kernel.org/linux-media/20230305212519.499-1-yongsuyoo0215@gmail.com

    Signed-off-by: YongSu Yoo <yongsuyoo0215@gmail.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6d4e7c6d9988f3c675f2ff5626c8f0d154ef801c
Author: Paweł Anikiel <pan@semihalf.com>
Date:   Mon May 8 13:30:37 2023 +0200

    ASoC: ssm2602: Add workaround for playback distortions

    [ Upstream commit f63550e2b165208a2f382afcaf5551df9569e1d4 ]

    Apply a workaround for what appears to be a hardware quirk.

    The problem seems to happen when enabling "whole chip power" (bit D7
    register R6) for the very first time after the chip receives power. If
    either "output" (D4) or "DAC" (D3) aren't powered on at that time,
    playback becomes very distorted later on.

    This happens on the Google Chameleon v3, as well as on a ZYBO Z7-10:
    https://ez.analog.com/audio/f/q-a/543726/solved-ssm2603-right-output-offset-issue/480229
    I suspect this happens only when using an external MCLK signal (which
    is the case for both of these boards).

    Here are some experiments run on a Google Chameleon v3. These were run
    in userspace using a wrapper around the i2cset utility:
    ssmset() {
            i2cset -y 0 0x1a $(($1*2)) $2
    }

    For each of the following sequences, we apply power to the ssm2603
    chip, set the configuration registers R0-R5 and R7-R8, run the selected
    sequence, and check for distortions on playback.

      ssmset 0x09 0x01 # core
      ssmset 0x06 0x07 # chip, out, dac
      OK

      ssmset 0x09 0x01 # core
      ssmset 0x06 0x87 # out, dac
      ssmset 0x06 0x07 # chip
      OK

      (disable MCLK)
      ssmset 0x09 0x01 # core
      ssmset 0x06 0x1f # chip
      ssmset 0x06 0x07 # out, dac
      (enable MCLK)
      OK
```

```
        ssmset 0x09 0x01 # core
        ssmset 0x06 0x1f # chip
        ssmset 0x06 0x07 # out, dac
        NOT OK

        ssmset 0x06 0x1f # chip
        ssmset 0x09 0x01 # core
        ssmset 0x06 0x07 # out, dac
        NOT OK

        ssmset 0x09 0x01 # core
        ssmset 0x06 0x0f # chip, out
        ssmset 0x06 0x07 # dac
        NOT OK

        ssmset 0x09 0x01 # core
        ssmset 0x06 0x17 # chip, dac
        ssmset 0x06 0x07 # out
        NOT OK

    For each of the following sequences, we apply power to the ssm2603
    chip, run the selected sequence, issue a reset with R15, configure
    R0-R5 and R7-R8, run one of the NOT OK sequences from above, and check
    for distortions.

        ssmset 0x09 0x01 # core
        ssmset 0x06 0x07 # chip, out, dac
        OK

        (disable MCLK)
        ssmset 0x09 0x01 # core
        ssmset 0x06 0x07 # chip, out, dac
        (enable MCLK after reset)
        NOT OK

        ssmset 0x09 0x01 # core
        ssmset 0x06 0x17 # chip, dac
        NOT OK

        ssmset 0x09 0x01 # core
        ssmset 0x06 0x0f # chip, out
        NOT OK

        ssmset 0x06 0x07 # chip, out, dac
        NOT OK

    Signed-off-by: Paweł Anikiel <pan@semihalf.com
    Link: https://lore.kernel.org/r/20230508113037.137627-8-pan@semihalf.com
    Signed-off-by: Mark Brown <broonie@kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 66d849167e8ae06b7e882c42d21d240a49622182
Author: Alexandru Sorodoc <ealex95@gmail.com>
Date:   Thu May 11 19:15:10 2023 +0300

    ALSA: hda/realtek: Add quirks for ASUS GU604V and GU603V

    [ Upstream commit 4b963ae1df6426f0e51de64133d379d9bde50c48 ]

    These models use 2 CS35L41 amplifiers using SPI for down-facing
    speakers.

    alc285_fixup_speaker2_to_dac1 is needed to fix volume control of the
    down-facing speakers.
```

```
    Pin configs are needed to enable headset mic detection.

    Note that these models lack the ACPI _DSD properties needed to
    initialize the amplifiers. They can be added during boot to get working
    sound out of the speakers:
      https://gist.github.com/lamperez/862763881c0e1c812392b5574727f6ff

    Signed-off-by: Alexandru Sorodoc <ealex95@gmail.com>
    Link: https://lore.kernel.org/r/20230511161510.315170-1-ealex95@gmail.com
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c148e7c204ac93909fe6f2326dc662d1bb7d8b46
Author: Martin Povišer <povik+lin@cutebit.org>
Date:   Tue May 9 17:34:12 2023 +0200

    ASoC: dt-bindings: Adjust #sound-dai-cells on TI's single-DAI codecs

    [ Upstream commit efb2bfd7b3d210c479b9361c176d7426e5eb8663 ]

    A bunch of TI's codecs have binding schemas which force #sound-dai-cells
    to one despite those codecs only having a single DAI. Allow for bindings
    with zero DAI cells and deprecate the former non-zero value.

    Signed-off-by: Martin Povišer <povik+lin@cutebit.org
    Link: https://lore.kernel.org/r/20230509153412.62847-1-povik+lin@cutebit.org
    Signed-off-by: Mark Brown <broonie@kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 735234ca421d6c4f9d0361bec8de92e02708f2cd
Author: Aidan MacDonald <aidanmacdonald.0x0@gmail.com>
Date:   Tue May 9 13:51:34 2023 +0100

    ASoC: jz4740-i2s: Make I2S divider calculations more robust

    [ Upstream commit ad721bc919edfd8b4b06977458a412011e2f0c50 ]

    When the CPU supplies bit/frame clocks, the system clock (clk_i2s)
    is divided to produce the bit clock. This is a simple 1/N divider
    with a fairly limited range, so for a given system clock frequency
    only a few sample rates can be produced. Usually a wider range of
    sample rates is supported by varying the system clock frequency.

    The old calculation method was not very robust and could easily
    produce the wrong clock rate, especially with non-standard rates.
    For example, if the system clock is 1.99x the target bit clock
    rate, the divider would be calculated as 1 instead of the more
    accurate 2.

    Instead, use a more accurate method that considers two adjacent
    divider settings and selects the one that produces the least error
    versus the requested rate. If the error is 5% or higher then the
    rate setting is rejected to prevent garbled audio.

    Skip divider calculation when the codec is supplying both the bit
    and frame clock; in that case, the divider outputs are unused and
    we don't want to constrain the sample rate.

    Signed-off-by: Aidan MacDonald <aidanmacdonald.0x0@gmail.com
    Link: https://lore.kernel.org/r/20230509125134.208129-1-aidanmacdonald.0x0@gmail.com
    Signed-off-by: Mark Brown <broonie@kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ac7a663054d9c5edd81f1f946a496b63271bb50f
```

```
Author: Benedict Wong <benedictwong@google.com>
Date:   Wed May 10 01:14:14 2023 +0000

    xfrm: Check if_id in inbound policy/secpath match

    [ Upstream commit 8680407b6f8f5fba59e8f1d63c869abc280f04df ]

    This change ensures that if configured in the policy, the if_id set in
    the policy and secpath states match during the inbound policy check.
    Without this, there is potential for ambiguity where entries in the
    secpath differing by only the if_id could be mismatched.

    Notably, this is checked in the outbound direction when resolving
    templates to SAs, but not on the inbound path when matching SAs and
    policies.

    Test: Tested against Android kernel unit tests & CTS
    Signed-off-by: Benedict Wong <benedictwong@google.com>
    Signed-off-by: Steffen Klassert <steffen.klassert@secunet.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 471419f17362231b4d8f81d5fbcefa09989695a3
Author: Johannes Berg <johannes.berg@intel.com>
Date:   Tue Apr 25 10:38:37 2023 +0200

    um: harddog: fix modular build

    [ Upstream commit 73a23d7710331a530e972903318528b75e5a5f58 ]

    Since we no longer (want to) export any libc symbols the
    _user portions of any drivers need to be built into image
    rather than the module. I missed this for the watchdog.
    Fix the watchdog accordingly.

    Reported-by: kernel test robot <lkp@intel.com>
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Richard Weinberger <richard@nod.at>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 19298331eca6b05f622ce40f6d784ba2f91d1831
Author: V sujith kumar Reddy <Vsujithkumar.Reddy@amd.com>
Date:   Mon May 8 12:35:08 2023 +0530

    ASoC: SOF: amd: Fix NULL pointer crash in acp_sof_ipc_msg_data function

    [ Upstream commit 051d71e073614a72ad423d6dacba37a7eeff274d ]

    Check substream and runtime variables before assigning.

    Signed-off-by: V sujith kumar Reddy <Vsujithkumar.Reddy@amd.com
    Link: https://lore.kernel.org/r/20230508070510.6100-1-Vsujithkumar.Reddy@amd.com
    Signed-off-by: Mark Brown <broonie@kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a62d40c1decf394f7e79a5bb0b405be2fb82a2f4
Author: Hao Zeng <zenghao@kylinos.cn>
Date:   Tue Apr 18 09:30:56 2023 +0800

    cpupower:Fix resource leaks in sysfs_get_enabled()

    [ Upstream commit e652be0f59d4ba4d5c636b1f7f4dcb73aae049fa ]

    The sysfs_get_enabled() opened file processor not closed,
    may cause a file handle leak.
    Putting error handling and resource cleanup code together
```

```
    makes the code easy to maintain and read.
    Removed the unnecessary else if branch from the original
    function, as it should return an error in cases other than '0'.

    Signed-off-by: Hao Zeng <zenghao@kylinos.cn>
    Suggested-by: Shuah Khan <skhan@linuxfoundation.org>
    Signed-off-by: Shuah Khan <skhan@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 23d41a123df267a797e54790c8913c2bac64cefb
Author: Maxim Kochetkov <fido_max@inbox.ru>
Date:   Fri May 5 09:28:20 2023 +0300

    ASoC: dwc: limit the number of overrun messages

    [ Upstream commit ab6ecfbf40fccf74b6ec2ba7ed6dd2fc024c3af2 ]

    On slow CPU (FPGA/QEMU emulated) printing overrun messages from
    interrupt handler to uart console may leads to more overrun errors.
    So use dev_err_ratelimited to limit the number of error messages.

    Signed-off-by: Maxim Kochetkov <fido_max@inbox.ru
    Link: https://lore.kernel.org/r/20230505062820.21840-1-fido_max@inbox.ru
    Signed-off-by: Mark Brown <broonie@kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 37dd96b1c53c4068ddc28a96d2938e96f279b3ac
Author: Jeremy Soller <jeremy@system76.com>
Date:   Fri May 5 10:14:58 2023 -0600

    ASoC: amd: yc: Add DMI entry to support System76 Pangolin 12

    [ Upstream commit 7b9891ad25246b18b5ccc19518da7abc7763aa0a ]

    Add pang12 quirk to enable the internal microphone.

    Signed-off-by: Jeremy Soller <jeremy@system76.com
    Signed-off-by: Tim Crawford <tcrawford@system76.com
    Link: https://lore.kernel.org/r/20230505161458.19676-1-tcrawford@system76.com
    Signed-off-by: Mark Brown <broonie@kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit da3a9a5a09c9dc661305ae4060eabe056c96d8a1
Author: Adrian Huang <ahuang12@lenovo.com>
Date:   Fri Apr 21 16:08:00 2023 +0800

    nvme-pci: clamp max_hw_sectors based on DMA optimized limitation

    [ Upstream commit 3710e2b056cb92ad816e4d79fa54a6a5b6ad8cbd ]

    When running the fio test on a 448-core AMD server + a NVME disk,
    a soft lockup or a hard lockup call trace is shown:

    [soft lockup]
    watchdog: BUG: soft lockup - CPU#126 stuck for 23s! [swapper/126:0]
    RIP: 0010:_raw_spin_unlock_irqrestore+0x21/0x50
    ...
    Call Trace:
     <IRQ>
     fq_flush_timeout+0x7d/0xd0
     ? __pfx_fq_flush_timeout+0x10/0x10
     call_timer_fn+0x2e/0x150
     run_timer_softirq+0x48a/0x560
     ? __pfx_fq_flush_timeout+0x10/0x10
     ? clockevents_program_event+0xaf/0x130
```

```
 __do_softirq+0xf1/0x335
 irq_exit_rcu+0x9f/0xd0
 sysvec_apic_timer_interrupt+0xb4/0xd0
 </IRQ>
 <TASK>
 asm_sysvec_apic_timer_interrupt+0x1f/0x30
...

Obvisouly, fq_flush_timeout spends over 20 seconds. Here is ftrace log:

               |  fq_flush_timeout() {
               |    fq_ring_free() {
               |      put_pages_list() {
   0.170 us    |        free_unref_page_list();
   0.810 us    |      }
               |      free_iova_fast() {
               |        free_iova() {
 * 85622.66 us |          _raw_spin_lock_irqsave();
   2.860 us    |          remove_iova();
   0.600 us    |          _raw_spin_unlock_irqrestore();
   0.470 us    |          lock_info_report();
   2.420 us    |          free_iova_mem.part.0();
 * 85638.27 us |        }
 * 85638.84 us |      }
               |      put_pages_list() {
   0.230 us    |        free_unref_page_list();
   0.470 us    |      }
   ...         |    ...
 $ 31017069 us |  }

Most of cores are under lock contention for acquiring iova_rbtree_lock due
to the iova flush queue mechanism.

[hard lockup]
NMI watchdog: Watchdog detected hard LOCKUP on cpu 351
RIP: 0010:native_queued_spin_lock_slowpath+0x2d8/0x330

Call Trace:
 <IRQ>
 _raw_spin_lock_irqsave+0x4f/0x60
 free_iova+0x27/0xd0
 free_iova_fast+0x4d/0x1d0
 fq_ring_free+0x9b/0x150
 iommu_dma_free_iova+0xb4/0x2e0
 __iommu_dma_unmap+0x10b/0x140
 iommu_dma_unmap_sg+0x90/0x110
 dma_unmap_sg_attrs+0x4a/0x50
 nvme_unmap_data+0x5d/0x120 [nvme]
 nvme_pci_complete_batch+0x77/0xc0 [nvme]
 nvme_irq+0x2ee/0x350 [nvme]
 ? __pfx_nvme_pci_complete_batch+0x10/0x10 [nvme]
 __handle_irq_event_percpu+0x53/0x1a0
 handle_irq_event_percpu+0x19/0x60
 handle_irq_event+0x3d/0x60
 handle_edge_irq+0xb3/0x210
 __common_interrupt+0x7f/0x150
 common_interrupt+0xc5/0xf0
 </IRQ>
 <TASK>
 asm_common_interrupt+0x2b/0x40
...

ftrace shows fq_ring_free spends over 10 seconds [1]. Again, most of
cores are under lock contention for acquiring iova_rbtree_lock due
to the iova flush queue mechanism.
```

```
    [Root Cause]
    The root cause is that the max_hw_sectors_kb of nvme disk (mdts=10)
    is 4096kb, which streaming DMA mappings cannot benefit from the
    scalable IOVA mechanism introduced by the commit 9257b4a206fc
    ("iommu/iova: introduce per-cpu caching to iova allocation") if
    the length is greater than 128kb.

    To fix the lock contention issue, clamp max_hw_sectors based on
    DMA optimized limitation in order to leverage scalable IOVA mechanism.

    Note: The issue does not happen with another NVME disk (mdts = 5
    and max_hw_sectors_kb = 128)

    [1] https://gist.github.com/AdrianHuang/bf8ec7338204837631fbdaed25d19cc4

    Suggested-by: Keith Busch <kbusch@kernel.org>
    Reported-and-tested-by: Jiwei Sun <sunjw10@lenovo.com>
    Signed-off-by: Adrian Huang <ahuang12@lenovo.com>
    Reviewed-by: Keith Busch <kbusch@kernel.org>
    Signed-off-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 361455467dfcbc1b1a280dc94ec5cde39a280ea5
Author: Hristo Venev <hristo@venev.name>
Date:   Tue Apr 25 22:58:54 2023 +0300

    nvme-pci: add quirk for missing secondary temperature thresholds

    [ Upstream commit bd375feeaf3408ed00e08c3bc918d6be15f691ad ]

    On Kingston KC3000 and Kingston FURY Renegade (both have the same PCI
    IDs) accessing temp3_{min,max} fails with an invalid field error (note
    that there is no problem setting the thresholds for temp1).

    This contradicts the NVM Express Base Specification 2.0b, page 292:

      The over temperature threshold and under temperature threshold
      features shall be implemented for all implemented temperature sensors
      (i.e., all Temperature Sensor fields that report a non-zero value).

    Define NVME_QUIRK_NO_SECONDARY_TEMP_THRESH that disables the thresholds
    for all but the composite temperature and set it for this device.

    Signed-off-by: Hristo Venev <hristo@venev.name>
    Reviewed-by: Guenter Roeck <linux@roeck-us.net>
    Signed-off-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 085ac5fff937bac079328d3bd863ace6330c76eb
Author: Sagi Grimberg <sagi@grimberg.me>
Date:   Wed May 3 18:57:33 2023 +0300

    nvme-pci: add NVME_QUIRK_BOGUS_NID for HS-SSD-FUTURE 2048G

    [ Upstream commit 1616d6c3717bae9041a4240d381ec56ccdaafedc ]

    Add a quirk to fix HS-SSD-FUTURE 2048G SSD drives reporting duplicate
    nsids.

    Link: https://bugzilla.kernel.org/show_bug.cgi?id=217384
    Reported-by: Andrey God <andreygod83@protonmail.com>
    Signed-off-by: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit faf4ccfaf1fd6b2ba76b7f16ed8cec1e748cc4ec
Author: Guoqing Jiang <guoqing.jiang@linux.dev>
Date:   Fri May 12 11:46:31 2023 +0800

    block/rnbd: replace REQ_OP_FLUSH with REQ_OP_WRITE

    [ Upstream commit 5e6e08087a4acb4ee3574cea32dbff0f63c7f608 ]

    Since flush bios are implemented as writes with no data and
    the preflush flag per Christoph's comment [1].

    And we need to change it in rnbd accordingly. Otherwise, I
    got splatting when create fs from rnbd client.

    [  464.028545] ------------[ cut here ]------------
    [  464.028553] WARNING: CPU: 0 PID: 65 at block/blk-core.c:751
submit_bio_noacct+0x32c/0x5d0
    [ ... ]
    [  464.028668] CPU: 0 PID: 65 Comm: kworker/0:1H Tainted: G           OE      6.4.0-rc1
#9
    [  464.028671] Hardware name: QEMU Standard PC (Q35 + ICH9, 2009), BIOS rel-1.15.0-0-
g2dd4b9b-rebuilt.opensuse.org 04/01/2014
    [  464.028673] Workqueue: ib-comp-wq ib_cq_poll_work [ib_core]
    [  464.028717] RIP: 0010:submit_bio_noacct+0x32c/0x5d0
    [  464.028720] Code: 03 0f 85 51 fe ff ff 48 8b 43 18 8b 88 04 03 00 00 85 c9 0f 85 3f
fe ff ff e9 be fd ff ff 0f b6 d0 3c 0d 74 26 83 fa 01 74 21 <0f> 0b b8 0a 00 00 00 e9 56 fd
ff ff 4c 89 e7 e8 70 a1 03 00 84 c0
    [  464.028722] RSP: 0018:ffffaf3680b57c68 EFLAGS: 00010202
    [  464.028724] RAX: 0000000000060802 RBX: ffffa09dcc18bf00 RCX: 0000000000000000
    [  464.028726] RDX: 0000000000000002 RSI: 0000000000000000 RDI: ffffa09dde081d00
    [  464.028727] RBP: ffffaf3680b57c98 R08: ffffa09dde081d00 R09: ffffa09e38327200
    [  464.028729] R10: 0000000000000000 R11: 0000000000000000 R12: ffffa09dde081d00
    [  464.028730] R13: ffffa09dcb06e1e8 R14: 0000000000000000 R15: 0000000000200000
    [  464.028733] FS:  0000000000000000(0000) GS:ffffa09e3bc00000(0000)
knlGS:0000000000000000
    [  464.028735] CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    [  464.028736] CR2: 000055a4e8206c40 CR3: 0000000119f06000 CR4: 00000000003506f0
    [  464.028738] Call Trace:
    [  464.028740]  <TASK>
    [  464.028746]  submit_bio+0x1b/0x80
    [  464.028748]  rnbd_srv_rdma_ev+0x50d/0x10c0 [rnbd_server]
    [  464.028754]  ? percpu_ref_get_many.constprop.0+0x55/0x140 [rtrs_server]
    [  464.028760]  ? __this_cpu_preempt_check+0x13/0x20
    [  464.028769]  process_io_req+0x1dc/0x450 [rtrs_server]
    [  464.028775]  rtrs_srv_inv_rkey_done+0x67/0xb0 [rtrs_server]
    [  464.028780]  __ib_process_cq+0xbc/0x1f0 [ib_core]
    [  464.028793]  ib_cq_poll_work+0x2b/0xa0 [ib_core]
    [  464.028804]  process_one_work+0x2a9/0x580

    [1]. https://lore.kernel.org/all/ZFHgefWofVt24tRl@infradead.org/

    Signed-off-by: Guoqing Jiang <guoqing.jiang@linux.dev>
    Reviewed-by: Christoph Hellwig <hch@lst.de>
    Reviewed-by: Chaitanya Kulkarni <kch@nvidia.com>
    Link: https://lore.kernel.org/r/20230512034631.28686-1-guoqing.jiang@linux.dev
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2ba70aa402996c3b365fd511aac410ff7cab97cc
Author: Ivan Orlov <ivan.orlov0322@gmail.com>
Date:   Fri May 12 17:05:32 2023 +0400

    nbd: Fix debugfs_create_dir error checking
```

```
    [ Upstream commit 4913cfcf014c95f0437db2df1734472fd3e15098 ]

    The debugfs_create_dir function returns ERR_PTR in case of error, and the
    only correct way to check if an error occurred is 'IS_ERR' inline function.
    This patch will replace the null-comparison with IS_ERR.

    Signed-off-by: Ivan Orlov <ivan.orlov0322@gmail.com>
    Link: https://lore.kernel.org/r/20230512130533.98709-1-ivan.orlov0322@gmail.com
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b6b3a0e571c16c05f43bbc67220038b7c6670f35
Author: Helge Deller <deller@gmx.de>
Date:   Fri May 12 11:50:33 2023 +0200

    fbdev: stifb: Fix info entry in sti_struct on error path

    [ Upstream commit 0bdf1ad8d10bd4e50a8b1a2c53d15984165f7fea ]

    Minor fix to reset the info field to NULL in case of error.

    Signed-off-by: Helge Deller <deller@gmx.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8e4aa73e787cbdc9e58ed41ccc44a48ceab2d890
Author: Helge Deller <deller@gmx.de>
Date:   Sat Apr 22 23:24:26 2023 +0200

    fbdev: modedb: Add 1920x1080 at 60 Hz video mode

    [ Upstream commit c8902258b2b8ecaa1b8d88c312853c5b14c2553d ]

    Add typical resolution for Full-HD monitors.

    Signed-off-by: Helge Deller <deller@gmx.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4c86974fb42281b8041a504d92ab341ad4697325
Author: Zheng Wang <zyytlz.wz@163.com>
Date:   Thu Apr 27 11:08:41 2023 +0800

    fbdev: imsttfb: Fix use after free bug in imsttfb_probe

    [ Upstream commit c75f5a55061091030a13fef71b9995b89bc86213 ]

    A use-after-free bug may occur if init_imstt invokes framebuffer_release
    and free the info ptr. The caller, imsttfb_probe didn't notice that and
    still keep the ptr as private data in pdev.

    If we remove the driver which will call imsttfb_remove to make cleanup,
    UAF happens.

    Fix it by return error code if bad case happens in init_imstt.

    Signed-off-by: Zheng Wang <zyytlz.wz@163.com>
    Signed-off-by: Helge Deller <deller@gmx.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c38ecb4ef8f3485f8a3cc4831d9ce0ccc5e49344
Author: Yifan Zhang <yifan1.zhang@amd.com>
Date:   Thu Apr 27 14:01:05 2023 +0800

    drm/amdgpu: set gfx9 onwards APU atomics support to be true

    [ Upstream commit af7828fbceed4f9e503034111066a0adef3db383 ]
```

```
    APUs w/ gfx9 onwards doesn't reply on PCIe atomics, rather
    it is internal path w/ native atomic support. Set have_atomics_support
    to true.

    Signed-off-by: Yifan Zhang <yifan1.zhang@amd.com>
    Reviewed-by: Lang Yu <lang.yu@amd.com>
    Acked-by: Felix Kuehling <Felix.Kuehling@amd.com>
    Acked-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit bfeaea4cb23d295e36aa967078604b426210232e
Author: Thong Thai <thong.thai@amd.com>
Date:   Mon May 1 11:04:36 2023 -0400

    drm/amdgpu/nv: update VCN 3 max HEVC encoding resolution

    [ Upstream commit 476ac50fc30540e29191615a26aaf5f9dee91c49 ]

    Update the maximum resolution reported for HEVC encoding on VCN 3
    devices to reflect its 8K encoding capability.

    v2: Also update the max height for H.264 encoding to match spec.
    (Ruijing)

    Signed-off-by: Thong Thai <thong.thai@amd.com>
    Reviewed-by: Ruijing Dong <ruijing.dong@amd.com>
    Acked-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 14c454764a37b194dc916c07488ce7339c82bc4f
Author: Bob Peterson <rpeterso@redhat.com>
Date:   Fri Apr 28 12:07:46 2023 -0400

    gfs2: Don't deref jdesc in evict

    [ Upstream commit 504a10d9e46bc37b23d0a1ae2f28973c8516e636 ]

    On corrupt gfs2 file systems the evict code can try to reference the
    journal descriptor structure, jdesc, after it has been freed and set to
    NULL. The sequence of events is:

    init_journal()
    ...
    fail_jindex:
       gfs2_jindex_free(sdp); <------frees journals, sets jdesc = NULL
          if (gfs2_holder_initialized(&ji_gh))
             gfs2_glock_dq_uninit(&ji_gh);
    fail:
       iput(sdp->sd_jindex); <--references jdesc in evict_linked_inode
          evict()
             gfs2_evict_inode()
                evict_linked_inode()
                   ret = gfs2_trans_begin(sdp, 0, sdp->sd_jdesc->jd_blocks);
    <------references the now freed/zeroed sd_jdesc pointer.

    The call to gfs2_trans_begin is done because the truncate_inode_pages
    call can cause gfs2 events that require a transaction, such as removing
    journaled data (jdata) blocks from the journal.

    This patch fixes the problem by adding a check for sdp->sd_jdesc to
    function gfs2_evict_inode. In theory, this should only happen to corrupt
    gfs2 file systems, when gfs2 detects the problem, reports it, then tries
```

```
    to evict all the system inodes it has read in up to that point.

    Reported-by: Yang Lan <lanyang0908@gmail.com>
    Signed-off-by: Bob Peterson <rpeterso@redhat.com>
    Signed-off-by: Andreas Gruenbacher <agruenba@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4d4575460fefac915aa37cdebf38a4a5df16d3e2
Author: Liming Sun <limings@nvidia.com>
Date:   Wed Apr 26 10:23:44 2023 -0400

    platform/mellanox: fix potential race in mlxbf-tmfifo driver

    [ Upstream commit 3d43f9f639542fadfb28f40b509bf147a6624d48 ]

    This commit adds memory barrier for the 'vq' update in function
    mlxbf_tmfifo_virtio_find_vqs() to avoid potential race due to
    out-of-order memory write. It also adds barrier for the 'is_ready'
    flag to make sure the initializations are visible before this flag
    is checked.

    Signed-off-by: Liming Sun <limings@nvidia.com>
    Reviewed-by: Vadim Pasternak <vadimp@nvidia.com>
    Link:
https://lore.kernel.org/r/b98c0ab61d644ba38fa9b3fd1607b138b0dd820b.1682518748.git.limings@nv
idia.com
    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 62826cee0d1a9b1f8cccafd6250207b811b3c1a7
Author: Julian Winkler <julian.winkler1@web.de>
Date:   Sun Apr 16 17:49:32 2023 +0200

    platform/x86: intel_scu_pcidrv: Add back PCI ID for Medfield

    [ Upstream commit 4a9b6850c794e4394cad99e2b863d75f5bc8e92f ]

    This id was removed in commit b47018a778c1 ("platform/x86: intel_scu_ipc:
    Remove Lincroft support"), saying it is only used on Moorestown,
    but apparently the same id is also used on Medfield.

    Tested on the Medfield based Motorola RAZR i smartphone.

    Signed-off-by: Julian Winkler <julian.winkler1@web.de>
    Link: https://lore.kernel.org/r/20230416154932.6579-1-julian.winkler1@web.de
    Reviewed-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8ea7c143c58ee6b0da216626fb12f2b090f92cdf
Author: Niklas Söderlund <niklas.soderlund+renesas@ragnatech.se>
Date:   Sat Feb 11 21:55:34 2023 +0100

    media: rcar-vin: Select correct interrupt mode for V4L2_FIELD_ALTERNATE

    [ Upstream commit e10707d5865c90d3dfe4ef589ce02ff4287fef85 ]

    When adding proper support for V4L2_FIELD_ALTERNATE it was missed that
    this field format should trigger an interrupt for each field, not just
    for the whole frame. Fix this by marking it as progressive in the
    capture setup, which will then select the correct interrupt mode.

    Tested on both Gen2 and Gen3 with the result of a doubling of the frame
    rate for V4L2_FIELD_ALTERNATE. From a PAL video source the frame rate is
    now 50, which is expected for alternate field capture.
```

```
    Signed-off-by: Niklas Söderlund <niklas.soderlund+renesas@ragnatech.se>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d54475ab7882bc5beb5414e4c3747b475984b2b7
Author: Niklas Söderlund <niklas.soderlund+renesas@ragnatech.se>
Date:   Sat Feb 11 21:54:32 2023 +0100

    media: rcar-vin: Fix NV12 size alignment

    [ Upstream commit cb88d8289fc222bd21b7a7f99b055e7e73e316f4 ]

    When doing format validation for NV12 the width and height should be
    aligned to 32 pixels.

    Signed-off-by: Niklas Söderlund <niklas.soderlund+renesas@ragnatech.se>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a24a65c066cc7dbce206296e4a3ed57bd08522e1
Author: Niklas Söderlund <niklas.soderlund+renesas@ragnatech.se>
Date:   Sat Feb 11 21:54:31 2023 +0100

    media: rcar-vin: Gen3 can not scale NV12

    [ Upstream commit 879c5a458e532b95783ce27f704d1b21573066f7 ]

    The VIN modules on Gen3 can not scale NV12, fail format validation if
    the user tries. Currently no frames are produced if this is attempted.

    Signed-off-by: Niklas Söderlund <niklas.soderlund+renesas@ragnatech.se>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9ccb9b69a08b814c7afb17504b86efe6a70e902d
Author: Mario Limonciello <mario.limonciello@amd.com>
Date:   Thu Apr 27 00:33:37 2023 -0500

    hwmon: (k10temp) Add PCI ID for family 19, model 78h

    [ Upstream commit 7d8accfaa0ab65e4282c8e58950f7d688342cd86 ]

    Enable k10temp on this system.

      [ bp: Massage. ]

    Signed-off-by: Mario Limonciello <mario.limonciello@amd.com>
    Signed-off-by: Borislav Petkov (AMD) <bp@alien8.de>
    Acked-by: Guenter Roeck <linux@roeck-us.net>
    Link: https://lore.kernel.org/r/20230427053338.16653-3-mario.limonciello@amd.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a9165207b2b07415eeb01b3ac8bb84976ec96984
Author: Kan Liang <kan.liang@linux.intel.com>
Date:   Fri Apr 21 11:45:28 2023 -0700

    perf/x86/intel/ds: Flush PEBS DS when changing PEBS_DATA_CFG

    [ Upstream commit b752ea0c28e3f7f0aaaad6abf84f735eebc37a60 ]

    Several similar kernel warnings can be triggered,

      [56605.607840] CPU0 PEBS record size 0, expected 32, config 0 cpuc->record_size=208
```

```
    when the below commands are running in parallel for a while on SPR.

      while true;
      do
            perf record --no-buildid -a --intr-regs=AX  \
                        -e cpu/event=0xd0,umask=0x81/pp \
                        -c 10003 -o /dev/null ./triad;
      done &

      while true;
      do
            perf record -o /tmp/out -W -d \
                        -e '{ld_blocks.store_forward:period=1000000, \
                             MEM_TRANS_RETIRED.LOAD_LATENCY:u:precise=2:ldlat=4}' \
                        -c 1037 ./triad;
      done

    The triad program is just the generation of loads/stores.

    The warnings are triggered when an unexpected PEBS record (with a
    different config and size) is found.

    A system-wide PEBS event with the large PEBS config may be enabled
    during a context switch. Some PEBS records for the system-wide PEBS
    may be generated while the old task is sched out but the new one
    hasn't been sched in yet. When the new task is sched in, the
    cpuc->pebs_record_size may be updated for the per-task PEBS events. So
    the existing system-wide PEBS records have a different size from the
    later PEBS records.

    The PEBS buffer should be flushed right before the hardware is
    reprogrammed. The new size and threshold should be updated after the
    old buffer has been flushed.

    Reported-by: Stephane Eranian <eranian@google.com>
    Suggested-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Signed-off-by: Kan Liang <kan.liang@linux.intel.com>
    Signed-off-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Link: https://lkml.kernel.org/r/20230421184529.3320912-1-kan.liang@linux.intel.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 74b28f6caf8a7f3f827067e1babc5d0437ec6b88
Author: Haibo Li <haibo.li@mediatek.com>
Date:   Mon Apr 17 10:17:07 2023 +0100

    ARM: 9295/1: unwind:fix unwind abort for uleb128 case

    [ Upstream commit fa3eeb638de0c1a9d2d860e5b48259facdd65176 ]

    When unwind instruction is 0xb2,the subsequent instructions
    are uleb128 bytes.
    For now,it uses only the first uleb128 byte in code.

    For vsp increments of 0x204~0x400,use one uleb128 byte like below:
    0xc06a00e4 <unwind_test_work>: 0x80b27fac
      Compact model index: 0
      0xb2 0x7f vsp = vsp + 1024
      0xac      pop {r4, r5, r6, r7, r8, r14}

    For vsp increments larger than 0x400,use two uleb128 bytes like below:
    0xc06a00e4 <unwind_test_work>: @0xc0cc9e0c
      Compact model index: 1
      0xb2 0x81 0x01 vsp = vsp + 1032
      0xac      pop {r4, r5, r6, r7, r8, r14}
    The unwind works well since the decoded uleb128 byte is also 0x81.
```

```
    For vsp increments larger than 0x600,use two uleb128 bytes like below:
    0xc06a00e4 <unwind_test_work>: @0xc0cc9e0c
      Compact model index: 1
      0xb2 0x81 0x02 vsp = vsp + 1544
      0xac      pop {r4, r5, r6, r7, r8, r14}
    In this case,the decoded uleb128 result is 0x101(vsp=0x204+(0x101<<2)).
    While the uleb128 used in code is 0x81(vsp=0x204+(0x81<<2)).
    The unwind aborts at this frame since it gets incorrect vsp.

    To fix this,add uleb128 decode to cover all the above case.

    Signed-off-by: Haibo Li <haibo.li@mediatek.com>
    Reviewed-by: Linus Walleij <linus.walleij@linaro.org>
    Reviewed-by: Alexandre Mergnat <amergnat@baylibre.com>
    Reviewed-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Signed-off-by: Russell King (Oracle) <rmk+kernel@armlinux.org.uk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 763f0c269f8aca6ab5b8d6211d7f17122b5b5d3e
Author: Filipe Manana <fdmanana@suse.com>
Date:   Wed Apr 26 11:51:35 2023 +0100

    btrfs: abort transaction when sibling keys check fails for leaves

    [ Upstream commit 9ae5afd02a03d4e22a17a9609b19400b77c36273 ]

    If the sibling keys check fails before we move keys from one sibling
    leaf to another, we are not aborting the transaction - we leave that to
    some higher level caller of btrfs_search_slot() (or anything else that
    uses it to insert items into a b+tree).

    This means that the transaction abort will provide a stack trace that
    omits the b+tree modification call chain. So change this to immediately
    abort the transaction and therefore get a more useful stack trace that
    shows us the call chain in the bt+tree modification code.

    It's also important to immediately abort the transaction just in case
    some higher level caller is not doing it, as this indicates a very
    serious corruption and we should stop the possibility of doing further
    damage.

    Reviewed-by: Qu Wenruo <wqu@suse.com>
    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: Sasha Levin <sashal@kernel.org>

commit fa048ec12a11c59c24a7d3500b650f200ce2bb3e
Author: Jammy Huang <jammy_huang@aspeedtech.com>
Date:   Fri Apr 21 08:33:54 2023 +0800

    drm/ast: Fix ARM compatibility

    [ Upstream commit 4327a6137ed43a091d900b1ac833345d60f32228 ]

    ARM architecture only has 'memory', so all devices are accessed by
    MMIO if possible.

    Signed-off-by: Jammy Huang <jammy_huang@aspeedtech.com>
    Reviewed-by: Thomas Zimmermann <tzimmermann@suse.de>
    Signed-off-by: Thomas Zimmermann <tzimmermann@suse.de>
    Link: https://patchwork.freedesktop.org/patch/msgid/20230421003354.27767-1-
jammy_huang@aspeedtech.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 54fb4a623722430e323d1a426e7d65ccac178fab
Author: Lee Jones <lee@kernel.org>
Date:   Thu Apr 20 08:27:18 2023 +0100

    mailbox: mailbox-test: Fix potential double-free in mbox_test_message_write()

    [ Upstream commit 2d1e952a2b8e5e92d8d55ac88a7cf7ca5ea591ad ]

    If a user can make copy_from_user() fail, there is a potential for
    UAF/DF due to a lack of locking around the allocation, use and freeing
    of the data buffers.

    This issue is not theoretical.  I managed to author a POC for it:

        BUG: KASAN: double-free in kfree+0x5c/0xac
        Free of addr ffff29280be5de00 by task poc/356
        CPU: 1 PID: 356 Comm: poc Not tainted 6.1.0-00001-g961aa6552c04-dirty #20
        Hardware name: linux,dummy-virt (DT)
        Call trace:
         dump_backtrace.part.0+0xe0/0xf0
         show_stack+0x18/0x40
         dump_stack_lvl+0x64/0x80
         print_report+0x188/0x48c
         kasan_report_invalid_free+0xa0/0xc0
         ____kasan_slab_free+0x174/0x1b0
         __kasan_slab_free+0x18/0x24
         __kmem_cache_free+0x130/0x2e0
         kfree+0x5c/0xac
         mbox_test_message_write+0x208/0x29c
         full_proxy_write+0x90/0xf0
         vfs_write+0x154/0x440
         ksys_write+0xcc/0x180
         __arm64_sys_write+0x44/0x60
         invoke_syscall+0x60/0x190
         el0_svc_common.constprop.0+0x7c/0x160
         do_el0_svc+0x40/0xf0
         el0_svc+0x2c/0x6c
         el0t_64_sync_handler+0xf4/0x120
         el0t_64_sync+0x18c/0x190

        Allocated by task 356:
         kasan_save_stack+0x3c/0x70
         kasan_set_track+0x2c/0x40
         kasan_save_alloc_info+0x24/0x34
         __kasan_kmalloc+0xb8/0xc0
         kmalloc_trace+0x58/0x70
         mbox_test_message_write+0x6c/0x29c
         full_proxy_write+0x90/0xf0
         vfs_write+0x154/0x440
         ksys_write+0xcc/0x180
         __arm64_sys_write+0x44/0x60
         invoke_syscall+0x60/0x190
         el0_svc_common.constprop.0+0x7c/0x160
         do_el0_svc+0x40/0xf0
         el0_svc+0x2c/0x6c
         el0t_64_sync_handler+0xf4/0x120
         el0t_64_sync+0x18c/0x190

        Freed by task 357:
         kasan_save_stack+0x3c/0x70
         kasan_set_track+0x2c/0x40
         kasan_save_free_info+0x38/0x5c
         ____kasan_slab_free+0x13c/0x1b0
         __kasan_slab_free+0x18/0x24
```

```
          __kmem_cache_free+0x130/0x2e0
          kfree+0x5c/0xac
          mbox_test_message_write+0x208/0x29c
          full_proxy_write+0x90/0xf0
          vfs_write+0x154/0x440
          ksys_write+0xcc/0x180
          __arm64_sys_write+0x44/0x60
          invoke_syscall+0x60/0x190
          el0_svc_common.constprop.0+0x7c/0x160
          do_el0_svc+0x40/0xf0
          el0_svc+0x2c/0x6c
          el0t_64_sync_handler+0xf4/0x120
          el0t_64_sync+0x18c/0x190

    Signed-off-by: Lee Jones <lee@kernel.org>
    Signed-off-by: Jassi Brar <jaswinder.singh@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 15994ba65f9ca77598b8b5e7ab5837ae919900c9
Author: lyndonli <Lyndon.Li@amd.com>
Date:   Sun Apr 23 17:05:15 2023 +0800

    drm/amdgpu: Use the default reset when loading or reloading the driver

    [ Upstream commit 4eea7fb980dc44545a32eec92e2662053b34cd9d ]

    Below call trace and errors are observed when reloading
    amdgpu driver with the module parameter reset_method=3.

    It should do a default reset when loading or reloading the
    driver, regardless of the module parameter reset_method.

    v2: add comments inside and modify commit messages.

    [  +2.180243] [drm] psp gfx command ID_LOAD_TOC(0x20) failed
    and response status is (0x0)
    [  +0.000011] [drm:psp_hw_start [amdgpu]] *ERROR* Failed to load toc
    [  +0.000890] [drm:psp_hw_start [amdgpu]] *ERROR* PSP tmr init failed!
    [  +0.020683] [drm:amdgpu_fill_buffer [amdgpu]] *ERROR* Trying to
    clear memory with ring turned off.
    [  +0.000003] RIP: 0010:amdgpu_bo_release_notify+0x1ef/0x210 [amdgpu]
    [  +0.000004] Call Trace:
    [  +0.000003]  <TASK>
    [  +0.000008]  ttm_bo_release+0x2c4/0x330 [amdttm]
    [  +0.000026]  amdttm_bo_put+0x3c/0x70 [amdttm]
    [  +0.000020]  amdgpu_bo_free_kernel+0xe6/0x140 [amdgpu]
    [  +0.000728]  psp_v11_0_ring_destroy+0x34/0x60 [amdgpu]
    [  +0.000826]  psp_hw_init+0xe7/0x2f0 [amdgpu]
    [  +0.000813]  amdgpu_device_fw_loading+0x1ad/0x2d0 [amdgpu]
    [  +0.000731]  amdgpu_device_init.cold+0x108e/0x2002 [amdgpu]
    [  +0.001071]  ? do_pci_enable_device+0xe1/0x110
    [  +0.000011]  amdgpu_driver_load_kms+0x1a/0x160 [amdgpu]
    [  +0.000729]  amdgpu_pci_probe+0x179/0x3a0 [amdgpu]

    Signed-off-by: lyndonli <Lyndon.Li@amd.com>
    Signed-off-by: Yunxiang Li <Yunxiang.Li@amd.com>
    Reviewed-by: Feifei Xu <Feifei.Xu@amd.com>
    Reviewed-by: Kenneth Feng <kenneth.feng@amd.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 675f7c801c3c055a6065158ac74bd09c6c6c2dc2
Author: Hans de Goede <hdegoede@redhat.com>
Date:   Sat Apr 29 12:47:21 2023 +0200
```

```
    ASoC: Intel: soc-acpi-cht: Add quirk for Nextbook Ares 8A tablet

    [ Upstream commit ec6f82b4c63cc68f8dc03316e725106d242706be ]

    The Nextbook Ares 8A tablet which has Android as factory OS, has a buggy
    DSDT with both ESSX8316 and 10EC5651 ACPI devices.

    This tablet actually uses an rt5651 codec, but the matching code ends up
    picking the ESSX8316 device, add a quirk to ignote the ESSX8316 device
    on this tablet.

    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Acked-by: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com>
    Message-Id: <20230429104721.7176-1-hdegoede@redhat.com>
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cc1a832c63b36ea751b899566282538849067a85
Author: Qing Zhang <zhangqing@loongson.cn>
Date:   Mon May 1 17:19:52 2023 +0800

    LoongArch: Add ARCH_HAS_FORTIFY_SOURCE selection

    [ Upstream commit d4c937c2a57bbba24790be6fe7a791456f5fbb60 ]

    FORTIFY_SOURCE could detect various overflows at compile and run time.
    ARCH_HAS_FORTIFY_SOURCE means that the architecture can be built and run
    with CONFIG_FORTIFY_SOURCE. So select it in LoongArch.

    See more about this feature from commit 6974f0c4555e285 ("include/linux/
    string.h: add the option of fortified string.h functions").

    Signed-off-by: Qing Zhang <zhangqing@loongson.cn>
    Signed-off-by: Huacai Chen <chenhuacai@loongson.cn>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 110b83617da833a36a947afe9d7751879eceebe9
Author: Hersen Wu <hersenxs.wu@amd.com>
Date:   Mon Mar 27 09:10:48 2023 -0400

    drm/amd/display: fix memleak in aconnector->timing_requested

    [ Upstream commit 025ce392b5f213696ca0af3e07735d0fae020694 ]

    [Why]
    when amdgpu_dm_update_connector_after_detect is called
    two times successively with valid sink, memory allocated of
    aconnector->timing_requested for the first call is not free.
    this causes memeleak.

    [How]
    allocate memory only when aconnector->timing_requested
    is null.

    Reviewed-by: Qingqing Zhuo <Qingqing.Zhuo@amd.com>
    Acked-by: Qingqing Zhuo <qingqing.zhuo@amd.com>
    Signed-off-by: Hersen Wu <hersenxs.wu@amd.com>
    Tested-by: Daniel Wheeler <daniel.wheeler@amd.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit fe9c3d0e76a7134e000ec92893148680e61fb4a7
Author: jasontao <jasontao@glenfly.com>
Date:   Wed Apr 26 09:30:59 2023 +0800
```

```
    ALSA: hda: Glenfly: add HD Audio PCI IDs and HDMI Codec Vendor IDs.

    [ Upstream commit c51e431052e2eacfb23fbf6b39bc6c8770d9827a ]

    Add a set of HD Audio PCI IDS, and the HDMI codec vendor IDs for
    Glenfly Gpus.

    - In default_bdl_pos_adj, set bdl to 128 as Glenfly Gpus have hardware
    limitation, need to increase hdac interrupt interval.
    - In azx_first_init, enable polling mode for Glenfly Gpu. When the codec
    complete the command, it sends interrupt and writes response entries to
    memory, howerver, the write requests sometimes are not actually
    synchronized to memory when driver handle hdac interrupt on Glenfly Gpus.
    If the RIRB status is not updated in the interrupt handler,
    azx_rirb_get_response keeps trying to recevie a response from rirb until
    1s timeout. Enabling polling mode for Glenfly Gpu can fix the issue.
    - In patch_gf_hdmi, set Glenlfy Gpu Codec's no_sticky_stream as it need
    driver to do actual clean-ups for the linked codec when switch from one
    codec to another.

    Signed-off-by: jasontao <jasontao@glenfly.com>
    Signed-off-by: Reaper Li <reaperlioc@glenfly.com>
    Link: https://lore.kernel.org/r/20230426013059.4329-1-reaperlioc@glenfly.com
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5e8d42514bf922f69caeb5cd59207ca6f9b39bce
Author: Johannes Thumshirn <jth@kernel.org>
Date:   Tue Apr 18 19:25:30 2023 +0200

    watchdog: menz069_wdt: fix watchdog initialisation

    [ Upstream commit 87b22656ca6a896d0378e9e60ffccb0c82f48b08 ]

    Doing a 'cat /dev/watchdog0' with menz069_wdt as watchdog0 will result in
    a NULL pointer dereference.

    This happens because we're passing the wrong pointer to
    watchdog_register_device(). Fix this by getting rid of the static
    watchdog_device structure and use the one embedded into the driver's
    per-instance private data.

    Signed-off-by: Johannes Thumshirn <jth@kernel.org>
    Reviewed-by: Guenter Roeck <linux@roeck-us.net>
    Link: https://lore.kernel.org/r/20230418172531.177349-2-jth@kernel.org
    Signed-off-by: Guenter Roeck <linux@roeck-us.net>
    Signed-off-by: Wim Van Sebroeck <wim@linux-watchdog.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b768aadce9071e421f3deffe532f5bcd6cda0699
Author: Chong Li <chongli2@amd.com>
Date:   Fri Apr 14 13:51:19 2023 +0800

    drm/amdgpu: release gpu full access after "amdgpu_device_ip_late_init"

    [ Upstream commit 38eecbe086a4e52f54b2bbda8feba65d44addbef ]

    [WHY]
     Function "amdgpu_irq_update()" called by "amdgpu_device_ip_late_init()" is an atomic
context.
     We shouldn't access registers through KIQ since "msleep()" may be called in
"amdgpu_kiq_rreg()".

    [HOW]
     Move function "amdgpu_virt_release_full_gpu()" after function
```

```
  "amdgpu_device_ip_late_init()",
       to ensure that registers be accessed through RLCG instead of KIQ.

      Call Trace:
        <TASK>
        show_stack+0x52/0x69
        dump_stack_lvl+0x49/0x6d
        dump_stack+0x10/0x18
        __schedule_bug.cold+0x4f/0x6b
        __schedule+0x473/0x5d0
        ? __wake_up_klogd.part.0+0x40/0x70
        ? vprintk_emit+0xbe/0x1f0
        schedule+0x68/0x110
        schedule_timeout+0x87/0x160
        ? timer_migration_handler+0xa0/0xa0
        msleep+0x2d/0x50
        amdgpu_kiq_rreg+0x18d/0x1f0 [amdgpu]
        amdgpu_device_rreg.part.0+0x59/0xd0 [amdgpu]
        amdgpu_device_rreg+0x3a/0x50 [amdgpu]
        amdgpu_sriov_rreg+0x3c/0xb0 [amdgpu]
        gfx_v10_0_set_gfx_eop_interrupt_state.constprop.0+0x16c/0x190 [amdgpu]
        gfx_v10_0_set_eop_interrupt_state+0xa5/0xb0 [amdgpu]
        amdgpu_irq_update+0x53/0x80 [amdgpu]
        amdgpu_irq_get+0x7c/0xb0 [amdgpu]
        amdgpu_fence_driver_hw_init+0x58/0x90 [amdgpu]
        amdgpu_device_init.cold+0x16b7/0x2022 [amdgpu]

      Signed-off-by: Chong Li <chongli2@amd.com>
      Reviewed-by: JingWen.Chen2@amd.com
      Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 290ae8751cc575e70e30878a1ef363c61f829cc4
  Author: Paolo Abeni <pabeni@redhat.com>
  Date:   Wed May 31 12:37:07 2023 -0700

      mptcp: add annotations around sk->sk_shutdown accesses

      [ Upstream commit 6b9831bfd9322b297eb6d44257808cc055fdc586 ]

      Christoph reported the mptcp variant of a recently addressed plain
      TCP issue. Similar to commit e14cadfd80d7 ("tcp: add annotations around
      sk->sk_shutdown accesses") add READ/WRITE ONCE annotations to silence
      KCSAN reports around lockless sk_shutdown access.

      Fixes: 71ba088ce0aa ("mptcp: cleanup accept and poll")
      Reported-by: Christoph Paasch <cpaasch@apple.com>
      Closes: https://github.com/multipath-tcp/mptcp_net-next/issues/401
      Reviewed-by: Mat Martineau <martineau@kernel.org>
      Signed-off-by: Paolo Abeni <pabeni@redhat.com>
      Signed-off-by: Mat Martineau <martineau@kernel.org>
      Signed-off-by: Jakub Kicinski <kuba@kernel.org>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 0ee88e82a2604b19e9d5ee75e0f3a4a5ebc29a04
  Author: Paolo Abeni <pabeni@redhat.com>
  Date:   Wed May 31 12:37:06 2023 -0700

      mptcp: fix data race around msk->first access

      [ Upstream commit 1b1b43ee7a208096ecd79e626f2fc90d4a321111 ]

      The first subflow socket is accessed outside the msk socket lock
      by mptcp_subflow_fail(), we need to annotate each write access
      with WRITE_ONCE, but a few spots still lacks it.
```

```
    Fixes: 76a13b315709 ("mptcp: invoke MP_FAIL response when needed")
    Reviewed-by: Mat Martineau <martineau@kernel.org>
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Mat Martineau <martineau@kernel.org>
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 75373b1a2330fae4b09d2cb3f8e8787c80c39eae
Author: Paolo Abeni <pabeni@redhat.com>
Date:   Wed May 31 12:37:05 2023 -0700

    mptcp: consolidate passive msk socket initialization

    [ Upstream commit 7e8b88ec35eef363040e08d99536d2bebef83774 ]

    When the msk socket is cloned at MPC handshake time, a few
    fields are initialized in a racy way outside mptcp_sk_clone()
    and the msk socket lock.

    The above is due historical reasons: before commit a88d0092b24b
    ("mptcp: simplify subflow_syn_recv_sock()") as the first subflow socket
    carrying all the needed date was not available yet at msk creation
    time

    We can now refactor the code moving the missing initialization bit
    under the socket lock, removing the init race and avoiding some
    code duplication.

    This will also simplify the next patch, as all msk->first write
    access are now under the msk socket lock.

    Fixes: 0397c6d85f9c ("mptcp: keep unaccepted MPC subflow into join list")
    Reviewed-by: Mat Martineau <martineau@kernel.org>
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Mat Martineau <martineau@kernel.org>
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1c8bb67514962154e44f381c8727d161d99d873b
Author: Paolo Abeni <pabeni@redhat.com>
Date:   Mon Mar 27 12:22:22 2023 +0200

    mptcp: simplify subflow_syn_recv_sock()

    [ Upstream commit a88d0092b24b8cddce57fe0e88e60a9e29e0b515 ]

    Postpone the msk cloning to the child process creation
    so that we can avoid a bunch of conditionals.

    Link: https://github.com/multipath-tcp/mptcp_net-next/issues/61
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Reviewed-by: Matthieu Baerts <matthieu.baerts@tessares.net>
    Signed-off-by: Matthieu Baerts <matthieu.baerts@tessares.net>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Stable-dep-of: 7e8b88ec35ee ("mptcp: consolidate passive msk socket initialization")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2078521546b2fcd967a2e710506e1cba429727fe
Author: Paolo Abeni <pabeni@redhat.com>
Date:   Mon Mar 27 12:22:21 2023 +0200

    mptcp: avoid unneeded address copy

    [ Upstream commit 2bb9a37f0e194ed95c70603b0efc7898a5a0d9b4 ]
```

```
    In the syn_recv fallback path, the msk is unused. We can skip
    setting the socket address.

    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Reviewed-by: Matthieu Baerts <matthieu.baerts@tessares.net>
    Signed-off-by: Matthieu Baerts <matthieu.baerts@tessares.net>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Stable-dep-of: 7e8b88ec35ee ("mptcp: consolidate passive msk socket initialization")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ddc8e3e24ca3ebbe82fee63b0fa37e7e1321fb6d
Author: Paolo Abeni <pabeni@redhat.com>
Date:   Wed May 31 12:37:04 2023 -0700

    mptcp: add annotations around msk->subflow accesses

    [ Upstream commit 5b825727d0871b23e8867f6371183e61628b4a26 ]

    The MPTCP can access the first subflow socket in a few spots
    outside the socket lock scope. That is actually safe, as MPTCP
    will delete the socket itself only after the msk sock close().

    Still the such accesses causes a few KCSAN splats, as reported
    by Christoph. Silence the harmless warning adding a few annotation
    around the relevant accesses.

    Fixes: 71ba088ce0aa ("mptcp: cleanup accept and poll")
    Reported-by: Christoph Paasch <cpaasch@apple.com>
    Closes: https://github.com/multipath-tcp/mptcp_net-next/issues/402
    Reviewed-by: Mat Martineau <martineau@kernel.org>
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Mat Martineau <martineau@kernel.org>
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f05182da4b6bc737f7af24b79bcb6c2273efe4fc
Author: Paolo Abeni <pabeni@redhat.com>
Date:   Fri Apr 14 16:08:01 2023 +0200

    mptcp: avoid unneeded __mptcp_nmpc_socket() usage

    [ Upstream commit 617612316953093bc859890e405e1b550c27d840 ]

    In a few spots, the mptcp code invokes the __mptcp_nmpc_socket() helper
    multiple times under the same socket lock scope. Additionally, in such
    places, the socket status ensures that there is no MP capable handshake
    running.

    Under the above condition we can replace the later __mptcp_nmpc_socket()
    helper invocation with direct access to the msk->subflow pointer and
    better document such access is not supposed to fail with WARN().

    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Reviewed-by: Matthieu Baerts <matthieu.baerts@tessares.net>
    Signed-off-by: Matthieu Baerts <matthieu.baerts@tessares.net>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Stable-dep-of: 5b825727d087 ("mptcp: add annotations around msk->subflow accesses")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9b3da191c0c77fee78c96f47b83c9a6b6b3d652e
Author: Xin Long <lucien.xin@gmail.com>
Date:   Wed May 31 12:01:44 2023 -0400

    rtnetlink: add the missing IFLA_GRO_ tb check in validate_linkmsg
```

```
    [ Upstream commit 65d6914e253f3d83b724a9bbfc889ae95711e512 ]

    This fixes the issue that dev gro_max_size and gso_ipv4_max_size
    can be set to a huge value:

      # ip link add dummy1 type dummy
      # ip link set dummy1 gro_max_size 4294967295
      # ip -d link show dummy1
        dummy addrgenmode eui64 ... gro_max_size 4294967295

    Fixes: 0fe79f28bfaf ("net: allow gro_max_size to exceed 65536")
    Fixes: 9eefedd58ae1 ("net: add gso_ipv4_max_size and gro_ipv4_max_size per device")
    Reported-by: Xiumei Mu <xmu@redhat.com>
    Signed-off-by: Xin Long <lucien.xin@gmail.com>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7bfab44e1346ff8511aa929f47c9f90332ece9f5
Author: Xin Long <lucien.xin@gmail.com>
Date:   Wed May 31 12:01:43 2023 -0400

    rtnetlink: move IFLA_GSO_ tb check to validate_linkmsg

    [ Upstream commit fef5b228dd38378148bc850f7e69a7783f3b95a4 ]

    These IFLA_GSO_* tb check should also be done for the new created link,
    otherwise, they can be set to a huge value when creating links:

      # ip link add dummy1 gso_max_size 4294967295 type dummy
      # ip -d link show dummy1
        dummy addrgenmode eui64 ... gso_max_size 4294967295

    Fixes: 46e6b992c250 ("rtnetlink: allow GSO maximums to be set on device creation")
    Fixes: 9eefedd58ae1 ("net: add gso_ipv4_max_size and gro_ipv4_max_size per device")
    Signed-off-by: Xin Long <lucien.xin@gmail.com>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7a622f0c56c25052e52d15bff3529caa81450200
Author: Xin Long <lucien.xin@gmail.com>
Date:   Wed May 31 12:01:42 2023 -0400

    rtnetlink: call validate_linkmsg in rtnl_create_link

    [ Upstream commit b0ad3c179059089d809b477a1d445c1183a7b8fe ]

    validate_linkmsg() was introduced by commit 1840bb13c22f5b ("[RTNL]:
    Validate hardware and broadcast address attribute for RTM_NEWLINK")
    to validate tb[IFLA_ADDRESS/BROADCAST] for existing links. The same
    check should also be done for newly created links.

    This patch adds validate_linkmsg() call in rtnl_create_link(), to
    avoid the invalid address set when creating some devices like:

      # ip link add dummy0 type dummy
      # ip link add link dummy0 name mac0 address 01:02 type macsec

    Fixes: 0e06877c6fdb ("[RTNETLINK]: rtnl_link: allow specifying initial device address")
    Signed-off-by: Xin Long <lucien.xin@gmail.com>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 9e016671cbd29944cf9fb8d762a51f5a0439739e
Author: Maciej Fijalkowski <maciej.fijalkowski@intel.com>
Date:   Wed May 31 08:44:57 2023 -0700

    ice: recycle/free all of the fragments from multi-buffer frame

    [ Upstream commit abaf8d51b0cedb16af51fb6b2189370d7515977c ]

    The ice driver caches next_to_clean value at the beginning of
    ice_clean_rx_irq() in order to remember the first buffer that has to be
    freed/recycled after main Rx processing loop. The end boundary is
    indicated by first descriptor of frame that Rx processing loop has ended
    its duties. Note that if mentioned loop ended in the middle of gathering
    multi-buffer frame, next_to_clean would be pointing to the descriptor in
    the middle of the frame BUT freeing/recycling stage will stop at the
    first descriptor. This means that next iteration of ice_clean_rx_irq()
    will miss the (first_desc, next_to_clean - 1) entries.

     When running various 9K MTU workloads, such splats were observed:

    [  540.780716] BUG: kernel NULL pointer dereference, address: 0000000000000000
    [  540.787787] #PF: supervisor read access in kernel mode
    [  540.793002] #PF: error_code(0x0000) - not-present page
    [  540.798218] PGD 0 P4D 0
    [  540.800801] Oops: 0000 [#1] PREEMPT SMP NOPTI
    [  540.805231] CPU: 18 PID: 3984 Comm: xskxceiver Tainted: G        W         6.3.0-
rc7+ #96
    [  540.813619] Hardware name: Intel Corporation S2600WFT/S2600WFT, BIOS
SE5C620.86B.02.01.0008.031920191559 03/19/2019
    [  540.824209] RIP: 0010:ice_clean_rx_irq+0x2b6/0xf00 [ice]
    [  540.829678] Code: 74 24 10 e9 aa 00 00 00 8b 55 78 41 31 57 10 41 09 c4 4d 85 ff 0f
84 83 00 00 00 49 8b 57 08 41 8b 4f 1c 65 8b 35 1a fa 4b 3f <48> 8b 02 48 c1 e8 3a 39 c6 0f
85 a2 00 00 00 f6 42 08 02 0f 85 98
    [  540.848717] RSP: 0018:ffffc9000f42fc50 EFLAGS: 00010282
    [  540.854029] RAX: 0000000000000004 RBX: 0000000000000002 RCX: 000000000000fffe
    [  540.861272] RDX: 0000000000000000 RSI: 0000000000000001 RDI: 00000000ffffffff
    [  540.868519] RBP: ffff88984a05ac00 R08: 0000000000000000 R09: dead000000000100
    [  540.875760] R10: ffff88983fffcd00 R11: 000000000010f2b8 R12: 0000000000000004
    [  540.883008] R13: 0000000000000003 R14: 0000000000000800 R15: ffff889847a10040
    [  540.890253] FS:  00007f6ddf7fe640(0000) GS:ffff88afdf800000(0000)
knlGS:0000000000000000
    [  540.898465] CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    [  540.904299] CR2: 0000000000000000 CR3: 000000010d3da001 CR4: 0000000007706e0
    [  540.911542] DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000
    [  540.918789] DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400
    [  540.926032] PKRU: 55555554
    [  540.928790] Call Trace:
    [  540.931276]  <TASK>
    [  540.933418]  ice_napi_poll+0x4ca/0x6d0 [ice]
    [  540.937804]  ? __pfx_ice_napi_poll+0x10/0x10 [ice]
    [  540.942716]  napi_busy_loop+0xd7/0x320
    [  540.946537]  xsk_recvmsg+0x143/0x170
    [  540.950178]  sock_recvmsg+0x99/0xa0
    [  540.953729]  __sys_recvfrom+0xa8/0x120
    [  540.957543]  ? do_futex+0xbd/0x1d0
    [  540.961008]  ? __x64_sys_futex+0x73/0x1d0
    [  540.965083]  __x64_sys_recvfrom+0x20/0x30
    [  540.969155]  do_syscall_64+0x38/0x90
    [  540.972796]  entry_SYSCALL_64_after_hwframe+0x72/0xdc
    [  540.977934] RIP: 0033:0x7f6de5f27934

    To fix this, set cached_ntc to first_desc so that at the end, when
    freeing/recycling buffers, descriptors from first to ntc are not missed.
```

```
    Fixes: 2fba7dc5157b ("ice: Add support for XDP multi-buffer on Rx side")
    Signed-off-by: Maciej Fijalkowski <maciej.fijalkowski@intel.com>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Tested-by: Chandan Kumar Rout <chandanx.rout@intel.com> (A Contingent Worker at Intel)
    Signed-off-by: Tony Nguyen <anthony.l.nguyen@intel.com>
    Link: https://lore.kernel.org/r/20230531154457.3216621-1-anthony.l.nguyen@intel.com
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c9e14b19e30067d63427812a31ddaaa422a4fa6e
Author: Xu Liang <lxu@maxlinear.com>
Date:   Wed May 31 15:48:22 2023 +0800

    net: phy: mxl-gpy: extend interrupt fix to all impacted variants

    [ Upstream commit 519d6487640835d19461817c75907e6308074a73 ]

    The interrupt fix in commit 97a89ed101bb should be applied on all variants
    of GPY2xx PHY and GPY115C.

    Fixes: 97a89ed101bb ("net: phy: mxl-gpy: disable interrupts on GPY215 by default")
    Signed-off-by: Xu Liang <lxu@maxlinear.com>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Link: https://lore.kernel.org/r/20230531074822.39136-1-lxu@maxlinear.com
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f7e7b8203084ac2a368e7547eddac5866895e22b
Author: Yoshihiro Shimoda <yoshihiro.shimoda.uh@renesas.com>
Date:   Mon May 29 16:38:17 2023 +0900

    net: renesas: rswitch: Fix return value in error path of xmit

    [ Upstream commit a60caf039e96d806b1ced893242bae82ba3ccf0d ]

    Fix return value in the error path of rswitch_start_xmit(). If TX
    queues are full, this function should return NETDEV_TX_BUSY.

    Fixes: 3590918b5d07 ("net: ethernet: renesas: Add support for "Ethernet Switch"")
    Signed-off-by: Yoshihiro Shimoda <yoshihiro.shimoda.uh@renesas.com>
    Link: https://lore.kernel.org/r/20230529073817.1145208-1-
yoshihiro.shimoda.uh@renesas.com
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7a5427ae3f1da6c61e38060a6c1865b0ab2e8f43
Author: Chris Packham <chris.packham@alliedtelesis.co.nz>
Date:   Thu May 25 12:31:53 2023 +1200

    mtd: rawnand: marvell: don't set the NAND frequency select

    [ Upstream commit c4d28e30a8d0b979e4029465ab8f312ab6ce2644 ]

    marvell_nfc_setup_interface() uses the frequency retrieved from the
    clock associated with the nand interface to determine the timings that
    will be used. By changing the NAND frequency select without reflecting
    this in the clock configuration this means that the timings calculated
    don't correctly meet the requirements of the NAND chip. This hasn't been
    an issue up to now because of a different bug that was stopping the
    timings being updated after they were initially set.

    Fixes: b25251414f6e ("mtd: rawnand: marvell: Stop implementing ->select_chip()")
    Signed-off-by: Chris Packham <chris.packham@alliedtelesis.co.nz>
    Signed-off-by: Miquel Raynal <miquel.raynal@bootlin.com>
    Link: https://lore.kernel.org/linux-mtd/20230525003154.2303012-2-
```

```
chris.packham@alliedtelesis.co.nz
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 86859c2215e4c38afb9d5deaf915fdfe3de22d7c
Author: Chris Packham <chris.packham@alliedtelesis.co.nz>
Date:   Thu May 25 12:31:52 2023 +1200

    mtd: rawnand: marvell: ensure timing values are written

    [ Upstream commit 8a6f4d346f3bad9c68b4a87701eb3f7978542d57 ]

    When new timing values are calculated in marvell_nfc_setup_interface()
    ensure that they will be applied in marvell_nfc_select_target() by
    clearing the selected_chip pointer.

    Fixes: b25251414f6e ("mtd: rawnand: marvell: Stop implementing ->select_chip()")
    Suggested-by: Miquel Raynal <miquel.raynal@bootlin.com>
    Signed-off-by: Chris Packham <chris.packham@alliedtelesis.co.nz>
    Signed-off-by: Miquel Raynal <miquel.raynal@bootlin.com>
    Link: https://lore.kernel.org/linux-mtd/20230525003154.2303012-1-
chris.packham@alliedtelesis.co.nz
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 65094d844b125a30abf28ee71047e4db15931234
Author: Andreas Svensson <andreas.svensson@axis.com>
Date:   Tue May 30 16:52:23 2023 +0200

    net: dsa: mv88e6xxx: Increase wait after reset deactivation

    [ Upstream commit 3c27f3d53d588618d81d30d6712459a3cc9489b8 ]

    A switch held in reset by default needs to wait longer until we can
    reliably detect it.

    An issue was observed when testing on the Marvell 88E6393X (Link Street).
    The driver failed to detect the switch on some upstarts. Increasing the
    wait time after reset deactivation solves this issue.

    The updated wait time is now also the same as the wait time in the
    mv88e6xxx_hardware_reset function.

    Fixes: 7b75e49de424 ("net: dsa: mv88e6xxx: wait after reset deactivation")
    Signed-off-by: Andreas Svensson <andreas.svensson@axis.com>
    Reviewed-by: Andrew Lunn <andrew@lunn.ch>
    Link: https://lore.kernel.org/r/20230530145223.1223993-1-andreas.svensson@axis.com
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 53c6c09aa2c8c430893e3e145a2d69131d6501c5
Author: Bert Karwatzki <spasswolf@web.de>
Date:   Wed May 31 12:36:19 2023 +0200

    net: ipa: Use correct value for IPA_STATUS_SIZE

    [ Upstream commit be7f8012a513f5099916ee2da28420156cbb8cf3 ]

    IPA_STATUS_SIZE was introduced in commit b8dc7d0eea5a as a replacement
    for the size of the removed struct ipa_status which had size
    sizeof(__le32[8]). Use this value as IPA_STATUS_SIZE.

    Fixes: b8dc7d0eea5a ("net: ipa: stop using sizeof(status)")
    Signed-off-by: Bert Karwatzki <spasswolf@web.de>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Link: https://lore.kernel.org/r/20230531103618.102608-1-spasswolf@web.de
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
```

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

commit b666755aabb34506feaf9d145c5b090a6b3e4e27
Author: fuyuanli <fuyuanli@didiglobal.com>
Date:   Wed May 31 16:01:50 2023 +0800

    tcp: fix mishandling when the sack compression is deferred.

    [ Upstream commit 30c6f0bf9579debce27e45fac34fdc97e46acacc ]

    In this patch, we mainly try to handle sending a compressed ack
    correctly if it's deferred.

    Here are more details in the old logic:
    When sack compression is triggered in the tcp_compressed_ack_kick(),
    if the sock is owned by user, it will set TCP_DELACK_TIMER_DEFERRED
    and then defer to the release cb phrase. Later once user releases
    the sock, tcp_delack_timer_handler() should send a ack as expected,
    which, however, cannot happen due to lack of ICSK_ACK_TIMER flag.
    Therefore, the receiver would not sent an ack until the sender's
    retransmission timeout. It definitely increases unnecessary latency.

    Fixes: 5d9f4262b7ea ("tcp: add SACK compression")
    Suggested-by: Eric Dumazet <edumazet@google.com>
    Signed-off-by: fuyuanli <fuyuanli@didiglobal.com>
    Signed-off-by: Jason Xing <kerneljasonxing@gmail.com>
    Link: https://lore.kernel.org/netdev/20230529113804.GA20300@didi-ThinkCentre-M920t-N000/
    Reviewed-by: Eric Dumazet <edumazet@google.com>
    Link: https://lore.kernel.org/r/20230531080150.GA20424@didi-ThinkCentre-M920t-N000
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 900fab73a9cd3dd6a3a69f89980f8f3c9a738d5a
Author: Hangyu Hua <hbh25y@gmail.com>
Date:   Wed May 31 18:28:04 2023 +0800

    net/sched: flower: fix possible OOB write in fl_set_geneve_opt()

    [ Upstream commit 4d56304e5827c8cc8cc18c75343d283af7c4825c ]

    If we send two TCA_FLOWER_KEY_ENC_OPTS_GENEVE packets and their total
    size is 252 bytes(key->enc_opts.len = 252) then
    key->enc_opts.len = opt->length = data_len / 4 = 0 when the third
    TCA_FLOWER_KEY_ENC_OPTS_GENEVE packet enters fl_set_geneve_opt. This
    bypasses the next bounds check and results in an out-of-bounds.

    Fixes: 0a6e77784f49 ("net/sched: allow flower to match tunnel options")
    Signed-off-by: Hangyu Hua <hbh25y@gmail.com>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Reviewed-by: Pieter Jansen van Vuuren <pieter.jansen-van-vuuren@amd.com>
    Link: https://lore.kernel.org/r/20230531102805.27090-1-hbh25y@gmail.com
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 57d5b9ee1874c864218fe92c1fdde6a4ba242b32
Author: Chen-Yu Tsai <wenst@chromium.org>
Date:   Fri May 26 16:53:59 2023 +0800

    iommu/mediatek: Flush IOTLB completely only if domain has been attached

    [ Upstream commit b3fc95709c54ffbe80f16801e0a792a4d2b3d55e ]

    If an IOMMU domain was never attached, it lacks any linkage to the
    actual IOMMU hardware. Attempting to do flush_iotlb_all() on it will
    result in a NULL pointer dereference. This seems to happen after the
```

```
    recent IOMMU core rework in v6.4-rc1.

        Unable to handle kernel read from unreadable memory at virtual address
0000000000000018
        Call trace:
         mtk_iommu_flush_iotlb_all+0x20/0x80
         iommu_create_device_direct_mappings.part.0+0x13c/0x230
         iommu_setup_default_domain+0x29c/0x4d0
         iommu_probe_device+0x12c/0x190
         of_iommu_configure+0x140/0x208
         of_dma_configure_id+0x19c/0x3c0
         platform_dma_configure+0x38/0x88
         really_probe+0x78/0x2c0

    Check if the "bank" field has been filled in before actually attempting
    the IOTLB flush to avoid it. The IOTLB is also flushed when the device
    comes out of runtime suspend, so it should have a clean initial state.

    Fixes: 08500c43d4f7 ("iommu/mediatek: Adjust the structure")
    Signed-off-by: Chen-Yu Tsai <wenst@chromium.org>
    Reviewed-by: Yong Wu <yong.wu@mediatek.com>
    Reviewed-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Link: https://lore.kernel.org/r/20230526085402.394239-1-wenst@chromium.org
    Signed-off-by: Joerg Roedel <jroedel@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ae3b39de89edac973e32955088dd7ac162a21941
Author: Edward Cree <ecree.xilinx@gmail.com>
Date:   Tue May 30 21:25:27 2023 +0100

    sfc: fix error unwinds in TC offload

    [ Upstream commit 622ab656344a288acf4fb03d628c3bb5dd241f34 ]

    Failure ladders weren't exactly unwinding what the function had done up
     to that point; most seriously, when we encountered an already offloaded
     rule, the failure path tried to remove the new rule from the hashtable,
     which would in fact remove the already-present 'old' rule (since it has
     the same key) from the table, and leak its resources.

    Reported-by: kernel test robot <lkp@intel.com>
    Reported-by: Dan Carpenter <error27@gmail.com>
    Closes: https://lore.kernel.org/r/202305200745.xmIlkqjH-lkp@intel.com/
    Fixes: d902e1a737d4 ("sfc: bare bones TC offload on EF100")
    Fixes: 17654d84b47c ("sfc: add offloading of 'foreign' TC (decap) rules")
    Signed-off-by: Edward Cree <ecree.xilinx@gmail.com>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Link: https://lore.kernel.org/r/20230530202527.53115-1-edward.cree@amd.com
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ccf553af9e25ee0e17c463855188c961a5c58057
Author: Moshe Shemesh <moshe@nvidia.com>
Date:   Fri Apr 28 13:48:13 2023 +0300

    net/mlx5: Read embedded cpu after init bit cleared

    [ Upstream commit bbfa4b58997e3d38ba629c9f6fc0bd1c163aaf43 ]

    During driver load it reads embedded_cpu bit from initialization
    segment, but the initialization segment is readable only after
    initialization bit is cleared.

    Move the call to mlx5_read_embedded_cpu() right after initialization bit
    cleared.
```

```
    Signed-off-by: Moshe Shemesh <moshe@nvidia.com>
    Fixes: 591905ba9679 ("net/mlx5: Introduce Mellanox SmartNIC and modify page management
logic")
    Reviewed-by: Shay Drory <shayd@nvidia.com>
    Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 042d64776dc79d0db308111cad8d5ed7de82b041
Author: Saeed Mahameed <saeedm@nvidia.com>
Date:   Sat May 27 23:07:08 2023 -0700

    net/mlx5e: Fix error handling in mlx5e_refresh_tirs

    [ Upstream commit b6193d7030e3c59f1d4c75648c9c8fa40cad2bcd ]

    Allocation failure is outside the critical lock section and should
    return immediately rather than jumping to the unlock section.

    Also unlock as soon as required and remove the now redundant jump label.

    Fixes: 80a2a9026b24 ("net/mlx5e: Add a lock on tir list")
    Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4cbc0a0a0338601cf6d13f18f8f00986e3216a26
Author: Christoph Hellwig <hch@lst.de>
Date:   Wed May 31 14:54:54 2023 +0200

    nvme: fix the name of Zone Append for verbose logging

    [ Upstream commit 856303797724d28f1d65b702f0eadcee1ea7abf5 ]

    No Management involved in Zone Appened.

    Fixes: bd83fe6f2cd2 ("nvme: add verbose error logging")
    Signed-off-by: Christoph Hellwig <hch@lst.de>
    Reviewed-by: Alan Adamson <alan.adamson@oracle.com>
    Signed-off-by: Keith Busch <kbusch@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7eafcd685fd6ecae627b68e6d1eee695b3f61618
Author: Bastien Nocera <hadess@hadess.net>
Date:   Wed May 31 10:24:28 2023 +0200

    HID: logitech-hidpp: Handle timeout differently from busy

    [ Upstream commit 6199d23c91ce53bfed455f09a8c5ed170d516824 ]

    If an attempt at contacting a receiver or a device fails because the
    receiver or device never responds, don't restart the communication, only
    restart it if the receiver or device answers that it's busy, as originally
    intended.

    This was the behaviour on communication timeout before commit 586e8fede795
    ("HID: logitech-hidpp: Retry commands when device is busy").

    This fixes some overly long waits in a critical path on boot, when
    checking whether the device is connected by getting its HID++ version.

    Signed-off-by: Bastien Nocera <hadess@hadess.net>
    Suggested-by: Mark Lord <mlord@pobox.com>
    Fixes: 586e8fede795 ("HID: logitech-hidpp: Retry commands when device is busy")
    Link: https://bugzilla.kernel.org/show_bug.cgi?id=217412
    Signed-off-by: Jiri Kosina <jkosina@suse.cz>
```

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

commit 634fae934ebb6bbf9e2291aa883a18c19d5f4d1d
Author: Dan Carpenter <dan.carpenter@linaro.org>
Date:   Mon May 29 14:35:55 2023 +0300

    nfsd: fix double fget() bug in __write_ports_addfd()

    [ Upstream commit c034203b6a9dae6751ef4371c18cb77983e30c28 ]

    The bug here is that you cannot rely on getting the same socket
    from multiple calls to fget() because userspace can influence
    that.  This is a kind of double fetch bug.

    The fix is to delete the svc_alien_sock() function and instead do
    the checking inside the svc_addsock() function.

    Fixes: 3064639423c4 ("nfsd: check passed socket's net matches NFSd superblock's one")
    Signed-off-by: Dan Carpenter <dan.carpenter@linaro.org>
    Reviewed-by: NeilBrown <neilb@suse.com>
    Reviewed-by: Jeff Layton <jlayton@kernel.org>
    Signed-off-by: Chuck Lever <chuck.lever@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit abaf030b6d62de3936cd6b7bd83dceec618a428a
Author: Vladislav Efanov <VEfanov@ispras.ru>
Date:   Tue May 30 14:39:41 2023 +0300

    udp6: Fix race condition in udp6_sendmsg & connect

    [ Upstream commit 448a5ce1120c5bdbce1f1ccdabcd31c7d029f328 ]

    Syzkaller got the following report:
    BUG: KASAN: use-after-free in sk_setup_caps+0x621/0x690 net/core/sock.c:2018
    Read of size 8 at addr ffff888027f82780 by task syz-executor276/3255

    The function sk_setup_caps (called by ip6_sk_dst_store_flow->
    ip6_dst_store) referenced already freed memory as this memory was
    freed by parallel task in udpv6_sendmsg->ip6_sk_dst_lookup_flow->
    sk_dst_check.

              task1 (connect)              task2 (udp6_sendmsg)
           sk_setup_caps->sk_dst_set |
                                     |  sk_dst_check->
                                     |      sk_dst_set
                                     |      dst_release
           sk_setup_caps references  |
           to already freed dst_entry|

    The reason for this race condition is: sk_setup_caps() keeps using
    the dst after transferring the ownership to the dst cache.

    Found by Linux Verification Center (linuxtesting.org) with syzkaller.

    Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
    Signed-off-by: Vladislav Efanov <VEfanov@ispras.ru>
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9ffd3d30e342ab911af0572baef13b463633dfa9
Author: Pedro Tammela <pctammela@mojatatu.com>
Date:   Mon May 29 12:33:35 2023 -0300

    net/netlink: fix NETLINK_LIST_MEMBERSHIPS length report
```

```
    [ Upstream commit f4e4534850a9d18c250a93f8d7fbb51310828110 ]

    The current code for the length calculation wrongly truncates the reported
    length of the groups array, causing an under report of the subscribed
    groups. To fix this, use 'BITS_TO_BYTES()' which rounds up the
    division by 8.

    Fixes: b42be38b2778 ("netlink: add API to retrieve all group memberships")
    Signed-off-by: Pedro Tammela <pctammela@mojatatu.com>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Link: https://lore.kernel.org/r/20230529153335.389815-1-pctammela@mojatatu.com
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 309f2241fbb5a60ead5a2b51e443508cf7a093a0
Author: Zhengchao Shao <shaozhengchao@huawei.com>
Date:   Sat May 27 17:37:47 2023 +0800

    net: sched: fix NULL pointer dereference in mq_attach

    [ Upstream commit 36eec020fab668719b541f34d97f44e232ffa165 ]

    When use the following command to test:
    1)ip link add bond0 type bond
    2)ip link set bond0 up
    3)tc qdisc add dev bond0 root handle ffff: mq
    4)tc qdisc replace dev bond0 parent ffff:fff1 handle ffff: mq

    The kernel reports NULL pointer dereference issue. The stack information
    is as follows:
    Unable to handle kernel NULL pointer dereference at virtual address 0000000000000000
    Internal error: Oops: 0000000096000006 [#1] SMP
    Modules linked in:
    pstate: 20000005 (nzCv daif -PAN -UAO -TCO -DIT -SSBS BTYPE=--)
    pc : mq_attach+0x44/0xa0
    lr : qdisc_graft+0x20c/0x5cc
    sp : ffff80000e2236a0
    x29: ffff80000e2236a0 x28: ffff0000c0e59d80 x27: ffff0000c0be19c0
    x26: ffff0000cae3e800 x25: 0000000000000010 x24: 00000000fffffff1
    x23: 0000000000000000 x22: ffff0000cae3e800 x21: ffff0000c9df4000
    x20: ffff0000c9df4000 x19: 0000000000000000 x18: ffff80000a934000
    x17: ffff8000f5b56000 x16: ffff80000bb08000 x15: 0000000000000000
    x14: 0000000000000000 x13: 6b6b6b6b6b6b6b6b x12: 6b6b6b6b00000001
    x11: 0000000000000000 x10: 0000000000000000 x9 : 0000000000000000
    x8 : ffff0000c0be0730 x7 : bbbbbbbbbbbbbbbb x6 : 0000000000000008
    x5 : ffff0000cae3e864 x4 : 0000000000000000 x3 : 0000000000000001
    x2 : 0000000000000001 x1 : ffff8000090bc23c x0 : 0000000000000000
    Call trace:
    mq_attach+0x44/0xa0
    qdisc_graft+0x20c/0x5cc
    tc_modify_qdisc+0x1c4/0x664
    rtnetlink_rcv_msg+0x354/0x440
    netlink_rcv_skb+0x64/0x144
    rtnetlink_rcv+0x28/0x34
    netlink_unicast+0x1e8/0x2a4
    netlink_sendmsg+0x308/0x4a0
    sock_sendmsg+0x64/0xac
    ____sys_sendmsg+0x29c/0x358
    ___sys_sendmsg+0x90/0xd0
    __sys_sendmsg+0x7c/0xd0
    __arm64_sys_sendmsg+0x2c/0x38
    invoke_syscall+0x54/0x114
    el0_svc_common.constprop.1+0x90/0x174
    do_el0_svc+0x3c/0xb0
```

```
      el0_svc+0x24/0xec
      el0t_64_sync_handler+0x90/0xb4
      el0t_64_sync+0x174/0x178

      This is because when mq is added for the first time, qdiscs in mq is set
      to NULL in mq_attach(). Therefore, when replacing mq after adding mq, we
      need to initialize qdiscs in the mq before continuing to graft. Otherwise,
      it will couse NULL pointer dereference issue in mq_attach(). And the same
      issue will occur in the attach functions of mqprio, taprio and htb.
      ffff:fff1 means that the repalce qdisc is ingress. Ingress does not allow
      any qdisc to be attached. Therefore, ffff:fff1 is incorrectly used, and
      the command should be dropped.

      Fixes: 6ec1c69a8f64 ("net_sched: add classful multiqueue dummy scheduler")
      Signed-off-by: Zhengchao Shao <shaozhengchao@huawei.com>
      Tested-by: Peilin Ye <peilin.ye@bytedance.com>
      Acked-by: Jamal Hadi Salim <jhs@mojatatu.com>
      Link: https://lore.kernel.org/r/20230527093747.3583502-1-shaozhengchao@huawei.com
      Signed-off-by: Jakub Kicinski <kuba@kernel.org>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 680e7a88f0b4015f25cd3c09748e8041204e7355
  Author: Peilin Ye <peilin.ye@bytedance.com>
  Date:   Mon May 29 12:54:26 2023 -0700

      net/sched: Prohibit regrafting ingress or clsact Qdiscs

      [ Upstream commit 9de95df5d15baa956c2b70b9e794842e790a8a13 ]

      Currently, after creating an ingress (or clsact) Qdisc and grafting it
      under TC_H_INGRESS (TC_H_CLSACT), it is possible to graft it again under
      e.g. a TBF Qdisc:

        $ ip link add ifb0 type ifb
        $ tc qdisc add dev ifb0 handle 1: root tbf rate 20kbit buffer 1600 limit 3000
        $ tc qdisc add dev ifb0 clsact
        $ tc qdisc link dev ifb0 handle ffff: parent 1:1
        $ tc qdisc show dev ifb0
        qdisc tbf 1: root refcnt 2 rate 20Kbit burst 1600b lat 560.0ms
        qdisc clsact ffff: parent ffff:fff1 refcnt 2
                                            ^^^^^^^^


      clsact's refcount has increased: it is now grafted under both
      TC_H_CLSACT and 1:1.

      ingress and clsact Qdiscs should only be used under TC_H_INGRESS
      (TC_H_CLSACT).  Prohibit regrafting them.

      Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
      Fixes: 1f211a1b929c ("net, sched: add clsact qdisc")
      Tested-by: Pedro Tammela <pctammela@mojatatu.com>
      Acked-by: Jamal Hadi Salim <jhs@mojatatu.com>
      Reviewed-by: Jamal Hadi Salim <jhs@mojatatu.com>
      Reviewed-by: Vlad Buslov <vladbu@nvidia.com>
      Signed-off-by: Peilin Ye <peilin.ye@bytedance.com>
      Signed-off-by: Jakub Kicinski <kuba@kernel.org>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 6087625ad3efc3306ff643988a1b1c9c3920cf53
  Author: Peilin Ye <peilin.ye@bytedance.com>
  Date:   Mon May 29 12:54:03 2023 -0700

      net/sched: Reserve TC_H_INGRESS (TC_H_CLSACT) for ingress (clsact) Qdiscs

      [ Upstream commit f85fa45d4a9408d98c46c8fa45ba2e3b2f4bf219 ]
```

```
    Currently it is possible to add e.g. an HTB Qdisc under ffff:fff1
    (TC_H_INGRESS, TC_H_CLSACT):

      $ ip link add name ifb0 type ifb
      $ tc qdisc add dev ifb0 parent ffff:fff1 htb
      $ tc qdisc add dev ifb0 clsact
      Error: Exclusivity flag on, cannot modify.
      $ drgn
      ...
      >>> ifb0 = netdev_get_by_name(prog, "ifb0")
      >>> qdisc = ifb0.ingress_queue.qdisc_sleeping
      >>> print(qdisc.ops.id.string_().decode())
      htb
      >>> qdisc.flags.value_() # TCQ_F_INGRESS
      2

    Only allow ingress and clsact Qdiscs under ffff:fff1.  Return -EINVAL
    for everything else.  Make TCQ_F_INGRESS a static flag of ingress and
    clsact Qdiscs.

    Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
    Fixes: 1f211a1b929c ("net, sched: add clsact qdisc")
    Tested-by: Pedro Tammela <pctammela@mojatatu.com>
    Acked-by: Jamal Hadi Salim <jhs@mojatatu.com>
    Reviewed-by: Jamal Hadi Salim <jhs@mojatatu.com>
    Reviewed-by: Vlad Buslov <vladbu@nvidia.com>
    Signed-off-by: Peilin Ye <peilin.ye@bytedance.com>
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 952649451545d1711f1b2a23f5335a776c5ee633
Author: Peilin Ye <peilin.ye@bytedance.com>
Date:   Mon May 29 12:53:21 2023 -0700

    net/sched: sch_clsact: Only create under TC_H_CLSACT

    [ Upstream commit 5eeebfe6c493192b10d516abfd72742900f2a162 ]

    clsact Qdiscs are only supposed to be created under TC_H_CLSACT (which
    equals TC_H_INGRESS).  Return -EOPNOTSUPP if 'parent' is not
    TC_H_CLSACT.

    Fixes: 1f211a1b929c ("net, sched: add clsact qdisc")
    Tested-by: Pedro Tammela <pctammela@mojatatu.com>
    Acked-by: Jamal Hadi Salim <jhs@mojatatu.com>
    Reviewed-by: Jamal Hadi Salim <jhs@mojatatu.com>
    Reviewed-by: Vlad Buslov <vladbu@nvidia.com>
    Signed-off-by: Peilin Ye <peilin.ye@bytedance.com>
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 79f44ff3a75c30d10cd62cebf4c82c919bb55374
Author: Peilin Ye <peilin.ye@bytedance.com>
Date:   Mon May 29 12:52:55 2023 -0700

    net/sched: sch_ingress: Only create under TC_H_INGRESS

    [ Upstream commit c7cfbd115001f94de9e4053657946a383147e803 ]

    ingress Qdiscs are only supposed to be created under TC_H_INGRESS.
    Return -EOPNOTSUPP if 'parent' is not TC_H_INGRESS, similar to
    mq_init().

    Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
```

```
    Reported-by: syzbot+b53a9c0d1ea4ad62da8b@syzkaller.appspotmail.com
    Closes: https://lore.kernel.org/r/0000000000006cf87705f79acf1a@google.com/
    Tested-by: Pedro Tammela <pctammela@mojatatu.com>
    Acked-by: Jamal Hadi Salim <jhs@mojatatu.com>
    Reviewed-by: Jamal Hadi Salim <jhs@mojatatu.com>
    Reviewed-by: Vlad Buslov <vladbu@nvidia.com>
    Signed-off-by: Peilin Ye <peilin.ye@bytedance.com>
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 134246b7acb4b2721936eb7ef5e0649944772548
Author: Wen Gu <guwen@linux.alibaba.com>
Date:   Fri May 26 19:49:01 2023 +0800

    net/smc: Don't use RMBs not mapped to new link in SMCRv2 ADD LINK

    [ Upstream commit 71c6aa0305e3d2365d3bfd0134b4025d9e7ba388 ]

    We encountered a crash when using SMCRv2. It is caused by a logical
    error in smc_llc_fill_ext_v2().

     BUG: kernel NULL pointer dereference, address: 0000000000000014
     #PF: supervisor read access in kernel mode
     #PF: error_code(0x0000) - not-present page
     PGD 0 P4D 0
     Oops: 0000 [#1] PREEMPT SMP PTI
     CPU: 7 PID: 453 Comm: kworker/7:4 Kdump: loaded Tainted: G        W   E      6.4.0-rc3+
#44
     Workqueue: events smc_llc_add_link_work [smc]
     RIP: 0010:smc_llc_fill_ext_v2+0x117/0x280 [smc]
     RSP: 0018:ffffacb5c064bd88 EFLAGS: 00010282
     RAX: ffff9a6bc1c3c02c RBX: ffff9a6be3558000 RCX: 0000000000000000
     RDX: 0000000000000002 RSI: 0000000000000002 RDI: 000000000000000a
     RBP: ffffacb5c064bdb8 R08: 0000000000000040 R09: 000000000000000c
     R10: ffff9a6bc0910300 R11: 0000000000000002 R12: 0000000000000000
     R13: 0000000000000002 R14: ffff9a6bc1c3c02c R15: ffff9a6be3558250
     FS:  0000000000000000(0000) GS:ffff9a6eefdc0000(0000) knlGS:0000000000000000
     CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
     CR2: 0000000000000014 CR3: 000000010b078003 CR4: 00000000003706e0
     DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000
     DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400
     Call Trace:
      <TASK>
      smc_llc_send_add_link+0x1ae/0x2f0 [smc]
      smc_llc_srv_add_link+0x2c9/0x5a0 [smc]
      ? cc_mkenc+0x40/0x60
      smc_llc_add_link_work+0xb8/0x140 [smc]
      process_one_work+0x1e5/0x3f0
      worker_thread+0x4d/0x2f0
      ? __pfx_worker_thread+0x10/0x10
      kthread+0xe5/0x120
      ? __pfx_kthread+0x10/0x10
      ret_from_fork+0x2c/0x50
      </TASK>

    When an alernate RNIC is available in system, SMC will try to add a new
    link based on the RNIC for resilience. All the RMBs in use will be mapped
    to the new link. Then the RMBs' MRs corresponding to the new link will be
    filled into SMCRv2 LLC ADD LINK messages.

    However, smc_llc_fill_ext_v2() mistakenly accesses to unused RMBs which
    haven't been mapped to the new link and have no valid MRs, thus causing
    a crash. So this patch fixes the logic.

    Fixes: b4ba4652b3f8 ("net/smc: extend LLC layer for SMC-Rv2")
```

```
    Signed-off-by: Wen Gu <guwen@linux.alibaba.com>
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit bf1eba8ccc8ea95f55c8ed7ac6221c131c09a5a1
Author: Wen Gu <guwen@linux.alibaba.com>
Date:   Fri May 26 19:49:00 2023 +0800

    net/smc: Scan from current RMB list when no position specified

    [ Upstream commit b24aa141c2ff26c919237aee61ea1818fc6780d9 ]

    When finding the first RMB of link group, it should start from the
    current RMB list whose index is 0. So fix it.

    Fixes: b4ba4652b3f8 ("net/smc: extend LLC layer for SMC-Rv2")
    Signed-off-by: Wen Gu <guwen@linux.alibaba.com>
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5e27b692c4b6d85673295d6a6d2e82c7392edc2c
Author: David Howells <dhowells@redhat.com>
Date:   Fri May 26 12:34:54 2023 +0100

    rxrpc: Truncate UTS_RELEASE for rxrpc version

    [ Upstream commit 020c69c1a793ed29d28793808eddd75210c858dd ]

    UTS_RELEASE has a maximum length of 64 which can cause rxrpc_version to
    exceed the 65 byte message limit.

    Per the rx spec[1]: "If a server receives a packet with a type value of 13,
    and the client-initiated flag set, it should respond with a 65-byte payload
    containing a string that identifies the version of AFS software it is
    running."

    The current implementation causes a compile error when WERROR is turned on
    and/or UTS_RELEASE exceeds the length of 49 (making the version string more
    than 64 characters).

    Fix this by generating the string during module initialisation and limiting
    the UTS_RELEASE segment of the string does not exceed 49 chars.  We need to
    make sure that the 64 bytes includes "linux-" at the front and " AF_RXRPC"
    at the back as this may be used in pattern matching.

    Fixes: 44ba06987c0b ("RxRPC: Handle VERSION Rx protocol packets")
    Reported-by: Kenny Ho <Kenny.Ho@amd.com>
    Link: https://lore.kernel.org/r/20230523223944.691076-1-Kenny.Ho@amd.com/
    Signed-off-by: David Howells <dhowells@redhat.com>
    Acked-by: Kenny Ho <Kenny.Ho@amd.com>
    cc: Marc Dionne <marc.dionne@auristor.com>
    cc: Andrew Lunn <andrew@lunn.ch>
    cc: David Laight <David.Laight@ACULAB.COM>
    cc: "David S. Miller" <davem@davemloft.net>
    cc: Eric Dumazet <edumazet@google.com>
    cc: Jakub Kicinski <kuba@kernel.org>
    cc: Paolo Abeni <pabeni@redhat.com>
    cc: linux-afs@lists.infradead.org
    cc: netdev@vger.kernel.org
    Link: https://web.mit.edu/kolya/afs/rx/rx-spec [1]
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Reviewed-by: Jeffrey Altman <jaltman@auristor.com>
    Link: https://lore.kernel.org/r/654974.1685100894@warthog.procyon.org.uk
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit df04b09fce636a3e36834688ae8fdb36fe97770b
Author: Cambda Zhu <cambda@linux.alibaba.com>
Date:   Sat May 27 12:03:17 2023 +0800

    tcp: Return user_mss for TCP_MAXSEG in CLOSE/LISTEN state if user_mss set

    [ Upstream commit 34dfde4ad87b84d21278a7e19d92b5b2c68e6c4d ]

    This patch replaces the tp->mss_cache check in getting TCP_MAXSEG
    with tp->rx_opt.user_mss check for CLOSE/LISTEN sock. Since
    tp->mss_cache is initialized with TCP_MSS_DEFAULT, checking if
    it's zero is probably a bug.

    With this change, getting TCP_MAXSEG before connecting will return
    default MSS normally, and return user_mss if user_mss is set.

    Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
    Reported-by: Jack Yang <mingliang@linux.alibaba.com>
    Suggested-by: Eric Dumazet <edumazet@google.com>
    Link:
https://lore.kernel.org/netdev/CANn89i+3kL9pYtkxkwxwNMzvC_w3LNUum_2=3u+UyLBmGmifHA@mail.gmai
l.com/#t
    Signed-off-by: Cambda Zhu <cambda@linux.alibaba.com>
    Link: https://lore.kernel.org/netdev/14D45862-36EA-4076-974C-
EA67513C92F6@linux.alibaba.com/
    Reviewed-by: Jason Xing <kerneljasonxing@gmail.com>
    Reviewed-by: Eric Dumazet <edumazet@google.com>
    Link: https://lore.kernel.org/r/20230527040317.68247-1-cambda@linux.alibaba.com
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 02364cd5e6e2db451bda084ea9f61d7e041cca14
Author: Eric Dumazet <edumazet@google.com>
Date:   Fri May 26 16:34:58 2023 +0000

    tcp: deny tcp_disconnect() when threads are waiting

    [ Upstream commit 4faeee0cf8a5d88d63cdbc3bab124fb0e6aed08c ]

    Historically connect(AF_UNSPEC) has been abused by syzkaller
    and other fuzzers to trigger various bugs.

    A recent one triggers a divide-by-zero [1], and Paolo Abeni
    was able to diagnose the issue.

    tcp_recvmsg_locked() has tests about sk_state being not TCP_LISTEN
    and TCP REPAIR mode being not used.

    Then later if socket lock is released in sk_wait_data(),
    another thread can call connect(AF_UNSPEC), then make this
    socket a TCP listener.

    When recvmsg() is resumed, it can eventually call tcp_cleanup_rbuf()
    and attempt a divide by 0 in tcp_rcv_space_adjust() [1]

    This patch adds a new socket field, counting number of threads
    blocked in sk_wait_event() and inet_wait_for_connect().

    If this counter is not zero, tcp_disconnect() returns an error.

    This patch adds code in blocking socket system calls, thus should
    not hurt performance of non blocking ones.

    Note that we probably could revert commit 499350a5a6e7 ("tcp:
```

```
    initialize rcv_mss to TCP_MIN_MSS instead of 0") to restore
    original tcpi_rcv_mss meaning (was 0 if no payload was ever
    received on a socket)

    [1]
    divide error: 0000 [#1] PREEMPT SMP KASAN
    CPU: 0 PID: 13832 Comm: syz-executor.5 Not tainted 6.3.0-rc4-syzkaller-00224-
g00c7b5f4ddc5 #0
    Hardware name: Google Google Compute Engine/Google Compute Engine, BIOS Google
03/02/2023
    RIP: 0010:tcp_rcv_space_adjust+0x36e/0x9d0 net/ipv4/tcp_input.c:740
    Code: 00 00 00 00 fc ff df 4c 89 64 24 48 8b 44 24 04 44 89 f9 41 81 c7 80 03 00 00 c1
e1 04 44 29 f0 48 63 c9 48 01 e9 48 0f af c1 <49> f7 f6 48 8d 04 41 48 89 44 24 40 48 8b 44
24 30 48 c1 e8 03 48
    RSP: 0018:ffffc900033af660 EFLAGS: 00010206
    RAX: 4a66b76cbade2c48 RBX: ffff888076640cc0 RCX: 00000000c334e4ac
    RDX: 0000000000000000 RSI: dfffffc0000000000 RDI: 0000000000000001
    RBP: 00000000c324e86c R08: 0000000000000001 R09: 0000000000000000
    R10: 0000000000000000 R11: 0000000000000000 R12: ffff8880766417f8
    R13: ffff888028fbb980 R14: 0000000000000000 R15: 0000000000010344
    FS: 00007f5bffbfe700(0000) GS:ffff8880b9800000(0000) knlGS:0000000000000000
    CS: 0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    CR2: 0000001b32f25000 CR3: 000000007ced0000 CR4: 00000000003506f0
    DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000
    DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400
    Call Trace:
    <TASK>
    tcp_recvmsg_locked+0x100e/0x22e0 net/ipv4/tcp.c:2616
    tcp_recvmsg+0x117/0x620 net/ipv4/tcp.c:2681
    inet6_recvmsg+0x114/0x640 net/ipv6/af_inet6.c:670
    sock_recvmsg_nosec net/socket.c:1017 [inline]
    sock_recvmsg+0xe2/0x160 net/socket.c:1038
    ____sys_recvmsg+0x210/0x5a0 net/socket.c:2720
    ___sys_recvmsg+0xf2/0x180 net/socket.c:2762
    do_recvmmsg+0x25e/0x6e0 net/socket.c:2856
    __sys_recvmmsg net/socket.c:2935 [inline]
    __do_sys_recvmmsg net/socket.c:2958 [inline]
    __se_sys_recvmmsg net/socket.c:2951 [inline]
    __x64_sys_recvmmsg+0x20f/0x260 net/socket.c:2951
    do_syscall_x64 arch/x86/entry/common.c:50 [inline]
    do_syscall_64+0x39/0xb0 arch/x86/entry/common.c:80
    entry_SYSCALL_64_after_hwframe+0x63/0xcd
    RIP: 0033:0x7f5c0108c0f9
    Code: 28 00 00 00 75 05 48 83 c4 28 c3 e8 f1 19 00 00 90 48 89 f8 48 89 f7 48 89 d6 48
89 ca 4d 89 c2 4d 89 c8 4c 8b 4c 24 08 0f 05 <48> 3d 01 f0 ff ff 73 01 c3 48 c7 c1 b8 ff ff
ff f7 d8 64 89 01 48
    RSP: 002b:00007f5bffbfe168 EFLAGS: 00000246 ORIG_RAX: 000000000000012b
    RAX: ffffffffffffffda RBX: 00007f5c011ac050 RCX: 00007f5c0108c0f9
    RDX: 0000000000000001 RSI: 0000000020000bc0 RDI: 0000000000000003
    RBP: 00007f5c010e7b39 R08: 0000000000000000 R09: 0000000000000000
    R10: 0000000000000122 R11: 0000000000000246 R12: 0000000000000000
    R13: 00007f5c012cfb1f R14: 00007f5bffbfe300 R15: 0000000000022000
    </TASK>

    Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
    Reported-by: syzbot <syzkaller@googlegroups.com>
    Reported-by: Paolo Abeni <pabeni@redhat.com>
    Diagnosed-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Eric Dumazet <edumazet@google.com>
    Tested-by: Paolo Abeni <pabeni@redhat.com>
    Link: https://lore.kernel.org/r/20230526163458.2880232-1-edumazet@google.com
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c12efc7e123b315b6d818ed4d102d6941cd38ae1
```

```
Author: Eric Dumazet <edumazet@google.com>
Date:   Fri May 26 15:43:42 2023 +0000

    af_packet: do not use READ_ONCE() in packet_bind()

    [ Upstream commit 6ffc57ea004234d9373c57b204fd10370a69f392 ]

    A recent patch added READ_ONCE() in packet_bind() and packet_bind_spkt()

    This is better handled by reading pkt_sk(sk)->num later
    in packet_do_bind() while appropriate lock is held.

    READ_ONCE() in writers are often an evidence of something being wrong.

    Fixes: 822b5a1c17df ("af_packet: Fix data-races of pkt_sk(sk)->num.")
    Signed-off-by: Eric Dumazet <edumazet@google.com>
    Reviewed-by: Willem de Bruijn <willemb@google.com>
    Reviewed-by: Jiri Pirko <jiri@nvidia.com>
    Reviewed-by: Kuniyuki Iwashima <kuniyu@amazon.com>
    Link: https://lore.kernel.org/r/20230526154342.2533026-1-edumazet@google.com
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0d516e61cc96f7ee10975b10d92116d2eda2ec01
Author: Mustafa Ismail <mustafa.ismail@intel.com>
Date:   Mon May 22 10:56:54 2023 -0500

    RDMA/irdma: Fix Local Invalidate fencing

    [ Upstream commit 5842d1d9c1b0d17e0c29eae65ae1f245f83682dd ]

    If the local invalidate fence is indicated in the WR, only the read fence
    is currently being set in WQE. Fix this to set both the read and local
    fence in the WQE.

    Fixes: b48c24c2d710 ("RDMA/irdma: Implement device supported verb APIs")
    Link: https://lore.kernel.org/r/20230522155654.1309-4-shiraz.saleem@intel.com
    Signed-off-by: Mustafa Ismail <mustafa.ismail@intel.com>
    Signed-off-by: Shiraz Saleem <shiraz.saleem@intel.com>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6ee53f82540769a6d6e77e40b901f9b9edfa5ff2
Author: Mustafa Ismail <mustafa.ismail@intel.com>
Date:   Mon May 22 10:56:53 2023 -0500

    RDMA/irdma: Prevent QP use after free

    [ Upstream commit c8f304d75f6c6cc679a73f89591f9a915da38f09 ]

    There is a window where the poll cq may use a QP that has been freed.
    This can happen if a CQE is polled before irdma_clean_cqes() can clear the
    CQE's related to the QP and the destroy QP races to free the QP memory.
    then the QP structures are used in irdma_poll_cq.  Fix this by moving the
    clearing of CQE's before the reference is removed and the QP is destroyed.

    Fixes: b48c24c2d710 ("RDMA/irdma: Implement device supported verb APIs")
    Link: https://lore.kernel.org/r/20230522155654.1309-3-shiraz.saleem@intel.com
    Signed-off-by: Mustafa Ismail <mustafa.ismail@intel.com>
    Signed-off-by: Shiraz Saleem <shiraz.saleem@intel.com>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f1c50a8f2782824085373af69a0822b4d3ca0a87
Author: Akihiro Suda <suda.kyoto@gmail.com>
```

```
Date:   Sun May 28 19:36:02 2023 +0200

    efi: Bump stub image version for macOS HVF compatibility

    [ Upstream commit 36e4fc57fc1619f462e669e939209c45763bc8f5 ]

    The macOS hypervisor framework includes a host-side VMM called
    VZLinuxBootLoader [1] which implements native support for booting the
    Linux kernel inside a guest directly (instead of, e.g., via GRUB
    installed inside the guest). On x86, it incorporates a BIOS style loader
    that does not implement or expose EFI to the loaded kernel. However,
    this loader appears to fail when the 'image minor version' field in the
    kernel image's PE/COFF header (which is generally only used by EFI based
    bootloaders) is set to any value other than 0x0. [2]

    Commit e346bebbd36b1576 ("efi: libstub: Always enable initrd command
    line loader and bump version") incremented the EFI stub image minor
    version to convey that all EFI stub kernels now implement support for
    the initrd= command line option, and do so in a way where it can load
    initrd images from any filesystem known to the EFI firmware (as opposed
    to prior implementations that could only load initrds from the same
    volume that the kernel image was loaded from).

    Unfortunately, bumping the version to v1.1 triggers this issue in
    VZLinuxBootLoader, breaking the boot on x86. So let's keep the image
    minor version at 0x0, and bump the image major version instead.

    While at it, convert this field to a bit field, so that individual
    features are discoverable from it, as suggested by Linus. So let's bump
    the major version to v3, and document the initrd= command line loading
    feature as being represented by bit 1 in the mask.

    Note that, due to the prior interpretation as a monotonically increasing
    version field, loaders are still permitted to assume that the LoadFile2
    initrd loading feature is supported for any major version value >= 1,
    even if bit 0 is not set.

    [1] https://developer.apple.com/documentation/virtualization/vzlinuxbootloader
    [2] https://lore.kernel.org/linux-
efi/CAG8fp8Teu4G9JuenQrqGndFt2Gy+V4YgJ=hN1xX7AD940YKf3A@mail.gmail.com/

    Fixes: e346bebbd36b1576 ("efi: libstub: Always enable initrd command ...")
    Closes: https://bugzilla.kernel.org/show_bug.cgi?id=217485
    Signed-off-by: Akihiro Suda <suda.kyoto@gmail.com>
    [ardb: rewrite comment and commit log]
    Signed-off-by: Ard Biesheuvel <ardb@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 021bce6d0b0b47940e6a1e7860ff5dbb351d40b0
Author: Arnd Bergmann <arnd@arndb.de>
Date:   Tue May 16 22:21:24 2023 +0200

    mtd: rawnand: ingenic: fix empty stub helper definitions

    [ Upstream commit 650a8884a364ff2568b51cde9009cfd43cdae6ad ]

    A few functions provide an empty interface definition when
    CONFIG_MTD_NAND_INGENIC_ECC is disabled, but they are accidentally
    defined as global functions in the header:

    drivers/mtd/nand/raw/ingenic/ingenic_ecc.h:39:5: error: no previous prototype for
'ingenic_ecc_calculate'
    drivers/mtd/nand/raw/ingenic/ingenic_ecc.h:46:5: error: no previous prototype for
'ingenic_ecc_correct'
    drivers/mtd/nand/raw/ingenic/ingenic_ecc.h:53:6: error: no previous prototype for
```

```
  'ingenic_ecc_release'
      drivers/mtd/nand/raw/ingenic/ingenic_ecc.h:57:21: error: no previous prototype for
  'of_ingenic_ecc_get'

      Turn them into 'static inline' definitions instead.

      Fixes: 15de8c6efd0e ("mtd: rawnand: ingenic: Separate top-level and SoC specific code")
      Signed-off-by: Arnd Bergmann <arnd@arndb.de>
      Reviewed-by: Paul Cercueil <paul@crapouillou.net>
      Signed-off-by: Miquel Raynal <miquel.raynal@bootlin.com>
      Link: https://lore.kernel.org/linux-mtd/20230516202133.559488-1-arnd@kernel.org
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit d24c6685afbd9a0f399503a137e355c3e7478d83
  Author: Namhyung Kim <namhyung@kernel.org>
  Date:   Thu May 25 14:20:38 2023 -0700

      perf ftrace latency: Remove unnecessary "--" from --use-nsec option

      [ Upstream commit 8d73259ef23f449329294dc187932f7470268126 ]

      The option name should not have the dashes.  Current version shows four
      dashes for the option.

        $ perf ftrace latency -h

         Usage: perf ftrace [<options>] [<command>]
            or: perf ftrace [<options>] -- [<command>] [<options>]
            or: perf ftrace {trace|latency} [<options>] [<command>]
            or: perf ftrace {trace|latency} [<options>] -- [<command>] [<options>]

            -b, --use-bpf         Use BPF to measure function latency
            -n, ----use-nsec      Use nano-second histogram
            -T, --trace-funcs <func>
                                  Show latency of given function

      Fixes: 84005bb6148618cc ("perf ftrace latency: Add -n/--use-nsec option")
      Signed-off-by: Namhyung Kim <namhyung@kernel.org>
      Tested-by: Arnaldo Carvalho de Melo <acme@redhat.com>
      Cc: Adrian Hunter <adrian.hunter@intel.com>
      Cc: Changbin Du <changbin.du@huawei.com>
      Cc: Ian Rogers <irogers@google.com>
      Cc: Ingo Molnar <mingo@kernel.org>
      Cc: Jiri Olsa <jolsa@kernel.org>
      Cc: Peter Zijlstra <peterz@infradead.org>
      Link: https://lore.kernel.org/r/20230525212038.3535851-1-namhyung@kernel.org
      Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 071df54c7bfa63c031cc9b2dee79a813274650a4
  Author: Raju Rangoju <Raju.Rangoju@amd.com>
  Date:   Thu May 25 23:56:12 2023 +0530

      amd-xgbe: fix the false linkup in xgbe_phy_status

      [ Upstream commit dc362e20cd6ab7a93d1b09669730c406f0910c35 ]

      In the event of a change in XGBE mode, the current auto-negotiation
      needs to be reset and the AN cycle needs to be re-triggerred. However,
      the current code ignores the return value of xgbe_set_mode(), leading to
      false information as the link is declared without checking the status
      register.

      Fix this by propagating the mode switch status information to
      xgbe_phy_status().
```

```
    Fixes: e57f7a3feaef ("amd-xgbe: Prepare for working with more than one type of phy")
    Co-developed-by: Sudheesh Mavila <sudheesh.mavila@amd.com>
    Signed-off-by: Sudheesh Mavila <sudheesh.mavila@amd.com>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Acked-by: Shyam Sundar S K <Shyam-sundar.S-k@amd.com>
    Signed-off-by: Raju Rangoju <Raju.Rangoju@amd.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cdf53b190f1474a1ddb32b67bb0157fe3c349a91
Author: Jakub Kicinski <kuba@kernel.org>
Date:   Wed May 24 22:17:41 2023 -0700

    tls: improve lockless access safety of tls_err_abort()

    [ Upstream commit 8a0d57df8938e9fd2e99d47a85b7f37d86f91097 ]

    Most protos' poll() methods insert a memory barrier between
    writes to sk_err and sk_error_report(). This dates back to
    commit a4d258036ed9 ("tcp: Fix race in tcp_poll").

    I guess we should do the same thing in TLS, tcp_poll() does
    not hold the socket lock.

    Fixes: 3c4d7559159b ("tls: kernel TLS support")
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9e76ba2beb97c0adb7ea5c90910068f4081cca9b
Author: Kuniyuki Iwashima <kuniyu@amazon.com>
Date:   Wed May 24 16:29:34 2023 -0700

    af_packet: Fix data-races of pkt_sk(sk)->num.

    [ Upstream commit 822b5a1c17df7e338b9f05d1cfe5764e37c7f74f ]

    syzkaller found a data race of pkt_sk(sk)->num.

    The value is changed under lock_sock() and po->bind_lock, so we
    need READ_ONCE() to access pkt_sk(sk)->num without these locks in
    packet_bind_spkt(), packet_bind(), and sk_diag_fill().

    Note that WRITE_ONCE() is already added by commit c7d2ef5dd4b0
    ("net/packet: annotate accesses to po->bind").

    BUG: KCSAN: data-race in packet_bind / packet_do_bind

    write (marked) to 0xffff88802ffd1cee of 2 bytes by task 7322 on cpu 0:
     packet_do_bind+0x446/0x640 net/packet/af_packet.c:3236
     packet_bind+0x99/0xe0 net/packet/af_packet.c:3321
     __sys_bind+0x19b/0x1e0 net/socket.c:1803
     __do_sys_bind net/socket.c:1814 [inline]
     __se_sys_bind net/socket.c:1812 [inline]
     __x64_sys_bind+0x40/0x50 net/socket.c:1812
     do_syscall_x64 arch/x86/entry/common.c:50 [inline]
     do_syscall_64+0x3b/0x90 arch/x86/entry/common.c:80
     entry_SYSCALL_64_after_hwframe+0x72/0xdc

    read to 0xffff88802ffd1cee of 2 bytes by task 7318 on cpu 1:
     packet_bind+0xbf/0xe0 net/packet/af_packet.c:3322
     __sys_bind+0x19b/0x1e0 net/socket.c:1803
     __do_sys_bind net/socket.c:1814 [inline]
```

```
      __se_sys_bind net/socket.c:1812 [inline]
      __x64_sys_bind+0x40/0x50 net/socket.c:1812
      do_syscall_x64 arch/x86/entry/common.c:50 [inline]
      do_syscall_64+0x3b/0x90 arch/x86/entry/common.c:80
      entry_SYSCALL_64_after_hwframe+0x72/0xdc

    value changed: 0x0300 -> 0x0000

    Reported by Kernel Concurrency Sanitizer on:
    CPU: 1 PID: 7318 Comm: syz-executor.4 Not tainted 6.3.0-13380-g7fddb5b5300c #4
    Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS rel-1.16.0-0-gd239552ce722-
prebuilt.qemu.org 04/01/2014

    Fixes: 96ec6327144e ("packet: Diag core and basic socket info dumping")
    Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
    Reported-by: syzkaller <syzkaller@googlegroups.com>
    Signed-off-by: Kuniyuki Iwashima <kuniyu@amazon.com>
    Reviewed-by: Willem de Bruijn <willemb@google.com>
    Link: https://lore.kernel.org/r/20230524232934.50950-1-kuniyu@amazon.com
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7c09dc1649bfb0e4569926b9289d7e32bf5401f6
Author: Eric Dumazet <edumazet@google.com>
Date:   Wed May 24 14:14:56 2023 +0000

    netrom: fix info-leak in nr_write_internal()

    [ Upstream commit 31642e7089df8fd3f54ca7843f7ee2952978cad1 ]

    Simon Kapadia reported the following issue:

    <quote>

    The Online Amateur Radio Community (OARC) has recently been experimenting
    with building a nationwide packet network in the UK.
    As part of our experimentation, we have been testing out packet on 300bps HF,
    and playing with net/rom.  For HF packet at this baud rate you really need
    to make sure that your MTU is relatively low; AX.25 suggests a PACLEN of 60,
    and a net/rom PACLEN of 40 to go with that.
    However the Linux net/rom support didn't work with a low PACLEN;
    the mkiss module would truncate packets if you set the PACLEN below about 200 or so,
e.g.:

    Apr 19 14:00:51 radio kernel: [12985.747310] mkiss: ax1: truncating oversized transmit
packet!

    This didn't make any sense to me (if the packets are smaller why would they
    be truncated?) so I started investigating.
    I looked at the packets using ethereal, and found that many were just huge
    compared to what I would expect.
    A simple net/rom connection request packet had the request and then a bunch
    of what appeared to be random data following it:

    </quote>

    Simon provided a patch that I slightly revised:
    Not only we must not use skb_tailroom(), we also do
    not want to count NR_NETWORK_LEN twice.

    Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
    Co-Developed-by: Simon Kapadia <szymon@kapadia.pl>
    Signed-off-by: Simon Kapadia <szymon@kapadia.pl>
    Signed-off-by: Eric Dumazet <edumazet@google.com>
    Tested-by: Simon Kapadia <szymon@kapadia.pl>
```

```
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Link: https://lore.kernel.org/r/20230524141456.1045467-1-edumazet@google.com
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 81306f8277a9fdf70e1fe5ff84c94f3d579d8b03
Author: Wei Fang <wei.fang@nxp.com>
Date:   Wed May 24 20:57:14 2023 +0800

    net: stmmac: fix call trace when stmmac_xdp_xmit() is invoked

    [ Upstream commit ffb3322181d9e8db880202e4f00991764a35d812 ]

    We encountered a kernel call trace issue which was related to
    ndo_xdp_xmit callback on our i.MX8MP platform. The reproduce
    steps show as follows.
    1. The FEC port (eth0) connects to a PC port, and the PC uses
    pktgen_sample03_burst_single_flow.sh to generate packets and
    send these packets to the FEC port. Notice that the script must
    be executed before step 2.
    2. Run the "./xdp_redirect eth0 eth1" command on i.MX8MP, the
    eth1 interface is the dwmac. Then there will be a call trace
    issue soon. Please see the log for more details.
    The root cause is that the NETDEV_XDP_ACT_NDO_XMIT feature is
    enabled by default, so when the step 2 command is exexcuted
    and packets have already been sent to eth0, the stmmac_xdp_xmit()
    starts running before the stmmac_xdp_set_prog() finishes. To
    resolve this issue, we disable the NETDEV_XDP_ACT_NDO_XMIT
    feature by default and turn on/off this feature when the bpf
    program is installed/uninstalled which just like the other
    ethernet drivers.

    Call Trace log:
    [  306.311271] ------------[ cut here ]------------
    [  306.315910] WARNING: CPU: 0 PID: 15 at lib/timerqueue.c:55 timerqueue_del+0x68/0x70
    [  306.323590] Modules linked in:
    [  306.326654] CPU: 0 PID: 15 Comm: ksoftirqd/0 Not tainted 6.4.0-rc1+ #37
    [  306.333277] Hardware name: NXP i.MX8MPlus EVK board (DT)
    [  306.338591] pstate: 600000c5 (nZCv daIF -PAN -UAO -TCO -DIT -SSBS BTYPE=--)
    [  306.345561] pc : timerqueue_del+0x68/0x70
    [  306.349577] lr : __remove_hrtimer+0x5c/0xa0
    [  306.353777] sp : ffff80000b7c3920
    [  306.357094] x29: ffff80000b7c3920 x28: 0000000000000000 x27: 0000000000000001
    [  306.364244] x26: ffff80000a763a40 x25: ffff0000d0285a00 x24: 0000000000000001
    [  306.371390] x23: 0000000000000001 x22: ffff000179389a40 x21: 0000000000000000
    [  306.378537] x20: ffff000179389aa0 x19: ffff0000d2951308 x18: 0000000000001000
    [  306.385686] x17: f1d3000000000000 x16: 00000000c39c1000 x15: 55e99bbe00001a00
    [  306.392835] x14: 09000900120aa8c0 x13: e49af1d300000000 x12: 000000000000c39c
    [  306.399987] x11: 100055e99bbe0000 x10: ffff8000090b1048 x9 : ffff8000081603fc
    [  306.407133] x8 : 000000000000003c x7 : 000000000000003c x6 : 0000000000000001
    [  306.414284] x5 : ffff0000d2950980 x4 : 0000000000000000 x3 : 0000000000000000
    [  306.421432] x2 : 0000000000000001 x1 : ffff0000d2951308 x0 : ffff0000d2951308
    [  306.428585] Call trace:
    [  306.431035]  timerqueue_del+0x68/0x70
    [  306.434706]  __remove_hrtimer+0x5c/0xa0
    [  306.438549]  hrtimer_start_range_ns+0x2bc/0x370
    [  306.443089]  stmmac_xdp_xmit+0x174/0x1b0
    [  306.447021]  bq_xmit_all+0x194/0x4b0
    [  306.450612]  __dev_flush+0x4c/0x98
    [  306.454024]  xdp_do_flush+0x18/0x38
    [  306.457522]  fec_enet_rx_napi+0x6c8/0xc68
    [  306.461539]  __napi_poll+0x40/0x220
    [  306.465038]  net_rx_action+0xf8/0x240
    [  306.468707]  __do_softirq+0x128/0x3a8
    [  306.472378]  run_ksoftirqd+0x40/0x58
```

```
      [  306.475961]  smpboot_thread_fn+0x1c4/0x288
      [  306.480068]  kthread+0x124/0x138
      [  306.483305]  ret_from_fork+0x10/0x20
      [  306.486889] ---[ end trace 0000000000000000 ]---

      Fixes: 66c0e13ad236 ("drivers: net: turn on XDP features")
      Signed-off-by: Wei Fang <wei.fang@nxp.com>
      Reviewed-by: Simon Horman <simon.horman@corigine.com>
      Link: https://lore.kernel.org/r/20230524125714.357337-1-wei.fang@nxp.com
      Signed-off-by: Jakub Kicinski <kuba@kernel.org>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 740f37c352ce914b09c5c50713ee349f0822bd5e
  Author: Thomas Bogendoerfer <tbogendoerfer@suse.de>
  Date:   Wed May 24 21:49:08 2023 +0200

      net: mellanox: mlxbf_gige: Fix skb_panic splat under memory pressure

      [ Upstream commit d68cb7cf1fd0ef4287bc0ecd1ed0b6ae8e05fc70 ]

      Do skb_put() after a new skb has been successfully allocated otherwise
      the reused skb leads to skb_panics or incorrect packet sizes.

      Fixes: f92e1869d74e ("Add Mellanox BlueField Gigabit Ethernet driver")
      Signed-off-by: Thomas Bogendoerfer <tbogendoerfer@suse.de>
      Reviewed-by: Simon Horman <simon.horman@corigine.com>
      Link: https://lore.kernel.org/r/20230524194908.147145-1-tbogendoerfer@suse.de
      Signed-off-by: Jakub Kicinski <kuba@kernel.org>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 7722d3e41cf202b7d7c4a4b3e74e6c1dfec79cdc
  Author: Jianbo Liu <jianbol@nvidia.com>
  Date:   Tue May 16 02:28:02 2023 +0000

      net/mlx5e: Move Ethernet driver debugfs to profile init callback

      [ Upstream commit c4c24fc30cc417ace332ceceaba4f70f81dcd521 ]

      As priv->dfs_root is cleared, and therefore missed, when change
      eswitch mode, move the creation of the root debugfs to the init
      callback of mlx5e_nic_profile and mlx5e_uplink_rep_profile, and
      the destruction to the cleanup callback for symmeter.

      Fixes: 288eca60cc31 ("net/mlx5e: Add Ethernet driver debugfs")
      Signed-off-by: Jianbo Liu <jianbol@nvidia.com>
      Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 75a13d8a47439a6d4d1353d456d4a6d80f25e3f8
  Author: Dmytro Linkin <dlinkin@nvidia.com>
  Date:   Wed Oct 13 14:39:24 2021 +0300

      net/mlx5e: Don't attach netdev profile while handling internal error

      [ Upstream commit bdf274750fca17b289404ef03453c4070725302c ]

      As part of switchdev mode disablement, driver changes port netdevice
      profile from uplink to nic. If this process is triggered by health
      recovery flow (PCI reset, for ex.) profile attach would fail because all
      fw commands aborted when internal error flag is set. As a result, nic
      netdevice profile is not attached and driver fails to rollback to uplink
      profile, which leave driver in broken state and cause crash later.

      To handle broken state do netdevice profile initialization only instead
      of full attachment and release mdev resources on driver suspend as
```

```
    expected. Actual netdevice attachment is done during driver load.

    Fixes: c4d7eb57687f ("net/mxl5e: Add change profile method")
    Signed-off-by: Dmytro Linkin <dlinkin@nvidia.com>
    Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit aeed0505b92a6aecd4879c61e4c1d279661a2afe
Author: Vlad Buslov <vladbu@nvidia.com>
Date:   Mon May 22 14:48:52 2023 +0200

    net/mlx5: Fix post parse infra to only parse every action once

    [ Upstream commit 5d862ec631f3d3cc3b4f8cdb5b9fc5879663f1d3 ]

    Caller of mlx5e_tc_act_post_parse() needs it to parse only the subset of
    actions starting after previous split and ending at the current action.
    However, that range is not provided as arguments and
    mlx5e_tc_act_post_parse() uses generic flow_action_for_each() that iterates
    over all flow actions. Not only this is redundant, it also causes a bug
    when mlx5e_tc_act->post_parse() callback is not idempotent since it will be
    called for every split. For example, ct action tc_act_post_parse_ct()
    callback obtains a reference to mlx5_ct_ft instance and calling it several
    times during parsing stage will cause reference counter imbalance.

    Fix the issue by providing a proper action range of the current split
    subset to mlx5e_tc_act_post_parse() and only calling
    mlx5e_tc_act->post_parse() for actions inside the subset range.

    Fixes: 8300f225268b ("net/mlx5e: Create new flow attr for multi table actions")
    Signed-off-by: Vlad Buslov <vladbu@nvidia.com>
    Reviewed-by: Roi Dayan <roid@nvidia.com>
    Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ba7ef5bb81d41167a6fcd50cefeb682c80e686d2
Author: Paul Blakey <paulb@nvidia.com>
Date:   Sun Dec 4 16:53:56 2022 +0200

    net/mlx5e: TC, Remove CT action reordering

    [ Upstream commit 67efaf45930df662111acf7c706d545c83f83999 ]

    CT action reordering was done as a workaround when CT misses
    used to restore the relevant filter's tc chain and continuing sw processing
    from that chain. As such, there was a need to reorder CT action to be before
    any packet modifying actions (e.g mac rewrite).

    Currently (after patch "net/mlx5e: TC, Set CT miss to the specific ct
    action instance"), CT misses continues from the relevant ct action in
    software, and so reordering isn't needed anymore.

    Remove the reordering.

    Signed-off-by: Paul Blakey <paulb@nvidia.com>
    Reviewed-by: Roi Dayan <roid@nvidia.com>
    Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
    Stable-dep-of: 5d862ec631f3 ("net/mlx5: Fix post parse infra to only parse every action
once")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2ba3653a422a277c17f3bd9481c6d83696efbafe
Author: Paul Blakey <paulb@nvidia.com>
Date:   Sun Feb 12 19:11:07 2023 +0200
```

```
      net/mlx5e: CT: Use per action stats

      [ Upstream commit 13aca17b450e87a8de4e4a3b3ad454efbc576740 ]

      CT action can miss in a middle of an action list, use
      per action stats to correctly report stats for missed
      packets.

      Signed-off-by: Paul Blakey <paulb@nvidia.com>
      Reviewed-by: Roi Dayan <roid@nvidia.com>
      Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
      Stable-dep-of: 5d862ec631f3 ("net/mlx5: Fix post parse infra to only parse every action
  once")
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit c85e8edcb2d554ba46d31ab09c1a9111974b68ab
  Author: Dragos Tatulea <dtatulea@nvidia.com>
  Date:   Thu Apr 13 15:48:30 2023 +0300

      net/mlx5e: Use query_special_contexts cmd only once per mdev

      [ Upstream commit 1db1f21caebbb1b6e9b1e7657df613616be3fb49 ]

      Don't query the firmware so many times (num rqs * num wqes * wqe frags)
      because it slows down linearly the interface creation time when the
      product is larger. Do it only once per mdev and store the result in
      mlx5e_param.

      Due to helper function being called from different files, move it to
      an appropriate location. Rename the function with a proper prefix and
      add a small cleanup.

      This fix applies only for legacy rq.

      Fixes: 1b1e4868836a ("net/mlx5e: Use query_special_contexts for mkeys")
      Signed-off-by: Dragos Tatulea <dtatulea@nvidia.com>
      Reviewed-by: Or Har-Toov <ohartoov@nvidia.com>
      Reviewed-by: Tariq Toukan <tariqt@nvidia.com>
      Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 00d46d576d7c4635f83da262a5d5477ca7419c65
  Author: Shay Drory <shayd@nvidia.com>
  Date:   Sat Apr 29 20:41:41 2023 +0300

      net/mlx5: fw_tracer, Fix event handling

      [ Upstream commit 341a80de2468f481b1f771683709b5649cbfe513 ]

      mlx5 driver needs to parse traces with event_id inside the range of
      first_string_trace and num_string_trace. However, mlx5 is parsing all
      events with event_id >= first_string_trace.

      Fix it by checking for the correct range.

      Fixes: c71ad41ccb0c ("net/mlx5: FW tracer, events handling")
      Signed-off-by: Shay Drory <shayd@nvidia.com>
      Reviewed-by: Moshe Shemesh <moshe@nvidia.com>
      Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit b16afe0f450760d31e672d2d952c4cfd640c32fd
  Author: Shay Drory <shayd@nvidia.com>
  Date:   Mon Apr 24 12:46:06 2023 +0300
```

```
    net/mlx5: SF, Drain health before removing device

    [ Upstream commit b4646da0573fae9dfa2b8f1f10936cb6eedd7230 ]

    There is no point in recovery during device removal. Also, if health
    work started need to wait for it to avoid races and NULL pointer
    access.

    Hence, drain health WQ before removing device.

    Fixes: 1958fc2f0712 ("net/mlx5: SF, Add auxiliary device driver")
    Signed-off-by: Shay Drory <shayd@nvidia.com>
    Reviewed-by: Moshe Shemesh <moshe@nvidia.com>
    Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 96b064c8db078442d5754c2b8c2004cdddf71a7a
Author: Shay Drory <shayd@nvidia.com>
Date:   Mon Apr 24 12:31:59 2023 +0300

    net/mlx5: Drain health before unregistering devlink

    [ Upstream commit 824c8dc4a470040bf0e56ba716543839c2498d49 ]

    mlx5 health mechanism is using devlink APIs, which are using devlink
    notify APIs. After the cited patch, using devlink notify APIs after
    devlink is unregistered triggers a WARN_ON().
    Hence, drain health WQ before devlink is unregistered.

    Fixes: cf530217408e ("devlink: Notify users when objects are accessible")
    Signed-off-by: Shay Drory <shayd@nvidia.com>
    Reviewed-by: Moshe Shemesh <moshe@nvidia.com>
    Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit aa830a0f55a800b100a66b9326804ddfd905c815
Author: Maher Sanalla <msanalla@nvidia.com>
Date:   Tue May 9 17:56:01 2023 +0300

    net/mlx5e: Do not update SBCM when prio2buffer command is invalid

    [ Upstream commit 623efc4cbd6115db36716e31037cb6d1f3ce6754 ]

    The shared buffer pools configuration which are stored in the SBCM
    register are updated when the user changes the prio2buffer mapping.

    However, in case the user desired prio2buffer change is invalid,
    which can occur due to mapping a lossless priority to a not large enough
    buffer, the SBCM update should not be performed, as the user command is
    failed.

    Thus, Perform the SBCM update only after xoff threshold calculation is
    performed and the user prio2buffer mapping is validated.

    Fixes: a440030d8946 ("net/mlx5e: Update shared buffer along with device buffer changes")
    Signed-off-by: Maher Sanalla <msanalla@nvidia.com>
    Reviewed-by: Moshe Shemesh <moshe@nvidia.com>
    Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8520caef25faa699033920fa71bc3aedd06d933b
Author: Maher Sanalla <msanalla@nvidia.com>
Date:   Mon May 1 17:31:40 2023 +0300

    net/mlx5e: Consider internal buffers size in port buffer calculations
```

```
    [ Upstream commit 81fe2be062915e2a2fdc494c3cd90e946e946c25 ]

    Currently, when a user triggers a change in port buffer headroom
    (buffers 0-7), the driver checks that the requested headroom does
    not exceed the total port buffer size. However, this check does not
    take into account the internal buffers (buffers 8-9), which are also
    part of the total port buffer. This can result in treating invalid port
    buffer change requests as valid, causing unintended changes to the shared
    buffer.

    To address this, include the internal buffers size in the calculation of
    available port buffer space which ensures that port buffer requests do not
    exceed the correct limit.

    Furthermore, remove internal buffers (8-9) size from the total_size
    calculation as these buffers are reserved for internal use and are not
    exposed to the user.

    While at it, add verbosity to the debug prints in
    mlx5e_port_query_buffer() function to ease future debugging.

    Fixes: ecdf2dadee8e ("net/mlx5e: Receive buffer support for DCBX")
    Signed-off-by: Maher Sanalla <msanalla@nvidia.com>
    Reviewed-by: Moshe Shemesh <moshe@nvidia.com>
    Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 095ad8cf7b1ce92780dd58cbbf2a58dca64ec782
Author: Chris Mi <cmi@nvidia.com>
Date:   Tue Feb 21 04:41:41 2023 +0200

    net/mlx5e: Prevent encap offload when neigh update is running

    [ Upstream commit 37c3b9fa7ccf5caad6d87ba4d42bf00be46be1cf ]

    The cited commit adds a compeletion to remove dependency on rtnl
    lock. But it causes a deadlock for multiple encapsulations:

     crash> bt ffff8aece8a64000
     PID: 1514557  TASK: ffff8aece8a64000  CPU: 3    COMMAND: "tc"
      #0 [ffffa6d14183f368] __schedule at ffffffffb8ba7f45
      #1 [ffffa6d14183f3f8] schedule at ffffffffb8ba8418
      #2 [ffffa6d14183f418] schedule_preempt_disabled at ffffffffb8ba8898
      #3 [ffffa6d14183f428] __mutex_lock at ffffffffb8baa7f8
      #4 [ffffa6d14183f4d0] mutex_lock_nested at ffffffffb8baabeb
      #5 [ffffa6d14183f4e0] mlx5e_attach_encap at ffffffffc0f48c17 [mlx5_core]
      #6 [ffffa6d14183f628] mlx5e_tc_add_fdb_flow at ffffffffc0f39680 [mlx5_core]
      #7 [ffffa6d14183f688] __mlx5e_add_fdb_flow at ffffffffc0f3b636 [mlx5_core]
      #8 [ffffa6d14183f6f0] mlx5e_tc_add_flow at ffffffffc0f3bcdf [mlx5_core]
      #9 [ffffa6d14183f728] mlx5e_configure_flower at ffffffffc0f3c1d1 [mlx5_core]
     #10 [ffffa6d14183f790] mlx5e_rep_setup_tc_cls_flower at ffffffffc0f3d529 [mlx5_core]
     #11 [ffffa6d14183f7a0] mlx5e_rep_setup_tc_cb at ffffffffc0f3d714 [mlx5_core]
     #12 [ffffa6d14183f7b0] tc_setup_cb_add at ffffffffb8931bb8
     #13 [ffffa6d14183f810] fl_hw_replace_filter at ffffffffc0dae901 [cls_flower]
     #14 [ffffa6d14183f8d8] fl_change at ffffffffc0db5c57 [cls_flower]
     #15 [ffffa6d14183f970] tc_new_tfilter at ffffffffb8936047
     #16 [ffffa6d14183fac8] rtnetlink_rcv_msg at ffffffffb88c7c31
     #17 [ffffa6d14183fb50] netlink_rcv_skb at ffffffffb8942853
     #18 [ffffa6d14183fbc0] rtnetlink_rcv at ffffffffb88c1835
     #19 [ffffa6d14183fbd0] netlink_unicast at ffffffffb8941f27
     #20 [ffffa6d14183fc18] netlink_sendmsg at ffffffffb8942245
     #21 [ffffa6d14183fc98] sock_sendmsg at ffffffffb887d482
     #22 [ffffa6d14183fcb8] ____sys_sendmsg at ffffffffb887d81a
     #23 [ffffa6d14183fd38] ___sys_sendmsg at ffffffffb88806e2
```

```
 #24 [ffffa6d14183fe90] __sys_sendmsg at ffffffffb88807a2
 #25 [ffffa6d14183ff28] __x64_sys_sendmsg at ffffffffb888080f
 #26 [ffffa6d14183ff38] do_syscall_64 at ffffffffb8b9b6a8
 #27 [ffffa6d14183ff50] entry_SYSCALL_64_after_hwframe at ffffffffb8c0007c
 crash> bt 0xffff8aeb07544000
 PID: 1110766  TASK: ffff8aeb07544000  CPU: 0    COMMAND: "kworker/u20:9"
  #0 [ffffa6d14e6b7bd8] __schedule at ffffffffb8ba7f45
  #1 [ffffa6d14e6b7c68] schedule at ffffffffb8ba8418
  #2 [ffffa6d14e6b7c88] schedule_timeout at ffffffffb8baef88
  #3 [ffffa6d14e6b7d10] wait_for_completion at ffffffffb8ba968b
  #4 [ffffa6d14e6b7d60] mlx5e_take_all_encap_flows at ffffffffc0f47ec4 [mlx5_core]
  #5 [ffffa6d14e6b7da0] mlx5e_rep_update_flows at ffffffffc0f3e734 [mlx5_core]
  #6 [ffffa6d14e6b7df8] mlx5e_rep_neigh_update at ffffffffc0f400bb [mlx5_core]
  #7 [ffffa6d14e6b7e50] process_one_work at ffffffffb80acc9c
  #8 [ffffa6d14e6b7ed0] worker_thread at ffffffffb80ad012
  #9 [ffffa6d14e6b7f10] kthread at ffffffffb80b615d
 #10 [ffffa6d14e6b7f50] ret_from_fork at ffffffffb8001b2f

    After the first encap is attached, flow will be added to encap
    entry's flows list. If neigh update is running at this time, the
    following encaps of the flow can't hold the encap_tbl_lock and
    sleep. If neigh update thread is waiting for that flow's init_done,
    deadlock happens.

    Fix it by holding lock outside of the for loop. If neigh update is
    running, prevent encap flows from offloading. Since the lock is held
    outside of the for loop, concurrent creation of encap entries is not
    allowed. So remove unnecessary wait_for_completion call for res_ready.

    Fixes: 95435ad7999b ("net/mlx5e: Only access fully initialized flows in neigh update")
    Signed-off-by: Chris Mi <cmi@nvidia.com>
    Reviewed-by: Roi Dayan <roid@nvidia.com>
    Reviewed-by: Vlad Buslov <vladbu@nvidia.com>
    Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2c24a9aa8720353c643b014f1e57f7d13de65b55
Author: Chris Mi <cmi@nvidia.com>
Date:   Wed Mar 1 10:50:53 2023 +0200

    net/mlx5e: Extract remaining tunnel encap code to dedicated file

    [ Upstream commit e2ab5aa11f191b54514f063a5b5c29f3559f4ab7 ]

    Move set_encap_dests() and clean_encap_dests() to the tunnel encap
    dedicated file. And rename them to mlx5e_tc_tun_encap_dests_set()
    and mlx5e_tc_tun_encap_dests_unset().

    No functional change in this patch. It is needed in the next patch.

    Signed-off-by: Chris Mi <cmi@nvidia.com>
    Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
    Stable-dep-of: 37c3b9fa7ccf ("net/mlx5e: Prevent encap offload when neigh update is
running")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 435091012bf3a866982fd6b94ec5a9649e166158
Author: Paul Blakey <paulb@nvidia.com>
Date:   Wed Jan 25 11:54:22 2023 +0200

    net/mlx5e: TC, Move main flow attribute cleanup to helper func

    [ Upstream commit a830ec485e8368a29e328d08e2eb28750bbc483f ]

    Actions that can be setup per flow attribute (so per split rule)
```

```
    are cleaned up from mlx5_free_flow_attr(), mlx5e_tc_del_fdb_flow(),
    and free_flow_post_acts().

    Remove the duplication by re-using the helper function for
    the main flow attribute and split rules attributes.

    Signed-off-by: Paul Blakey <paulb@nvidia.com>
    Reviewed-by: Roi Dayan <roid@nvidia.com>
    Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
    Stable-dep-of: 37c3b9fa7ccf ("net/mlx5e: Prevent encap offload when neigh update is
running")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 431d1b25efb56311a74b6c3ecfda71f984c00f56
Author: Paul Blakey <paulb@nvidia.com>
Date:   Thu Dec 1 12:04:58 2022 +0200

    net/mlx5e: TC, Remove unused vf_tun variable

    [ Upstream commit 7195d9a0c8df0ab78c9d7a587809d16b00432426 ]

    vf_tun is being assigned but never being used so remove it.

    Signed-off-by: Paul Blakey <paulb@nvidia.com>
    Reviewed-by: Roi Dayan <roid@nvidia.com>
    Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
    Stable-dep-of: 37c3b9fa7ccf ("net/mlx5e: Prevent encap offload when neigh update is
running")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ea67d47fd0d7f57ba379b317c4de80e4e9889096
Author: Alexandre Ghiti <alexghiti@rivosinc.com>
Date:   Fri May 19 15:13:11 2023 +0200

    riscv: Fix unused variable warning when BUILTIN_DTB is set

    [ Upstream commit 33d418da6f476b15e4510e0a590062583f63cd36 ]

    commit ef69d2559fe9 ("riscv: Move early dtb mapping into the fixmap
    region") wrongly moved the #ifndef CONFIG_BUILTIN_DTB surrounding the pa
    variable definition in create_fdt_early_page_table(), so move it back to
    its right place to quiet the following warning:

    ../arch/riscv/mm/init.c: In function ‘create_fdt_early_page_table’:
    ../arch/riscv/mm/init.c:925:12: warning: unused variable ‘pa’ [-Wunused-variable]
      925 |  uintptr_t pa = dtb_pa & ~(PMD_SIZE - 1);

    Fixes: ef69d2559fe9 ("riscv: Move early dtb mapping into the fixmap region")
    Signed-off-by: Alexandre Ghiti <alexghiti@rivosinc.com>
    Reviewed-by: Conor Dooley <conor.dooley@microchip.com>
    Link: https://lore.kernel.org/r/20230519131311.391960-1-alexghiti@rivosinc.com
    Signed-off-by: Palmer Dabbelt <palmer@rivosinc.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7dca248f2899584b4b19622f833cb7b7e82ed8bf
Author: Ard Biesheuvel <ardb@kernel.org>
Date:   Tue May 16 20:14:19 2023 +0200

    crypto: x86/aria - Use 16 byte alignment for GFNI constant vectors

    [ Upstream commit 6ab39f99927eed605728b02d512438d828183c97 ]

    The GFNI routines in the AVX version of the ARIA implementation now use
    explicit VMOVDQA instructions to load the constant input vectors, which
    means they must be 16 byte aligned. So ensure that this is the case, by
```

```
    dropping the section split and the incorrect .align 8 directive, and
    emitting the constants into the 16-byte aligned section instead.

    Note that the AVX2 version of this code deviates from this pattern, and
    does not require a similar fix, given that it loads these contants as
    8-byte memory operands, for which AVX2 permits any alignment.

    Cc: Taehee Yoo <ap420073@gmail.com>
    Fixes: 8b84475318641c2b ("crypto: x86/aria-avx - Do not use avx2 instructions")
    Reported-by: syzbot+a6abcf08bad8b18fd198@syzkaller.appspotmail.com
    Tested-by: syzbot+a6abcf08bad8b18fd198@syzkaller.appspotmail.com
    Signed-off-by: Ard Biesheuvel <ardb@kernel.org>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 74c08d4352c52dae63791fb7fead4312a115eda5
Author: Randy Dunlap <rdunlap@infradead.org>
Date:   Tue May 23 21:53:10 2023 -0700

    dmaengine: pl330: rename _start to prevent build error

    [ Upstream commit a1a5f2c887252dec161c1e12e04303ca9ba56fa9 ]

    "_start" is used in several arches and proably should be reserved
    for ARCH usage. Using it in a driver for a private symbol can cause
    a build error when it conflicts with ARCH usage of the same symbol.

    Therefore rename pl330's "_start" to "pl330_start_thread" so that there
    is no conflict and no build error.

    drivers/dma/pl330.c:1053:13: error: '_start' redeclared as different kind of symbol
     1053 | static bool _start(struct pl330_thread *thrd)
          |             ^~~~~~
    In file included from ../include/linux/interrupt.h:21,
                     from ../drivers/dma/pl330.c:18:
    arch/riscv/include/asm/sections.h:11:13: note: previous declaration of '_start' with
type 'char[]'
       11 | extern char _start[];
          |             ^~~~~~

    Fixes: b7d861d93945 ("DMA: PL330: Merge PL330 driver into drivers/dma/")
    Fixes: ae43b3289186 ("ARM: 8202/1: dmaengine: pl330: Add runtime Power Management
support v12")
    Signed-off-by: Randy Dunlap <rdunlap@infradead.org>
    Cc: Jaswinder Singh <jassisinghbrar@gmail.com>
    Cc: Boojin Kim <boojin.kim@samsung.com>
    Cc: Krzysztof Kozlowski <krzk@kernel.org>
    Cc: Russell King <rmk+kernel@arm.linux.org.uk>
    Cc: Vinod Koul <vkoul@kernel.org>
    Cc: dmaengine@vger.kernel.org
    Cc: linux-riscv@lists.infradead.org
    Link: https://lore.kernel.org/r/20230524045310.27923-1-rdunlap@infradead.org
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c2d8941aef07d93e7f126e66c1132fbb9aa40888
Author: Jeff Layton <jlayton@kernel.org>
Date:   Wed May 17 12:26:44 2023 -0400

    nfsd: make a copy of struct iattr before calling notify_change

    [ Upstream commit d53d70084d27f56bcdf5074328f2c9ec861be596 ]

    notify_change can modify the iattr structure. In particular it can
    end up setting ATTR_MODE when ATTR_KILL_SUID is already set, causing
```

```
    a BUG() if the same iattr is passed to notify_change more than once.

    Make a copy of the struct iattr before calling notify_change.

    Reported-by: Zhi Li <yieli@redhat.com>
    Link: https://bugzilla.redhat.com/show_bug.cgi?id=2207969
    Tested-by: Zhi Li <yieli@redhat.com>
    Fixes: 34b91dda7124 ("NFSD: Make nfsd4_setattr() wait before returning NFS4ERR_DELAY")
    Signed-off-by: Jeff Layton <jlayton@kernel.org>
    Signed-off-by: Chuck Lever <chuck.lever@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 28d542e8caf3344141f3b08c7d4f97db7e2fb6c5
Author: Jason Gunthorpe <jgg@ziepe.ca>
Date:   Tue Apr 25 16:04:15 2023 -0300

    iommu/amd: Add missing domain type checks

    [ Upstream commit 29f54745f24547a84b18582e054df9bea1a7bf3e ]

    Drivers are supposed to list the domain types they support in their
    domain_alloc() ops so when we add new domain types, like BLOCKING or SVA,
    they don't start breaking.

    This ended up providing an empty UNMANAGED domain when the core code asked
    for a BLOCKING domain, which happens to be the fallback for drivers that
    don't support it, but this is completely wrong for SVA.

    Check for the DMA types AMD supports and reject every other kind.

    Fixes: 136467962e49 ("iommu: Add IOMMU SVA domain support")
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Reviewed-by: Vasant Hegde <vasant.hegde@amd.com>
    Reviewed-by: Kevin Tian <kevin.tian@intel.com>
    Link: https://lore.kernel.org/r/0-v1-2ac37b893728+da-amd_check_types_jgg@nvidia.com
    Signed-off-by: Joerg Roedel <jroedel@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0e481712cbebd6c8188677a7c1153c134a48ef67
Author: Jerry Snitselaar <jsnitsel@redhat.com>
Date:   Thu Apr 20 12:20:13 2023 -0700

    iommu/amd: Fix up merge conflict resolution

    [ Upstream commit 8ec4e2befef10c7679cd59251956a428e783c0b5 ]

    Merge commit e17c6debd4b2 ("Merge branches 'arm/mediatek', 'arm/msm', 'arm/renesas',
'arm/rockchip', 'arm/smmu', 'x86/vt-d' and 'x86/amd' into next")
    added amd_iommu_init_devices, amd_iommu_uninit_devices,
    and amd_iommu_init_notifier back to drivers/iommu/amd/amd_iommu.h.
    The only references to them are here, so clean them up.

    Fixes: e17c6debd4b2 ("Merge branches 'arm/mediatek', 'arm/msm', 'arm/renesas',
'arm/rockchip', 'arm/smmu', 'x86/vt-d' and 'x86/amd' into next")
    Cc: Joerg Roedel <joro@8bytes.org>
    Cc: Suravee Suthikulpanit <suravee.suthikulpanit@amd.com>
    Cc: Will Deacon <will@kernel.org>
    Cc: Robin Murphy <robin.murphy@arm.com>
    Signed-off-by: Jerry Snitselaar <jsnitsel@redhat.com>
    Reviewed-by: Vasant Hegde <vasant.hegde@amd.com>
    Link: https://lore.kernel.org/r/20230420192013.733331-1-jsnitsel@redhat.com
    Signed-off-by: Joerg Roedel <jroedel@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c799ad3ab386d73375c931ef5b067805a5d380f8
```

```
Author: Joao Martins <joao.m.martins@oracle.com>
Date:   Wed Apr 19 21:11:54 2023 +0100

    iommu/amd: Handle GALog overflows

    [ Upstream commit af47b0a24058e56e983881993752f88288ca6511 ]

    GALog exists to propagate interrupts into all vCPUs in the system when
    interrupts are marked as non running (e.g. when vCPUs aren't running). A
    GALog overflow happens when there's in no space in the log to record the
    GATag of the interrupt. So when the GALOverflow condition happens, the
    GALog queue is processed and the GALog is restarted, as the IOMMU
    manual indicates in section "2.7.4 Guest Virtual APIC Log Restart
    Procedure":

    | * Wait until MMIO Offset 2020h[GALogRun]=0b so that all request
    |   entries are completed as circumstances allow. GALogRun must be 0b to
    |   modify the guest virtual APIC log registers safely.
    | * Write MMIO Offset 0018h[GALogEn]=0b.
    | * As necessary, change the following values (e.g., to relocate or
    | resize the guest virtual APIC event log):
    |   - the Guest Virtual APIC Log Base Address Register
    |      [MMIO Offset 00E0h],
    |   - the Guest Virtual APIC Log Head Pointer Register
    |      [MMIO Offset 2040h][GALogHead], and
    |   - the Guest Virtual APIC Log Tail Pointer Register
    |      [MMIO Offset 2048h][GALogTail].
    | * Write MMIO Offset 2020h[GALOverflow] = 1b to clear the bit (W1C).
    | * Write MMIO Offset 0018h[GALogEn] = 1b, and either set
    |   MMIO Offset 0018h[GAIntEn] to enable the GA log interrupt or clear
    |   the bit to disable it.

    Failing to handle the GALog overflow means that none of the VFs (in any
    guest) will work with IOMMU AVIC forcing the user to power cycle the
    host. When handling the event it resumes the GALog without resizing
    much like how it is done in the event handler overflow. The
    [MMIO Offset 2020h][GALOverflow] bit might be set in status register
    without the [MMIO Offset 2020h][GAInt] bit, so when deciding to poll
    for GA events (to clear space in the galog), also check the overflow
    bit.

    [suravee: Check for GAOverflow without GAInt, toggle CONTROL_GAINT_EN]

    Co-developed-by: Suravee Suthikulpanit <suravee.suthikulpanit@amd.com>
    Signed-off-by: Suravee Suthikulpanit <suravee.suthikulpanit@amd.com>
    Signed-off-by: Joao Martins <joao.m.martins@oracle.com>
    Reviewed-by: Vasant Hegde <vasant.hegde@amd.com>
    Link: https://lore.kernel.org/r/20230419201154.83880-3-joao.m.martins@oracle.com
    Signed-off-by: Joerg Roedel <jroedel@suse.de>
    Stable-dep-of: 8ec4e2befef1 ("iommu/amd: Fix up merge conflict resolution")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 781561c1ad74f2df35069efb12ee44a0178018ef
Author: Joao Martins <joao.m.martins@oracle.com>
Date:   Wed Apr 19 21:11:53 2023 +0100

    iommu/amd: Don't block updates to GATag if guest mode is on

    [ Upstream commit ed8a2f4ddef2eaaf864ab1efbbca9788187036ab ]

    On KVM GSI routing table updates, specially those where they have vIOMMUs
    with interrupt remapping enabled (to boot >255vcpus setups without relying
    on KVM_FEATURE_MSI_EXT_DEST_ID), a VMM may update the backing VF MSIs
    with a new VCPU affinity.
```

```
    On AMD with AVIC enabled, the new vcpu affinity info is updated via:
            avic_pi_update_irte()
                    irq_set_vcpu_affinity()
                            amd_ir_set_vcpu_affinity()
                                    amd_iommu_{de}activate_guest_mode()

    Where the IRTE[GATag] is updated with the new vcpu affinity. The GATag
    contains VM ID and VCPU ID, and is used by IOMMU hardware to signal KVM
    (via GALog) when interrupt cannot be delivered due to vCPU is in
    blocking state.

    The issue is that amd_iommu_activate_guest_mode() will essentially
    only change IRTE fields on transitions from non-guest-mode to guest-mode
    and otherwise returns *with no changes to IRTE* on already configured
    guest-mode interrupts. To the guest this means that the VF interrupts
    remain affined to the first vCPU they were first configured, and guest
    will be unable to issue VF interrupts and receive messages like this
    from spurious interrupts (e.g. from waking the wrong vCPU in GALog):

    [  167.759472] __common_interrupt: 3.34 No irq handler for vector
    [  230.680927] mlx5_core 0000:00:02.0: mlx5_cmd_eq_recover:247:(pid
    3122): Recovered 1 EQEs on cmd_eq
    [  230.681799] mlx5_core 0000:00:02.0:
    wait_func_handle_exec_timeout:1113:(pid 3122): cmd[0]: CREATE_CQ(0x400)
    recovered after timeout
    [  230.683266] __common_interrupt: 3.34 No irq handler for vector

    Given the fact that amd_ir_set_vcpu_affinity() uses
    amd_iommu_activate_guest_mode() underneath it essentially means that VCPU
    affinity changes of IRTEs are nops. Fix it by dropping the check for
    guest-mode at amd_iommu_activate_guest_mode(). Same thing is applicable to
    amd_iommu_deactivate_guest_mode() although, even if the IRTE doesn't change
    underlying DestID on the host, the VFIO IRQ handler will still be able to
    poke at the right guest-vCPU.

    Fixes: b9c6ff94e43a ("iommu/amd: Re-factor guest virtual APIC (de-)activation code")
    Signed-off-by: Joao Martins <joao.m.martins@oracle.com>
    Reviewed-by: Suravee Suthikulpanit <suravee.suthikulpanit@amd.com>
    Link: https://lore.kernel.org/r/20230419201154.83880-2-joao.m.martins@oracle.com
    Signed-off-by: Joerg Roedel <jroedel@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 184dda03dc0a803faa2a34aace2a6a1eedbaf9af
Author: Chao Wang <D202280639@hust.edu.cn>
Date:   Mon Apr 17 03:04:21 2023 +0000

    iommu/rockchip: Fix unwind goto issue

    [ Upstream commit ec014683c564fb74fc68e8f5e84691d3b3839d24 ]

    Smatch complains that
    drivers/iommu/rockchip-iommu.c:1306 rk_iommu_probe() warn: missing unwind goto?

    The rk_iommu_probe function, after obtaining the irq value through
    platform_get_irq, directly returns an error if the returned value
    is negative, without releasing any resources.

    Fix this by adding a new error handling label "err_pm_disable" and
    use a goto statement to redirect to the error handling process. In
    order to preserve the original semantics, set err to the value of irq.

    Fixes: 1aa55ca9b14a ("iommu/rockchip: Move irq request past pm_runtime_enable")
    Signed-off-by: Chao Wang <D202280639@hust.edu.cn>
    Reviewed-by: Dongliang Mu <dzm91@hust.edu.cn>
    Reviewed-by: Heiko Stuebner <heiko@sntech.de>
```

```
    Link: https://lore.kernel.org/r/20230417030421.2777-1-D202280639@hust.edu.cn
    Signed-off-by: Joerg Roedel <jroedel@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1d16b07334758d1808c68755f4b2ad3a6c300058
Author: Randy Dunlap <rdunlap@infradead.org>
Date:   Thu Mar 30 09:58:17 2023 -0700

    iommu: Make IPMMU_VMSA dependencies more strict

    [ Upstream commit e332003bb216a9f91e08004b9e2de0745f321290 ]

    On riscv64, linux-next-20233030 (and for several days earlier),
    there is a kconfig warning:

    WARNING: unmet direct dependencies detected for IOMMU_IO_PGTABLE_LPAE
      Depends on [n]: IOMMU_SUPPORT [=y] && (ARM || ARM64 || COMPILE_TEST [=n]) &&
!GENERIC_ATOMIC64 [=n]
      Selected by [y]:
      - IPMMU_VMSA [=y] && IOMMU_SUPPORT [=y] && (ARCH_RENESAS [=y] || COMPILE_TEST [=n]) &&
!GENERIC_ATOMIC64 [=n]

    and build errors:

    riscv64-linux-ld: drivers/iommu/io-pgtable-arm.o: in function `.L140':
    io-pgtable-arm.c:(.init.text+0x1e8): undefined reference to `alloc_io_pgtable_ops'
    riscv64-linux-ld: drivers/iommu/io-pgtable-arm.o: in function `.L168':
    io-pgtable-arm.c:(.init.text+0xab0): undefined reference to `free_io_pgtable_ops'
    riscv64-linux-ld: drivers/iommu/ipmmu-vmsa.o: in function `.L140':
    ipmmu-vmsa.c:(.text+0xbc4): undefined reference to `free_io_pgtable_ops'
    riscv64-linux-ld: drivers/iommu/ipmmu-vmsa.o: in function `.L0 ':
    ipmmu-vmsa.c:(.text+0x145e): undefined reference to `alloc_io_pgtable_ops'

    Add ARM || ARM64 || COMPILE_TEST dependencies to IPMMU_VMSA to prevent
    these issues, i.e., so that ARCH_RENESAS on RISC-V is not allowed.

    This makes the ARCH dependencies become:
            depends on (ARCH_RENESAS && (ARM || ARM64)) || COMPILE_TEST
    but that can be a bit hard to read.

    Fixes: 8292493c22c8 ("riscv: Kconfig.socs: Add ARCH_RENESAS kconfig option")
    Signed-off-by: Randy Dunlap <rdunlap@infradead.org>
    Suggested-by: Geert Uytterhoeven <geert@linux-m68k.org>
    Cc: Joerg Roedel <joro@8bytes.org>
    Cc: Will Deacon <will@kernel.org>
    Cc: Robin Murphy <robin.murphy@arm.com>
    Cc: iommu@lists.linux.dev
    Cc: Conor Dooley <conor@kernel.org>
    Cc: Lad Prabhakar <prabhakar.mahadev-lad.rj@bp.renesas.com>
    Reviewed-by: Robin Murphy <robin.murphy@arm.com>
    Link: https://lore.kernel.org/r/20230330165817.21920-1-rdunlap@infradead.org
    Signed-off-by: Joerg Roedel <jroedel@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 958ba6cf2758cc40058550c0c6175796fd22533f
Author: Kalesh AP <kalesh-anakkur.purayil@broadcom.com>
Date:   Thu May 18 01:11:00 2023 -0700

    RDMA/bnxt_re: Fix return value of bnxt_re_process_raw_qp_pkt_rx

    [ Upstream commit 0fa0d520e2a878cb4c94c4dc84395905d3f14f54 ]

    bnxt_re_process_raw_qp_pkt_rx() always return 0 and ignores the return
    value of bnxt_re_post_send_shadow_qp().
```

```
    Fixes: 1ac5a4047975 ("RDMA/bnxt_re: Add bnxt_re RoCE driver")
    Link: https://lore.kernel.org/r/1684397461-23082-3-git-send-email-
selvin.xavier@broadcom.com
    Reviewed-by: Hongguang Gao <hongguang.gao@broadcom.com>
    Reviewed-by: Ajit Khaparde <ajit.khaparde@broadcom.com>
    Signed-off-by: Kalesh AP <kalesh-anakkur.purayil@broadcom.com>
    Signed-off-by: Selvin Xavier <selvin.xavier@broadcom.com>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 34f6c54f4a9852d625b33276d11ad6761eeb3bfd
Author: Kalesh AP <kalesh-anakkur.purayil@broadcom.com>
Date:   Thu May 18 01:10:59 2023 -0700

    RDMA/bnxt_re: Fix a possible memory leak

    [ Upstream commit 349e3c0cf239cc01d58a1e6c749e171de014cd6a ]

    Inside bnxt_qplib_create_cq(), when the check for NULL DPI fails, driver
    returns directly without freeing the memory allocated inside
    bnxt_qplib_alloc_init_hwq() routine.

    Fixed this by moving the check for NULL DPI before invoking
    bnxt_qplib_alloc_init_hwq().

    Fixes: 1ac5a4047975 ("RDMA/bnxt_re: Add bnxt_re RoCE driver")
    Link: https://lore.kernel.org/r/1684397461-23082-2-git-send-email-
selvin.xavier@broadcom.com
    Reviewed-by: Kashyap Desai <kashyap.desai@broadcom.com>
    Signed-off-by: Kalesh AP <kalesh-anakkur.purayil@broadcom.com>
    Signed-off-by: Selvin Xavier <selvin.xavier@broadcom.com>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 43190a5c9855846942be9114a4028bf9e5fa783c
Author: Dan Carpenter <dan.carpenter@linaro.org>
Date:   Mon May 15 13:32:10 2023 +0300

    dmaengine: at_xdmac: fix potential Oops in at_xdmac_prep_interleaved()

    [ Upstream commit 4d43acb145c363626d76f49febb4240c488cd1cf ]

    There are two place if the at_xdmac_interleaved_queue_desc() fails which
    could lead to a NULL dereference where "first" is NULL and we call
    list_add_tail(&first->desc_node, ...).  In the first caller, the return
    is not checked so add a check for that.  In the next caller, the return
    is checked but if it fails on the first iteration through the loop then
    it will lead to a NULL pointer dereference.

    Fixes: 4e5385784e69 ("dmaengine: at_xdmac: handle numf > 1")
    Fixes: 62b5cb757f1d ("dmaengine: at_xdmac: fix memory leak in interleaved mode")
    Signed-off-by: Dan Carpenter <dan.carpenter@linaro.org>
    Reviewed-by: Tudor Ambarus <tudor.ambarus@linaro.org>
    Link: https://lore.kernel.org/r/21282b66-9860-410a-83df-39c17fcf2f1b@kili.mountain
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e7d50dc1a1f376dd93c769a33fd56f99dfd9eabd
Author: Yangyang Li <liyangyang20@huawei.com>
Date:   Fri May 12 17:22:45 2023 +0800

    RDMA/hns: Modify the value of long message loopback slice

    [ Upstream commit 56518a603fd2bf74762d176ac980572db84a3e14 ]
```

```
    Long message loopback slice is used for achieving traffic balance between
    QPs. It prevents the problem that QPs with large traffic occupying the
    hardware pipeline for a long time and QPs with small traffic cannot be
    scheduled.

    Currently, its maximum value is set to 16K, which means only after a QP
    sends 16K will the second QP be scheduled. This value is too large, which
    will lead to unbalanced traffic scheduling, and thus it needs to be
    modified.

    The setting range of the long message loopback slice is modified to be
    from 1024 (the lower limit supported by hardware) to mtu. Actual testing
    shows that this value can significantly reduce error in hardware traffic
    scheduling.

    This solution is compatible with both HIP08 and HIP09. The modified
    lp_pktn_ini has a maximum value of 2 (when mtu is 256), so the range
    checking code for lp_pktn_ini is no longer necessary and needs to be
    deleted.

    Fixes: 0e60778efb07 ("RDMA/hns: Modify the value of MAX_LP_MSG_LEN to meet hardware
compatibility")
    Link: https://lore.kernel.org/r/20230512092245.344442-4-huangjunxian6@hisilicon.com
    Signed-off-by: Yangyang Li <liyangyang20@huawei.com>
    Signed-off-by: Junxian Huang <huangjunxian6@hisilicon.com>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 941226203f0595f7dbab9c4e0ec73dbd64da83ea
Author: Chengchang Tang <tangchengchang@huawei.com>
Date:   Fri May 12 17:22:44 2023 +0800

    RDMA/hns: Fix base address table allocation

    [ Upstream commit 7f3969b14f356dd65fa95b3528eb05c32e68bc06 ]

    For hns, the specification of an entry like resource (E.g. WQE/CQE/EQE)
    depends on BT page size, buf page size and hopnum. For user mode, the buf
    page size depends on UMEM. Therefore, the actual specification is
    controlled by BT page size and hopnum.

    The current BT page size and hopnum are obtained from firmware. This makes
    the driver inflexible and introduces unnecessary constraints.  Resource
    allocation failures occur in many scenarios.

    This patch will calculate whether the BT page size set by firmware is
    sufficient before allocating BT, and increase the BT page size if it is
    insufficient.

    Fixes: 1133401412a9 ("RDMA/hns: Optimize base address table config flow for qp buffer")
    Link: https://lore.kernel.org/r/20230512092245.344442-3-huangjunxian6@hisilicon.com
    Signed-off-by: Chengchang Tang <tangchengchang@huawei.com>
    Signed-off-by: Junxian Huang <huangjunxian6@hisilicon.com>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4cf0362028ee83894965b673f4b383df25dbafd4
Author: Chengchang Tang <tangchengchang@huawei.com>
Date:   Fri May 12 17:22:43 2023 +0800

    RDMA/hns: Fix timeout attr in query qp for HIP08

    [ Upstream commit 58caa2a51ad4fd21763696cc6c4defc9fc1b4b4f ]

    On HIP08, the queried timeout attr is different from the timeout attr
```

```
    configured by the user.

    It is found by rdma-core testcase test_rdmacm_async_traffic:

    ======================================================================
    FAIL: test_rdmacm_async_traffic (tests.test_rdmacm.CMTestCase)
    ----------------------------------------------------------------------
    Traceback (most recent call last):
      File "./tests/test_rdmacm.py", line 33, in test_rdmacm_async_traffic
        self.two_nodes_rdmacm_traffic(CMAsyncConnection, self.rdmacm_traffic,
      File "./tests/base.py", line 382, in two_nodes_rdmacm_traffic
        raise(res)
    AssertionError

    Fixes: 926a01dc000d ("RDMA/hns: Add QP operations support for hip08 SoC")
    Link: https://lore.kernel.org/r/20230512092245.344442-2-huangjunxian6@hisilicon.com
    Signed-off-by: Chengchang Tang <tangchengchang@huawei.com>
    Signed-off-by: Junxian Huang <huangjunxian6@hisilicon.com>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b6be64ba6fbc2d9707df175b6e921686be7cdd66
Author: Yonatan Nachum <ynachum@amazon.com>
Date:   Thu May 11 11:51:03 2023 +0000

    RDMA/efa: Fix unsupported page sizes in device

    [ Upstream commit 866422cdddcdf59d8c68e9472d49ba1be29b5fcf ]

    Device uses 4KB size blocks for user pages indirect list while the
    driver creates those blocks with the size of PAGE_SIZE of the kernel. On
    kernels with PAGE_SIZE different than 4KB (ARM RHEL), this leads to a
    failure on register MR with indirect list because of the miss
    communication between driver and device.

    Fixes: 40909f664d27 ("RDMA/efa: Add EFA verbs implementation")
    Link: https://lore.kernel.org/r/20230511115103.13876-1-ynachum@amazon.com
    Reviewed-by: Firas Jahjah <firasj@amazon.com>
    Reviewed-by: Michael Margolin <mrgolin@amazon.com>
    Signed-off-by: Yonatan Nachum <ynachum@amazon.com>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d3eff2e9b8378b73a3e3b8794964b86db6fb0907
Author: Neil Armstrong <neil.armstrong@linaro.org>
Date:   Fri May 12 15:11:41 2023 +0200

    phy: amlogic: phy-meson-g12a-mipi-dphy-analog: fix CNTL2_DIF_TX_CTL0 value

    [ Upstream commit b949193011540bb17cf1da7795ec42af1b875203 ]

    Use the same CNTL2_DIF_TX_CTL0 value used by the vendor, it was reported
    fixing timings issues.

    Fixes: 2a56dc650e54 ("phy: amlogic: Add G12A Analog MIPI D-PHY driver")
    Signed-off-by: Neil Armstrong <neil.armstrong@linaro.org>
    Link: https://lore.kernel.org/r/20230512-amlogic-v6-4-upstream-dsi-ccf-vim3-v4-10-
2592c29ea263@linaro.org
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7618a00871b2cd11158db07f6ba0f1c90c55734c
Author: Selvin Xavier <selvin.xavier@broadcom.com>
Date:   Sun May 7 11:29:29 2023 -0700
```

```
    RDMA/bnxt_re: Fix the page_size used during the MR creation

    [ Upstream commit 08c7f09356e45d093d1867c7a3c6ac6526e2f98b ]

    Driver populates the list of pages used for Memory region wrongly when
    page size is more than system page size. This is causing a failure when
    some of the applications that creates MR with page size as 2M.  Since HW
    can support multiple page sizes, pass the correct page size while creating
    the MR.

    Also, driver need not adjust the number of pages when HW Queues are
    created with user memory. It should work with the number of dma blocks
    returned by ib_umem_num_dma_blocks. Fix this calculation also.

    Fixes: 0c4dcd602817 ("RDMA/bnxt_re: Refactor hardware queue memory allocation")
    Fixes: f6919d56388c ("RDMA/bnxt_re: Code refactor while populating user MRs")
    Link: https://lore.kernel.org/r/1683484169-9539-1-git-send-email-
selvin.xavier@broadcom.com
    Signed-off-by: Kalesh AP <kalesh-anakkur.purayil@broadcom.com>
    Signed-off-by: Kashyap Desai <kashyap.desai@broadcom.com>
    Signed-off-by: Selvin Xavier <selvin.xavier@broadcom.com>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```