```
commit 17f2d782f18c9a49943ea723d7628da1837c9204
Author: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
Date:   Wed May 24 17:30:25 2023 +0100

    Linux 6.3.4

    Link: https://lore.kernel.org/r/20230522190412.801391872@linuxfoundation.org
    Tested-by: Ronald Warsow <rwarsow@gmx.de>
    Tested-by: Florian Fainelli <f.fainelli@gmail.com>
    Tested-by: Shuah Khan <skhan@linuxfoundation.org>
    Tested-by: Bagas Sanjaya <bagasdotme@gmail.com>
    Tested-by: Ron Economos <re@w6rz.net>
    Tested-by: Jon Hunter <jonathanh@nvidia.com>
    Tested-by: Rudi Heitbaum <rudi@heitbaum.com>
    Tested-by: Justin M. Forbes <jforbes@fedoraproject.org>
    Tested-by: Conor Dooley <conor.dooley@microchip.com>
    Tested-by: Markus Reichelt <lkt+2023@mareichelt.com>
    Link: https://lore.kernel.org/r/20230523164950.435226211@linuxfoundation.org
    Tested-by: Chris Paterson (CIP) <chris.paterson2@renesas.com>
    Tested-by: Ronald Warsow <rwarsow@gmx.de>
    Tested-by: Markus Reichelt <lkt+2023@mareichelt.com>
    Tested-by: Linux Kernel Functional Testing <lkft@linaro.org>
    Tested-by: Rudi Heitbaum <rudi@heitbaum.com>
    Tested-by: Ron Economos <re@w6rz.net>
    Tested-by: Guenter Roeck <linux@roeck-us.net>
    Tested-by: Florian Fainelli <f.fainelli@gmail.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 35fe6fa57b994e7da222893adf0bb748d6055e73
Author: Wenchao Hao <haowenchao2@huawei.com>
Date:   Mon May 15 15:01:55 2023 +0800

    scsi: Revert "scsi: core: Do not increase scsi_device's iorequest_cnt if dispatch
failed"

    commit 6ca9818d1624e136a76ae8faedb6b6c95ca66903 upstream.

    The "atomic_inc(&cmd->device->iorequest_cnt)" in scsi_queue_rq() would
    cause kernel panic because cmd->device may be freed after returning from
    scsi_dispatch_cmd().

    This reverts commit cfee29ffb45b1c9798011b19d454637d1b0fe87d.

    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/20230515070156.1790181-2-haowenchao2@huawei.com
    Reviewed-by: Ming Lei <ming.lei@redhat.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 71355fa31e2081ce78e11f590189dfe630ddb40d
Author: Li Ma <li.ma@amd.com>
Date:   Wed Apr 12 22:06:34 2023 +0800

    drm/amdgpu: reserve the old gc_11_0_*_mes.bin

    commit 8855818ce7554fb7420200187fac9c3b69500da0 upstream.

    Reserve the MOUDLE_FIRMWARE declaration of gc_11_0_*_mes.bin
    to fix falling back to old mes bin on failure via autoload.

    Fixes: 97998b893c30 ("drm/amd/amdgpu: introduce gc_*_mes_2.bin v2")
    Signed-off-by: Li Ma <li.ma@amd.com>
```

```
    Reviewed-by: Yifan Zhang <yifan1.zhang@amd.com>
    Reviewed-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 7ef5aaa13cbe468ba44169299d171e06ce78f94d
Author: Jack Xiao <Jack.Xiao@amd.com>
Date:   Fri Mar 24 16:55:15 2023 +0800

    drm/amd/amdgpu: introduce gc_*_mes_2.bin v2

    commit 97998b893c3000b27a780a4982e16cfc8f4ea555 upstream.

    To avoid new mes fw running with old driver, rename
    mes schq fw to gc_*_mes_2.bin.

    v2: add MODULE_FIRMWARE declaration
    v3: squash in fixup patch

    Signed-off-by: Jack Xiao <Jack.Xiao@amd.com>
    Reviewed-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a293498f54b6d05191f7546f0cba10036d99620e
Author: Ard Biesheuvel <ardb@kernel.org>
Date:   Mon May 8 13:40:19 2023 +0100

    ARM: 9297/1: vfp: avoid unbalanced stack on 'success' return path

    [ Upstream commit 2b951b0efbaa6c805854b60c11f08811054d50cd ]

    Commit c76c6c4ecbec0deb5 ("ARM: 9294/2: vfp: Fix broken softirq handling
    with instrumentation enabled") updated the VFP exception entry logic to
    go via a C function, so that we get the compiler's version of
    local_bh_disable(), which may be instrumented, and isn't generally
    callable from assembler.

    However, this assumes that passing an alternative 'success' return
    address works in C as it does in asm, and this is only the case if the C
    calls in question are tail calls, as otherwise, the stack will need some
    unwinding as well.

    I have already sent patches to the list that replace most of the asm
    logic with C code, and so it is preferable to have a minimal fix that
    addresses the issue and can be backported along with the commit that it

    fixes to v6.3 from v6.4. Hopefully, we can land the C conversion for v6.5.

    So instead of passing the 'success' return address as a function
    argument, pass the stack address from where to pop it so that both LR
    and SP have the expected value.

    Fixes: c76c6c4ecbec0deb5 ("ARM: 9294/2: vfp: Fix broken softirq handling with ...")
    Reported-by: syzbot+d4b00edc2d0c910d4bf4@syzkaller.appspotmail.com
    Tested-by: syzbot+d4b00edc2d0c910d4bf4@syzkaller.appspotmail.com
    Reviewed-by: Linus Walleij <linus.walleij@linaro.org>
    Tested-by: Andrew Lunn <andrew@lunn.ch>
    Signed-off-by: Ard Biesheuvel <ardb@kernel.org>
    Tested-by: Andre Przywara <andre.przywara@arm.com>
    Signed-off-by: Russell King (Oracle) <rmk+kernel@armlinux.org.uk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit af88272c93c6aaaa000076f95c6b52498a169f10
Author: Ard Biesheuvel <ardb@kernel.org>
```

```
Date:   Wed Apr 12 09:50:20 2023 +0100

    ARM: 9294/2: vfp: Fix broken softirq handling with instrumentation enabled

    [ Upstream commit c76c6c4ecbec0deb56a4f9e932b26866024a508f ]

    Commit 62b95a7b44d1 ("ARM: 9282/1: vfp: Manipulate task VFP state with
    softirqs disabled") replaced the en/disable preemption calls inside the
    VFP state handling code with en/disabling of soft IRQs, which is
    necessary to allow kernel use of the VFP/SIMD unit when handling a soft
    IRQ.

    Unfortunately, when lockdep is enabled (or other instrumentation that
    enables TRACE_IRQFLAGS), the disable path implemented in asm fails to
    perform the lockdep and RCU related bookkeeping, resulting in spurious
    warnings and other badness.

    Set let's rework the VFP entry code a little bit so we can make the
    local_bh_disable() call from C, with all the instrumentations that
    happen to have been configured. Calling local_bh_enable() can be done
    from asm, as it is a simple wrapper around __local_bh_enable_ip(), which
    is always a callable function.

    Link: https://lore.kernel.org/all/ZBBYCSZUJOWBg1s8@localhost.localdomain/

    Fixes: 62b95a7b44d1 ("ARM: 9282/1: vfp: Manipulate task VFP state with softirqs
disabled")
    Signed-off-by: Ard Biesheuvel <ardb@kernel.org>
    Reviewed-by: Linus Walleij <linus.walleij@linaro.org>
    Tested-by: Guenter Roeck <linux@roeck-us.net>
    Signed-off-by: Russell King (Oracle) <rmk+kernel@armlinux.org.uk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d51c013916106b4d8ff3572129a34124a5d5115d
Author: Mathieu Poirier <mathieu.poirier@linaro.org>
Date:   Fri Apr 7 10:14:29 2023 -0600

    remoteproc: imx_dsp_rproc: Fix kernel test robot sparse warning

    [ Upstream commit 3c497f624d40171ebead1a6705793100d92ecb85 ]

    This patch fixes the kernel test robot warning reported here:

    https://lore.kernel.org/bpf/642f916b.pPIKZ%2Fl%2F%2Fbw8tvIH%25lkp@intel.com/T/

    Fixes: 408ec1ff0caa ("remoteproc: imx_dsp_rproc: Add custom memory copy implementation
for i.MX DSP Cores")
    Link: https://lore.kernel.org/r/20230407161429.3973177-1-mathieu.poirier@linaro.org
    Tested-by: Iuliana Prodan <iuliana.prodan@nxp.com>
    Reviewed-by: Iuliana Prodan <iuliana.prodan@nxp.com>
    Signed-off-by: Mathieu Poirier <mathieu.poirier@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b26d6872d2d6189c6d28555e3bc18d855cb2fddd
Author: Ze Gao <zegao2021@gmail.com>
Date:   Wed May 17 11:45:09 2023 +0800

    rethook, fprobe: do not trace rethook related functions

    commit 571a2a50a8fc546145ffd3bf673547e9fe128ed2 upstream.

    These functions are already marked as NOKPROBE to prevent recursion and
    we have the same reason to blacklist them if rethook is used with fprobe,
    since they are beyond the recursion-free region ftrace can guard.
```

```
    Link: https://lore.kernel.org/all/20230517034510.15639-5-zegao@tencent.com/

    Fixes: f3a112c0c40d ("x86,rethook,kprobes: Replace kretprobe with rethook on x86")
    Signed-off-by: Ze Gao <zegao@tencent.com>
    Reviewed-by: Steven Rostedt (Google) <rostedt@goodmis.org>
    Acked-by: Masami Hiramatsu (Google) <mhiramat@kernel.org>
    Cc: stable@vger.kernel.org
    Signed-off-by: Masami Hiramatsu (Google) <mhiramat@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 71db1742cd20ff03a574df98058d511f9ed71e8a
Author: Ze Gao <zegao2021@gmail.com>
Date:   Wed May 17 11:45:06 2023 +0800

    rethook: use preempt_{disable, enable}_notrace in rethook_trampoline_handler

    commit be243bacfb25f5219f2396d787408e8cf1301dd1 upstream.

    This patch replaces preempt_{disable, enable} with its corresponding
    notrace version in rethook_trampoline_handler so no worries about stack
    recursion or overflow introduced by preempt_count_{add, sub} under
    fprobe + rethook context.

    Link: https://lore.kernel.org/all/20230517034510.15639-2-zegao@tencent.com/

    Fixes: 54ecbe6f1ed5 ("rethook: Add a generic return hook")
    Signed-off-by: Ze Gao <zegao@tencent.com>
    Acked-by: Masami Hiramatsu (Google) <mhiramat@kernel.org>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Masami Hiramatsu (Google) <mhiramat@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1426cf02abaa3ecf8ee68df09d3ef96aeeb4116d
Author: Peter Collingbourne <pcc@google.com>
Date:   Thu Apr 20 14:43:27 2023 -0700

    arm64: mte: Do not set PG_mte_tagged if tags were not initialized

    commit c4c597f1b367433c52c531dccd6859a39b4580fb upstream.

    The mte_sync_page_tags() function sets PG_mte_tagged if it initializes
    page tags. Then we return to mte_sync_tags(), which sets PG_mte_tagged
    again. At best, this is redundant. However, it is possible for
    mte_sync_page_tags() to return without having initialized tags for the
    page, i.e. in the case where check_swap is true (non-compound page),
    is_swap_pte(old_pte) is false and pte_is_tagged is false. So at worst,
    we set PG_mte_tagged on a page with uninitialized tags. This can happen
    if, for example, page migration causes a PTE for an untagged page to
    be replaced. If the userspace program subsequently uses mprotect() to
    enable PROT_MTE for that page, the uninitialized tags will be exposed
    to userspace.

    Fix it by removing the redundant call to set_page_mte_tagged().

    Fixes: e059853d14ca ("arm64: mte: Fix/clarify the PG_mte_tagged semantics")
    Signed-off-by: Peter Collingbourne <pcc@google.com>
    Cc: <stable@vger.kernel.org> # 6.1
    Link: https://linux-review.googlesource.com/id/Ib02d004d435b2ed87603b858ef7480f7b1463052
    Reviewed-by: Catalin Marinas <catalin.marinas@arm.com>
    Reviewed-by: Alexandru Elisei <alexandru.elisei@arm.com>
    Link: https://lore.kernel.org/r/20230420214327.2357985-1-pcc@google.com
    Signed-off-by: Will Deacon <will@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 57a6da9cc6a3bddf22e05e40d51477a7d78541dd
```

```
Author: Peter Collingbourne <pcc@google.com>
Date:   Thu Apr 20 14:09:45 2023 -0700

    arm64: Also reset KASAN tag if page is not PG_mte_tagged

    commit 2efbafb91e12ff5a16cbafb0085e4c10c3fca493 upstream.

    Consider the following sequence of events:

    1) A page in a PROT_READ|PROT_WRITE VMA is faulted.
    2) Page migration allocates a page with the KASAN allocator,
       causing it to receive a non-match-all tag, and uses it
       to replace the page faulted in 1.
    3) The program uses mprotect() to enable PROT_MTE on the page faulted in 1.

    As a result of step 3, we are left with a non-match-all tag for a page
    with tags accessible to userspace, which can lead to the same kind of
    tag check faults that commit e74a68468062 ("arm64: Reset KASAN tag in
    copy_highpage with HW tags only") intended to fix.

    The general invariant that we have for pages in a VMA with VM_MTE_ALLOWED
    is that they cannot have a non-match-all tag. As a result of step 2, the
    invariant is broken. This means that the fix in the referenced commit
    was incomplete and we also need to reset the tag for pages without
    PG_mte_tagged.

    Fixes: e5b8d9218951 ("arm64: mte: reset the page tag in page->flags")
    Cc: <stable@vger.kernel.org> # 5.15
    Link: https://linux-review.googlesource.com/id/I7409cdd41acbcb215c2a7417c1e50d37b875beff
    Signed-off-by: Peter Collingbourne <pcc@google.com>
    Reviewed-by: Catalin Marinas <catalin.marinas@arm.com>
    Link: https://lore.kernel.org/r/20230420210945.2313627-1-pcc@google.com
    Signed-off-by: Will Deacon <will@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 7f0dc8f975067943b382a38d86bd8ff4ffe2bef1
Author: Heiko Carstens <hca@linux.ibm.com>
Date:   Thu May 11 17:04:41 2023 +0200

    s390/qdio: fix do_sqbs() inline assembly constraint

    commit 2862a2fdfae875888e3c1c3634e3422e01d98147 upstream.

    Use "a" constraint instead of "d" constraint to pass the state parameter to
    the do_sqbs() inline assembly. This prevents that general purpose register
    zero is used for the state parameter.

    If the compiler would select general purpose register zero this would be
    problematic for the used instruction in rsy format: the register used for
    the state parameter is a base register. If the base register is general
    purpose register zero the contents of the register are unexpectedly ignored
    when the instruction is executed.

    This only applies to z/VM guests using QIOASSIST with dedicated (pass through)
    QDIO-based devices such as FCP [zfcp driver] as well as real OSA or
    HiperSockets [qeth driver].

    A possible symptom for this case using zfcp is the following repeating kernel
    message pattern:

    zfcp <devbusid>: A QDIO problem occurred
    zfcp <devbusid>: A QDIO problem occurred
    zfcp <devbusid>: qdio: ZFCP on SC <sc> using AI:1 QEBSM:1 PRI:1 TDD:1 SIGA: W
    zfcp <devbusid>: A QDIO problem occurred
    zfcp <devbusid>: A QDIO problem occurred
```

```
    Each of the qdio problem message can be accompanied by the following entries
    for the affected subchannel <sc> in
    /sys/kernel/debug/s390dbf/qdio_error/hex_ascii for zfcp or qeth:

    <sc> ccq: 69....
    <sc> SQBS ERROR.

    Reviewed-by: Benjamin Block <bblock@linux.ibm.com>
    Cc: Steffen Maier <maier@linux.ibm.com>
    Fixes: 8129ee164267 ("[PATCH] s390: qdio V=V pass-through")
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Heiko Carstens <hca@linux.ibm.com>
    Signed-off-by: Alexander Gordeev <agordeev@linux.ibm.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit debb7797bba0caffdbdadc3e7968bb2c414f50da
Author: Heiko Carstens <hca@linux.ibm.com>
Date:   Thu Apr 20 13:31:29 2023 +0200

    s390/crypto: use vector instructions only if available for ChaCha20

    commit 8703dd6b238da0ec6c276e53836f8200983d3d9b upstream.

    Commit 349d03ffd5f6 ("crypto: s390 - add crypto library interface for
    ChaCha20") added a library interface to the s390 specific ChaCha20
    implementation. However no check was added to verify if the required
    facilities are installed before branching into the assembler code.

    If compiled into the kernel, this will lead to the following crash,
    if vector instructions are not available:

    data exception: 0007 ilc:3 [#1] SMP
    Modules linked in:
    CPU: 0 PID: 1 Comm: swapper/0 Not tainted 6.3.0-rc7+ #11
    Hardware name: IBM 3931 A01 704 (KVM/Linux)
    Krnl PSW : 0704e00180000000 000000001857277a (chacha20_vx+0x32/0x818)
               R:0 T:1 IO:1 EX:1 Key:0 M:1 W:0 P:0 AS:3 CC:2 PM:0 RI:0 EA:3
    Krnl GPRS: 0000037f0000000a fffffffffffffff60 000000008184b000 0000000019f5c8e6
               0000000000000109 0000037fffb13c58 0000037fffb13c78 0000000019bb1780
               0000037fffb13c58 0000000019f5c8e6 000000008184b000 0000000000000109
               00000000802d8000 0000000000000109 0000000018571ebc 0000037fffb13718
    Krnl Code: 000000001857276a: c07000b1f80b        larl    %r7,0000000019bb1780
               0000000018572770: a708000a            lhi     %r0,10
              #0000000018572774: e78950000c36        vlm     %v24,%v25,0(%r5),0
              >000000001857277a: e7a060000806        vl      %v26,0(%r6),0
               0000000018572780: e7bf70004c36        vlm     %v27,%v31,0(%r7),4
               0000000018572786: e70b00000456        vlr     %v0,%v27
               000000001857278c: e71800000456        vlr     %v1,%v24
               0000000018572792: e74b00000456        vlr     %v4,%v27
    Call Trace:
     [<000000001857277a>] chacha20_vx+0x32/0x818
    Last Breaking-Event-Address:
     [<0000000018571eb6>] chacha20_crypt_s390.constprop.0+0x6e/0xd8
    ---[ end trace 0000000000000000 ]---
    Kernel panic - not syncing: Attempted to kill init! exitcode=0x0000000b

    Fix this by adding a missing MACHINE_HAS_VX check.

    Fixes: 349d03ffd5f6 ("crypto: s390 - add crypto library interface for ChaCha20")
    Reported-by: Marc Hartmayer <mhartmay@linux.ibm.com>
    Cc: <stable@vger.kernel.org> # 5.19+
    Reviewed-by: Harald Freudenberger <freude@linux.ibm.com>
    [agordeev@linux.ibm.com: remove duplicates in commit message]
    Signed-off-by: Heiko Carstens <hca@linux.ibm.com>
```

```
    Signed-off-by: Alexander Gordeev <agordeev@linux.ibm.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 808e0ebf7ec2c4cd63fb14264119d26f62a94ea8
Author: Stefan Haberland <sth@linux.ibm.com>
Date:   Fri May 19 12:23:40 2023 +0200

    s390/dasd: fix command reject error on ESE devices

    commit c99bff34290f1b994073557b754aff86e4c7b22e upstream.

    Formatting a thin-provisioned (ESE) device that is part of a PPRC copy
    relation might fail with the following error:

    dasd-eckd 0.0.f500: An error occurred in the DASD device driver, reason=09
    [...]
    24 Byte: 0 MSG 4, no MSGb to SYSOP

    During format of an ESE disk the Release Allocated Space command is used.
    A bit in the payload of the command is set that is not allowed to be set
    for devices in a copy relation. This bit is set to allow the partial
    release of an extent.

    Check for the existence of a copy relation before setting the respective
    bit.

    Fixes: 91dc4a197569 ("s390/dasd: Add new ioctl to release space")
    Cc: stable@kernel.org # 5.3+
    Signed-off-by: Stefan Haberland <sth@linux.ibm.com>
    Reviewed-by: Jan Hoeppner <hoeppner@linux.ibm.com>
    Link: https://lore.kernel.org/r/20230519102340.3854819-2-sth@linux.ibm.com
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit fb8e8d58f116d069e5939e1f786ac84e7fa4533e
Author: Ryusuke Konishi <konishi.ryusuke@gmail.com>
Date:   Wed May 10 00:29:56 2023 +0900

    nilfs2: fix use-after-free bug of nilfs_root in nilfs_evict_inode()

    commit 9b5a04ac3ad9898c4745cba46ea26de74ba56a8e upstream.

    During unmount process of nilfs2, nothing holds nilfs_root structure after
    nilfs2 detaches its writer in nilfs_detach_log_writer().  However, since
    nilfs_evict_inode() uses nilfs_root for some cleanup operations, it may
    cause use-after-free read if inodes are left in "garbage_list" and
    released by nilfs_dispose_list() at the end of nilfs_detach_log_writer().

    Fix this issue by modifying nilfs_evict_inode() to only clear inode
    without additional metadata changes that use nilfs_root if the file system
    is degraded to read-only or the writer is detached.

    Link: https://lkml.kernel.org/r/20230509152956.8313-1-konishi.ryusuke@gmail.com
    Signed-off-by: Ryusuke Konishi <konishi.ryusuke@gmail.com>
    Reported-by: syzbot+78d4495558999f55d1da@syzkaller.appspotmail.com
    Closes: https://lkml.kernel.org/r/00000000000099e5ac05fb1c3b85@google.com
    Tested-by: Ryusuke Konishi <konishi.ryusuke@gmail.com>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e1a767ae657fdd0502ae07f0fdca1a38a81d2290
Author: Huayu Chen <huayu.chen@corigine.com>
Date:   Thu May 11 08:50:56 2023 +0200
```

```
    nfp: fix NFP_NET_MAX_DSCP definition error

    commit de9c1a23add9e7842ce63ce6f498a05c66344311 upstream.

    The patch corrects the NFP_NET_MAX_DSCP definition in the main.h file.

    The incorrect definition result DSCP bits not being mapped properly when
    DCB is set. When NFP_NET_MAX_DSCP was defined as 4, the next 60 DSCP
    bits failed to be set.

    Fixes: 9b7fe8046d74 ("nfp: add DCB IEEE support")
    Cc: stable@vger.kernel.org
    Signed-off-by: Huayu Chen <huayu.chen@corigine.com>
    Acked-by: Simon Horman <simon.horman@corigine.com>
    Signed-off-by: Louis Peens <louis.peens@corigine.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit bc4f671161b6c3535b4b9e6fb6e408b064fb4f0f
Author: Hari Bathini <hbathini@linux.ibm.com>
Date:   Tue Apr 25 12:28:29 2023 +0530

    powerpc/bpf: populate extable entries only during the last pass

    commit 35a4b8ce4ac00e940b46b1034916ccb22ce9bdef upstream.

    Since commit 85e031154c7c ("powerpc/bpf: Perform complete extra passes
    to update addresses"), two additional passes are performed to avoid
    space and CPU time wastage on powerpc. But these extra passes led to
    WARN_ON_ONCE() hits in bpf_add_extable_entry() as extable entries are
    populated again, during the extra pass, without resetting the index.
    Fix it by resetting entry index before repopulating extable entries,
    if and when there is an additional pass.

    Fixes: 85e031154c7c ("powerpc/bpf: Perform complete extra passes to update addresses")
    Cc: stable@vger.kernel.org # v6.3+
    Signed-off-by: Hari Bathini <hbathini@linux.ibm.com>
    Reviewed-by: Naveen N. Rao <naveen.n.rao@linux.vnet.ibm.com>
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Link: https://msgid.link/20230425065829.18189-1-hbathini@linux.ibm.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 4e913882a3719e7b56858abad8e79e73d346f46e
Author: Michael Ellerman <mpe@ellerman.id.au>
Date:   Thu May 11 21:42:24 2023 +1000

    powerpc/64s/radix: Fix soft dirty tracking

    commit 66b2ca086210732954a7790d63d35542936fc664 upstream.

    It was reported that soft dirty tracking doesn't work when using the
    Radix MMU.

    The tracking is supposed to work by clearing the soft dirty bit for a
    mapping and then write protecting the PTE. If/when the page is written
    to, a page fault occurs and the soft dirty bit is added back via
    pte_mkdirty(). For example in wp_page_reuse():

            entry = maybe_mkwrite(pte_mkdirty(entry), vma);
            if (ptep_set_access_flags(vma, vmf->address, vmf->pte, entry, 1))
                    update_mmu_cache(vma, vmf->address, vmf->pte);

    Unfortunately on radix _PAGE_SOFTDIRTY is being dropped by
    radix__ptep_set_access_flags(), called from ptep_set_access_flags(),
    meaning the soft dirty bit is not set even though the page has been
```

```
    written to.

    Fix it by adding _PAGE_SOFTDIRTY to the set of bits that are able to be
    changed in radix__ptep_set_access_flags().

    Fixes: b0b5e9b13047 ("powerpc/mm/radix: Add radix pte #defines")
    Cc: stable@vger.kernel.org # v4.7+
    Reported-by: Dan Horák <dan@danny.cz>
    Link: https://lore.kernel.org/r/20230511095558.56663a50f86bdc4cd97700b7@danny.cz
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Link: https://msgid.link/20230511114224.977423-1-mpe@ellerman.id.au
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 29b3242cef4e06fa354a2ce0f2ecadcba0576b46
Author: Jerry Snitselaar <jsnitsel@redhat.com>
Date:   Wed May 10 17:54:03 2023 -0700

    tpm/tpm_tis: Disable interrupts for more Lenovo devices

    commit e7d3e5c4b1dd50a70b31524c3228c62bb41bbab2 upstream.

    The P360 Tiny suffers from an irq storm issue like the T490s, so add
    an entry for it to tpm_tis_dmi_table, and force polling. There also
    previously was a report from the previous attempt to enable interrupts
    that involved a ThinkPad L490. So an entry is added for it as well.

    Cc: stable@vger.kernel.org
    Reported-by: Peter Zijlstra <peterz@infradead.org> # P360 Tiny
    Closes: https://lore.kernel.org/linux-
integrity/20230505130731.GO83892@hirez.programming.kicks-ass.net/
    Signed-off-by: Jerry Snitselaar <jsnitsel@redhat.com>
    Signed-off-by: Jarkko Sakkinen <jarkko@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 36d0d1a0dd9cc2c3fa4c55366b77033456bef565
Author: Gaurav Batra <gbatra@linux.vnet.ibm.com>
Date:   Fri May 5 13:47:01 2023 -0500

    powerpc/iommu: Incorrect DDW Table is referenced for SR-IOV device

    commit 1f7aacc5eb9ed2cc17be7a90da5cd559effb9d59 upstream.

    For an SR-IOV device, while enabling DDW, a new table is created and
    added at index 1 in the group. In the below 2 scenarios, the table is
    incorrectly referenced at index 0 (which is where the table is for
    default DMA window).

    1. When adding DDW

       This issue is exposed with "slub_debug". Error thrown out from
       dma_iommu_dma_supported()

       Warning: IOMMU offset too big for device mask
       mask: 0xffffffff, table offset: 0x800000000000000

    2. During Dynamic removal of the PCI device.

       Error is from iommu_tce_table_put() since a NULL table pointer is
       passed in.

    Fixes: 381ceda88c4c ("powerpc/pseries/iommu: Make use of DDW for indirect mapping")
    Cc: stable@vger.kernel.org # v5.15+
    Signed-off-by: Gaurav Batra <gbatra@linux.vnet.ibm.com>
    Reviewed-by: Brian King <brking@linux.vnet.ibm.com>
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
```

```
    Link: https://msgid.link/20230505184701.91613-1-gbatra@linux.vnet.ibm.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 4da40194ac904429e93060c98970e5815c29dd0d
Author: Gaurav Batra <gbatra@linux.vnet.ibm.com>
Date:   Thu May 4 12:59:13 2023 -0500

    powerpc/iommu: DMA address offset is incorrectly calculated with 2MB TCEs

    commit 096339ab84f36beae0b1db25e0ce63fb3873e8b2 upstream.

    When DMA window is backed by 2MB TCEs, the DMA address for the mapped
    page should be the offset of the page relative to the 2MB TCE. The code
    was incorrectly setting the DMA address to the beginning of the TCE
    range.

    Mellanox driver is reporting timeout trying to ENABLE_HCA for an SR-IOV
    ethernet port, when DMA window is backed by 2MB TCEs.

    Fixes: 387273118714 ("powerps/pseries/dma: Add support for 2M IOMMU page size")
    Cc: stable@vger.kernel.org # v5.16+
    Signed-off-by: Gaurav Batra <gbatra@linux.vnet.ibm.com>
    Reviewed-by: Greg Joyce <gjoyce@linux.vnet.ibm.com>
    Reviewed-by: Brian King <brking@linux.vnet.ibm.com>
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Link: https://msgid.link/20230504175913.83844-1-gbatra@linux.vnet.ibm.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 0b3524c16d1333f3b30888dae1be48fb74e229d0
Author: Michal Simek <michal.simek@amd.com>
Date:   Fri May 12 13:52:04 2023 +0200

    dt-bindings: ata: ahci-ceva: Cover all 4 iommus entries

    commit a7844528722619d2f97740ae5ec747afff18c4be upstream.

    Current only one entry is enabled but IP itself is using 4 different IDs
    which are already listed in zynqmp.dtsi.

    sata: ahci@fd0c0000 {
            compatible = "ceva,ahci-1v84";
            ...
            iommus = <&smmu 0x4c0>, <&smmu 0x4c1>,
                     <&smmu 0x4c2>, <&smmu 0x4c3>;
    };

    Fixes: 8ac47837f0e0 ("arm64: dts: zynqmp: Add missing iommu IDs")
    Cc: stable@vger.kernel.org # v5.12+
    Signed-off-by: Michal Simek <michal.simek@amd.com>
    Acked-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Signed-off-by: Damien Le Moal <dlemoal@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 2125d41ae4742d628f91d82555a6a966d6d29454
Author: Rob Clark <robdclark@chromium.org>
Date:   Tue May 16 15:20:36 2023 -0700

    iommu/arm-smmu-qcom: Fix missing adreno_smmu's

    commit e36ca2fad6bb4ef0603bdb5556578e9082fe0056 upstream.

    When the special handling of qcom,adreno-smmu was moved into
    qcom_smmu_create(), it was overlooked that we didn't have all the
    required entries in qcom_smmu_impl_of_match.  So we stopped getting
    adreno_smmu_priv on sc7180, breaking per-process pgtables.
```

```
    Fixes: 30b912a03d91 ("iommu/arm-smmu-qcom: Move the qcom,adreno-smmu check into
qcom_smmu_create")
    Cc: <stable@vger.kernel.org>
    Suggested-by: Lepton Wu <lepton@chromium.org>
    Signed-off-by: Rob Clark <robdclark@chromium.org>
    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Reviewed-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Patchwork: https://patchwork.freedesktop.org/patch/537357/
    Link: https://lore.kernel.org/r/20230516222039.907690-1-robdclark@gmail.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit fc332a4d3794410d4f9f8d68ed30fb361d809638
Author: Alex Deucher <alexander.deucher@amd.com>
Date:   Mon Apr 10 12:02:29 2023 -0400

    drm/amdgpu/gfx11: update gpu_clock_counter logic

    commit d5aa417808cf14c052ca042920b3c6b9f1dc6aa4 upstream.

    This code was written prior to previous updates to this
    logic for other chips.  The RSC registers are part of
    SMUIO which is an always on block so there is no need
    to disable gfxoff.  Additionally add the carryover and
    preemption checks.

    v2: rebase

    Reviewed-by: Hawking Zhang <Hawking.Zhang@amd.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Cc: stable@vger.kernel.org # 6.1.y: 5591a051b86b: drm/amdgpu: refine get gpu clock
counter method
    Cc: stable@vger.kernel.org # 6.2.y: 5591a051b86b: drm/amdgpu: refine get gpu clock
counter method
    Cc: stable@vger.kernel.org # 6.3.y: 5591a051b86b: drm/amdgpu: refine get gpu clock
counter method
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 5af10fd3cb68798ae4b1209e897ff6e983dfb30b
Author: Tong Liu01 <Tong.Liu01@amd.com>
Date:   Thu Apr 6 15:58:31 2023 +0800

    drm/amdgpu: refine get gpu clock counter method

    commit 5591a051b86be170a84943698ab140342602ff7b upstream.

    [why]
    regGOLDEN_TSC_COUNT_LOWER/regGOLDEN_TSC_COUNT_UPPER are protected and
    unaccessible under sriov.
    The clock counter high bit may update during reading process.

    [How]
    Replace regGOLDEN_TSC_COUNT_LOWER/regGOLDEN_TSC_COUNT_UPPER with
    regCP_MES_MTIME_LO/regCP_MES_MTIME_HI to get gpu clock under sriov.
    Refine get gpu clock counter method to make the result more precise.

    Signed-off-by: Tong Liu01 <Tong.Liu01@amd.com>
    Acked-by: Luben Tuikov <luben.tuikov@amd.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c5b44c03d4ae2c1e33f09743f1e727429779d767
Author: Guilherme G. Piccoli <gpiccoli@igalia.com>
Date:   Tue May 9 18:49:47 2023 +0200
```

```
      drm/amdgpu/gfx11: Adjust gfxoff before powergating on gfx11 as well

      commit 11fbdda2ab6bf049e2869139c07016022b4e045b upstream.

      (Bas: speculative change to mirror gfx10/gfx9)

      Signed-off-by: Guilherme G. Piccoli <gpiccoli@igalia.com>
      Signed-off-by: Bas Nieuwenhuizen <bas@basnieuwenhuizen.nl>
      Cc: 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 4177bdc99455eb741c97ab8f94dab8009d6293ed
  Author: Bas Nieuwenhuizen <bas@basnieuwenhuizen.nl>
  Date:   Tue May 9 18:49:46 2023 +0200

      drm/amdgpu/gfx10: Disable gfxoff before disabling powergating.

      commit 8173cab3368a13cdc3cad0bd5cf14e9399b0f501 upstream.

      Otherwise we get a full system lock (looks like a FW mess).

      Copied the order from the GFX9 powergating code.

      Fixes: 366468ff6c34 ("drm/amdgpu: Allow GfxOff on Vangogh as default")
      Link: https://gitlab.freedesktop.org/drm/amd/-/issues/2545
      Signed-off-by: Bas Nieuwenhuizen <bas@basnieuwenhuizen.nl>
      Tested-by: Guilherme G. Piccoli <gpiccoli@igalia.com>
      Cc: 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 ce7155379296f31d61c98b75d4435fb556f56c56
  Author: Alex Deucher <alexander.deucher@amd.com>
  Date:   Thu May 11 10:40:03 2023 -0400

      drm/amdgpu/gmc11: implement get_vbios_fb_size()

      commit 68518294d00da6a2433357af75a63abc6030676e upstream.

      Implement get_vbios_fb_size() so we can properly reserve
      the vbios splash screen to avoid potential artifacts on the
      screen during the transition from the pre-OS console to the
      OS console.

      Acked-by: Sunil Khatri <sunil.khatri@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 52b7b69a6602f239c0dc9d584fe265f904f58d97
  Author: Evan Quan <evan.quan@amd.com>
  Date:   Thu May 11 15:41:27 2023 +0800

      drm/amd/pm: fix possible power mode mismatch between driver and PMFW

      commit bf4823267a817f7c155876a125b94336d7113e77 upstream.

      PMFW may boots the ASIC with a different power mode from the system's
      real one. Notify PMFW explicitly the power mode the system in. This
      is needed only when ACDC switch via gpio is not supported.

      Signed-off-by: Evan Quan <evan.quan@amd.com>
```

```
    Reviewed-by: Kenneth Feng <kenneth.feng@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 a6dc9e620970d712c5728f060915078ca32143ac
Author: Xiubo Li <xiubli@redhat.com>
Date:   Thu May 18 09:47:23 2023 +0800

    ceph: force updating the msg pointer in non-split case

    commit 4cafd0400bcb6187c0d4ab4d4b0229a89ac4f8c2 upstream.

    When the MClientSnap reqeust's op is not CEPH_SNAP_OP_SPLIT the
    request may still contain a list of 'split_realms', and we need
    to skip it anyway. Or it will be parsed as a corrupt snaptrace.

    Cc: stable@vger.kernel.org
    Link: https://tracker.ceph.com/issues/61200
    Reported-by: Frank Schilder <frans@dtu.dk>
    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: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1de42e7653d6714a7507ba6696151a1fa028c69f
Author: George Kennedy <george.kennedy@oracle.com>
Date:   Fri May 12 06:08:48 2023 -0500

    vc_screen: reload load of struct vc_data pointer in vcs_write() to avoid UAF

    commit 8fb9ea65c9d1338b0d2bb0a9122dc942cdd32357 upstream.

    After a call to console_unlock() in vcs_write() the vc_data struct can be
    freed by vc_port_destruct(). Because of that, the struct vc_data pointer
    must be reloaded in the while loop in vcs_write() after console_lock() to
    avoid a UAF when vcs_size() is called.

    Syzkaller reported a UAF in vcs_size().

    BUG: KASAN: slab-use-after-free in vcs_size (drivers/tty/vt/vc_screen.c:215)
    Read of size 4 at addr ffff8880beab89a8 by task repro_vcs_size/4119

    Call Trace:
     <TASK>
    __asan_report_load4_noabort (mm/kasan/report_generic.c:380)
    vcs_size (drivers/tty/vt/vc_screen.c:215)
    vcs_write (drivers/tty/vt/vc_screen.c:664)
    vfs_write (fs/read_write.c:582 fs/read_write.c:564)
    ...
     <TASK>

    Allocated by task 1213:
    kmalloc_trace (mm/slab_common.c:1064)
    vc_allocate (./include/linux/slab.h:559 ./include/linux/slab.h:680
        drivers/tty/vt/vt.c:1078 drivers/tty/vt/vt.c:1058)
    con_install (drivers/tty/vt/vt.c:3334)
    tty_init_dev (drivers/tty/tty_io.c:1303 drivers/tty/tty_io.c:1415
        drivers/tty/tty_io.c:1392)
    tty_open (drivers/tty/tty_io.c:2082 drivers/tty/tty_io.c:2128)
    chrdev_open (fs/char_dev.c:415)
    do_dentry_open (fs/open.c:921)
    vfs_open (fs/open.c:1052)
    ...
```

```
    Freed by task 4116:
    kfree (mm/slab_common.c:1016)
    vc_port_destruct (drivers/tty/vt/vt.c:1044)
    tty_port_destructor (drivers/tty/tty_port.c:296)
    tty_port_put (drivers/tty/tty_port.c:312)
    vt_disallocate_all (drivers/tty/vt/vt_ioctl.c:662 (discriminator 2))
    vt_ioctl (drivers/tty/vt/vt_ioctl.c:903)
    tty_ioctl (drivers/tty/tty_io.c:2778)
    ...

    The buggy address belongs to the object at ffff8880beab8800
     which belongs to the cache kmalloc-1k of size 1024
    The buggy address is located 424 bytes inside of
     freed 1024-byte region [ffff8880beab8800, ffff8880beab8c00)

    The buggy address belongs to the physical page:
    page:00000000afc77580 refcount:1 mapcount:0 mapping:0000000000000000
        index:0x0 pfn:0xbeab8
    head:00000000afc77580 order:3 entire_mapcount:0 nr_pages_mapped:0
        pincount:0
    flags: 0xfffffc0010200(slab|head|node=0|zone=1|lastcpupid=0x1fffff)
    page_type: 0xffffffff()
    raw: 000fffffc0010200 ffff888100042dc0 ffffea000426de00 dead000000000002
    raw: 0000000000000000 0000000000100010 00000001ffffffff 0000000000000000
    page dumped because: kasan: bad access detected

    Memory state around the buggy address:
     ffff8880beab8880: fb fb fb fb fb fb fb fb fb fb fb fb fb fb fb fb
     ffff8880beab8900: fb fb fb fb fb fb fb fb fb fb fb fb fb fb fb fb
    >ffff8880beab8980: fb fb fb fb fb fb fb fb fb fb fb fb fb fb fb fb
                                  ^
     ffff8880beab8a00: fb fb fb fb fb fb fb fb fb fb fb fb fb fb fb fb
     ffff8880beab8a80: fb fb fb fb fb fb fb fb fb fb fb fb fb fb fb fb
    ==================================================================
    Disabling lock debugging due to kernel taint

    Fixes: ac751efa6a0d ("console: rename acquire/release_console_sem() to
 console_lock/unlock()")
    Cc: stable <stable@kernel.org>
    Reported-by: syzkaller <syzkaller@googlegroups.com>
    Signed-off-by: George Kennedy <george.kennedy@oracle.com>
    Reviewed-by: Thomas Weißschuh <linux@weissschuh.net>
    Link: https://lore.kernel.org/r/1683889728-10411-1-git-send-email-
 george.kennedy@oracle.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

 commit 27b98a67a8ab21de9ae0f6a9553af9aa6aa2b43a
 Author: Mario Limonciello <mario.limonciello@amd.com>
 Date:   Mon Apr 24 14:55:54 2023 -0500

    thunderbolt: Clear registers properly when auto clear isn't in use

    commit c4af8e3fecd03b0aedcd38145955605cfebe7e3a upstream.

    When `QUIRK_AUTO_CLEAR_INT` isn't set, interrupt masking should be
    cleared by writing to Interrupt Mask Clear (IMR) and interrupt
    status should be cleared properly at shutdown/init.

    This fixes an error where interrupts are left enabled during resume
    from hibernation with `CONFIG_USB4=y`.

    Fixes: 468c49f44759 ("thunderbolt: Disable interrupt auto clear for rings")
    Cc: stable@vger.kernel.org # v6.3
    Reported-by: Takashi Iwai <tiwai@suse.de>
    Link: https://bugzilla.kernel.org/show_bug.cgi?id=217343
```

```
    Signed-off-by: Mario Limonciello <mario.limonciello@amd.com>
    Signed-off-by: Mika Westerberg <mika.westerberg@linux.intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a599be1bb34690dffd8dd1a4e555bf9efc68bb92
Author: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
Date:   Fri May 5 17:23:01 2023 +0200

    serial: qcom-geni: fix enabling deactivated interrupt

    commit 5f949f140f73696f64acb89a1f16ff9153d017e0 upstream.

    The driver have a race, experienced only with PREEMPT_RT patchset:

    CPU0                          | CPU1
    ==============================================================
    qcom_geni_serial_probe        |
      uart_add_one_port           |
                                  | serdev_drv_probe
                                  |   qca_serdev_probe
                                  |     serdev_device_open
                                  |       uart_open
                                  |         uart_startup
                                  |           qcom_geni_serial_startup
                                  |             enable_irq
                                  |               __irq_startup
                                  |                 WARN_ON()
                                  |                 IRQ not activated
      request_threaded_irq        |
        irq_domain_activate_irq   |

    The warning:

      894000.serial: ttyHS1 at MMIO 0x894000 (irq = 144, base_baud = 0) is a MSM
      serial serial0: tty port ttyHS1 registered
      WARNING: CPU: 7 PID: 107 at kernel/irq/chip.c:241 __irq_startup+0x78/0xd8
      ...
      qcom_geni_serial 894000.serial: serial engine reports 0 RX bytes in!

    Adding UART port triggers probe of child serial devices - serdev and
    eventually Qualcomm Bluetooth hci_qca driver.  This opens UART port
    which enables the interrupt before it got activated in
    request_threaded_irq().  The issue originates in commit f3974413cf02
    ("tty: serial: qcom_geni_serial: Wakeup IRQ cleanup") and discussion on
    mailing list [1].  However the above commit does not explain why the
    uart_add_one_port() is moved above requesting interrupt.

    [1] https://lore.kernel.org/all/5d9f3dfa.1c69fb81.84c4b.30bf@mx.google.com/

    Fixes: f3974413cf02 ("tty: serial: qcom_geni_serial: Wakeup IRQ cleanup")
    Cc: <stable@vger.kernel.org>
    Cc: Stephen Boyd <swboyd@chromium.org>
    Signed-off-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Reviewed-by: Stephen Boyd <swboyd@chromium.org>
    Link: https://lore.kernel.org/r/20230505152301.2181270-1-krzysztof.kozlowski@linaro.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e3e52010822fcb005ff1d39745578d5e7292e0d0
Author: Andrew Davis <afd@ti.com>
Date:   Thu Apr 20 11:02:09 2023 -0500

    serial: 8250_exar: Add support for USR298x PCI Modems

    commit 95d698869b404772cc8b72560df71548491c10bc upstream.
```

```
    Possibly the last PCI controller-based (i.e. not a soft/winmodem)
    dial-up modem one can still buy.

    Looks to have a stock XR17C154 PCI UART chip for communication, but for
    some reason when provisioning the PCI IDs they swapped the vendor and
    subvendor IDs. Otherwise this card would have worked out of the box.

    Searching online, some folks seem to not have this issue and others do,
    so it is possible only some batches of cards have this error.

    Create a new macro to handle the switched IDs and add support here.

    Signed-off-by: Andrew Davis <afd@ti.com>
    Cc: stable <stable@kernel.org>
    Reviewed-by: Andy Shevchenko <andriy.shevchenko@linux.intel.com>
    Link: https://lore.kernel.org/r/20230420160209.28221-1-afd@ti.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit bb6cc3e05c2c36e75147933fff189ea8ae6b1621
Author: Vitaliy Tomin <tomin@iszf.irk.ru>
Date:   Sun Apr 23 11:45:12 2023 +0800

    serial: Add support for Advantech PCI-1611U card

    commit d2b00516de0e1d696724247098f6733a6ea53908 upstream.

    Add support for Advantech PCI-1611U card

    Advantech provides opensource drivers for this and many others card
    based on legacy copy of 8250_pci driver called adv950

    https://www.advantech.com/emt/support/details/driver?id=1-TDOIMJ

    It is hard to maintain to run as out of tree module on newer kernels.
    Just adding PCI ID to kernel 8250_pci works perfect.

    Signed-off-by: Vitaliy Tomin <tomin@iszf.irk.ru>
    Cc: stable <stable@kernel.org>
    Link: https://lore.kernel.org/r/20230423034512.2671157-1-tomin@iszf.irk.ru
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 18ba08c83957f03f5944a2f1716223b9c95dfa31
Author: Sandipan Das <sandipan.das@amd.com>
Date:   Fri May 5 15:32:53 2023 +0530

    perf script: Skip aggregation for stat events

    commit 2fe6575924612f1014a0539ab3053b106aded926 upstream.

    The script command does not support aggregation modes by itself although
    that can be achieved using post-processing scripts. Because of this, it
    does not allocate memory for aggregated event values.

    Upon running perf stat record, the aggregation mode is set in the perf
    data file. If the mode is AGGR_GLOBAL, the aggregated event values are
    accessed and this leads to a segmentation fault since these were never
    allocated to begin with. Set the mode to AGGR_NONE explicitly to avoid
    this.

    E.g.

      $ perf stat record -e cycles true
      $ perf script
```

```
    Before:
      Segmentation fault (core dumped)

    After:
      CPU   THREAD             VAL             ENA             RUN           TIME EVENT
       -1   231919          162831          362069          362069          935289 cycles:u

    Fixes: 8b76a3188b85724f ("perf stat: Remove unused perf_counts.aggr field")
    Signed-off-by: Sandipan Das <sandipan.das@amd.com>
    Acked-by: Namhyung Kim <namhyung@kernel.org>
    Cc: Adrian Hunter <adrian.hunter@intel.com>
    Cc: Alexander Shishkin <alexander.shishkin@linux.intel.com>
    Cc: Ananth Narayan <ananth.narayan@amd.com>
    Cc: Ian Rogers <irogers@google.com>
    Cc: Ingo Molnar <mingo@redhat.com>
    Cc: Jiri Olsa <jolsa@kernel.org>
    Cc: Mark Rutland <mark.rutland@arm.com>
    Cc: Namhyung Kim <namhyung@kernel.org>
    Cc: Nick Terrell <terrelln@fb.com>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Cc: Ravi Bangoria <ravi.bangoria@amd.com>
    Cc: stable@vger.kernel.org # v6.2+
    Link:
https://lore.kernel.org/r/83d6c6c05c54bf00c5a9df32ac160718efca0c7a.1683280603.git.sandipan.d
as@amd.com
    Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ba700ea13bf0105a4773c654f7d3bef8adb64ab2
Author: Domenico Cerasuolo <cerasuolodomenico@gmail.com>
Date:   Wed May 3 17:12:00 2023 +0200

    mm: fix zswap writeback race condition

    commit 04fc7816089c5a32c29a04ec94b998e219dfb946 upstream.

    The zswap writeback mechanism can cause a race condition resulting in
    memory corruption, where a swapped out page gets swapped in with data that
    was written to a different page.

    The race unfolds like this:
    1. a page with data A and swap offset X is stored in zswap
    2. page A is removed off the LRU by zpool driver for writeback in
       zswap-shrink work, data for A is mapped by zpool driver
    3. user space program faults and invalidates page entry A, offset X is
       considered free
    4. kswapd stores page B at offset X in zswap (zswap could also be
       full, if so, page B would then be IOed to X, then skip step 5.)
    5. entry A is replaced by B in tree->rbroot, this doesn't affect the
       local reference held by zswap-shrink work
    6. zswap-shrink work writes back A at X, and frees zswap entry A
    7. swapin of slot X brings A in memory instead of B

    The fix:
    Once the swap page cache has been allocated (case ZSWAP_SWAPCACHE_NEW),
    zswap-shrink work just checks that the local zswap_entry reference is
    still the same as the one in the tree.  If it's not the same it means that
    it's either been invalidated or replaced, in both cases the writeback is
    aborted because the local entry contains stale data.

    Reproducer:
    I originally found this by running `stress` overnight to validate my work
    on the zswap writeback mechanism, it manifested after hours on my test
    machine.  The key to make it happen is having zswap writebacks, so
    whatever setup pumps /sys/kernel/debug/zswap/written_back_pages should do
```

```
    the trick.

    In order to reproduce this faster on a vm, I setup a system with ~100M of
    available memory and a 500M swap file, then running `stress --vm 1
    --vm-bytes 300000000 --vm-stride 4000` makes it happen in matter of tens
    of minutes.  One can speed things up even more by swinging
    /sys/module/zswap/parameters/max_pool_percent up and down between, say, 20
    and 1; this makes it reproduce in tens of seconds.  It's crucial to set
    `--vm-stride` to something other than 4096 otherwise `stress` won't
    realize that memory has been corrupted because all pages would have the
    same data.

    Link: https://lkml.kernel.org/r/20230503151200.19707-1-cerasuolodomenico@gmail.com
    Signed-off-by: Domenico Cerasuolo <cerasuolodomenico@gmail.com>
    Acked-by: Johannes Weiner <hannes@cmpxchg.org>
    Reviewed-by: Chris Li (Google) <chrisl@kernel.org>
    Cc: Dan Streetman <ddstreet@ieee.org>
    Cc: Johannes Weiner <hannes@cmpxchg.org>
    Cc: Minchan Kim <minchan@kernel.org>
    Cc: Nitin Gupta <ngupta@vflare.org>
    Cc: Seth Jennings <sjenning@redhat.com>
    Cc: Vitaly Wool <vitaly.wool@konsulko.com>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a0306268276e2354f84ccd1296e9f229e86302ed
Author: Peng Zhang <zhangpeng.00@bytedance.com>
Date:   Fri May 5 22:58:29 2023 +0800

    maple_tree: make maple state reusable after mas_empty_area()

    commit 0257d9908d38c0b1669af4bb1bc4dbca1f273fe6 upstream.

    Make mas->min and mas->max point to a node range instead of a leaf entry
    range.  This allows mas to still be usable after mas_empty_area() returns.
    Users would get unexpected results from other operations on the maple
    state after calling the affected function.

    For example, x86 MAP_32BIT mmap() acts as if there is no suitable gap when
    there should be one.

    Link: https://lkml.kernel.org/r/20230505145829.74574-1-zhangpeng.00@bytedance.com
    Fixes: 54a611b60590 ("Maple Tree: add new data structure")
    Signed-off-by: Peng Zhang <zhangpeng.00@bytedance.com>
    Reported-by: "Edgecombe, Rick P" <rick.p.edgecombe@intel.com>
    Reported-by: Tad <support@spotco.us>
    Reported-by: Michael Keyes <mgkeyes@vigovproductions.net>
      Link: https://lore.kernel.org/linux-
mm/32f156ba80010fd97dbaf0a0cdfc84366608624d.camel@intel.com/
      Link: https://lore.kernel.org/linux-
mm/e6108286ac025c268964a7ead3aab9899f9bc6e9.camel@spotco.us/
    Reviewed-by: Liam R. Howlett <Liam.Howlett@oracle.com>
    Tested-by: Rick Edgecombe <rick.p.edgecombe@intel.com>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 2497870442d4aab258bb299b48eb356c6944495f
Author: Ilya Leoshkevich <iii@linux.ibm.com>
Date:   Thu May 4 16:40:20 2023 +0200

    statfs: enforce statfs[64] structure initialization

    commit ed40866ec7d328b3dfb70db7e2011640a16202c3 upstream.
```

```
    s390's struct statfs and struct statfs64 contain padding, which
    field-by-field copying does not set. Initialize the respective structs
    with zeros before filling them and copying them to userspace, like it's
    already done for the compat versions of these structs.

    Found by KMSAN.

    [agordeev@linux.ibm.com: fixed typo in patch description]
    Acked-by: Heiko Carstens <hca@linux.ibm.com>
    Cc: stable@vger.kernel.org # v4.14+
    Signed-off-by: Ilya Leoshkevich <iii@linux.ibm.com>
    Reviewed-by: Andrew Morton <akpm@linux-foundation.org>
    Link: https://lore.kernel.org/r/20230504144021.808932-2-iii@linux.ibm.com
    Signed-off-by: Alexander Gordeev <agordeev@linux.ibm.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 556b57acb3de215a2e1fd4bac9e8bf32a1a8c463
Author: Michal Luczaj <mhal@rbox.co>
Date:   Wed May 10 16:04:09 2023 +0200

    KVM: Fix vcpu_array[0] races

    commit afb2acb2e3a32e4d56f7fbd819769b98ed1b7520 upstream.

    In kvm_vm_ioctl_create_vcpu(), add vcpu to vcpu_array iff it's safe to
    access vcpu via kvm_get_vcpu() and kvm_for_each_vcpu(), i.e. when there's
    no failure path requiring vcpu removal and destruction. Such order is
    important because vcpu_array accessors may end up referencing vcpu at
    vcpu_array[0] even before online_vcpus is set to 1.

    When online_vcpus=0, any call to kvm_get_vcpu() goes through
    array_index_nospec() and ends with an attempt to xa_load(vcpu_array, 0):

            int num_vcpus = atomic_read(&kvm->online_vcpus);
            i = array_index_nospec(i, num_vcpus);
            return xa_load(&kvm->vcpu_array, i);

    Similarly, when online_vcpus=0, a kvm_for_each_vcpu() does not iterate over
    an "empty" range, but actually [0, ULONG_MAX]:

            xa_for_each_range(&kvm->vcpu_array, idx, vcpup, 0, \
                              (atomic_read(&kvm->online_vcpus) - 1))

    In both cases, such online_vcpus=0 edge case, even if leading to
    unnecessary calls to XArray API, should not be an issue; requesting
    unpopulated indexes/ranges is handled by xa_load() and xa_for_each_range().

    However, this means that when the first vCPU is created and inserted in
    vcpu_array *and* before online_vcpus is incremented, code calling
    kvm_get_vcpu()/kvm_for_each_vcpu() already has access to that first vCPU.

    This should not pose a problem assuming that once a vcpu is stored in
    vcpu_array, it will remain there, but that's not the case:
    kvm_vm_ioctl_create_vcpu() first inserts to vcpu_array, then requests a
    file descriptor. If create_vcpu_fd() fails, newly inserted vcpu is removed
    from the vcpu_array, then destroyed:

            vcpu->vcpu_idx = atomic_read(&kvm->online_vcpus);
            r = xa_insert(&kvm->vcpu_array, vcpu->vcpu_idx, vcpu, GFP_KERNEL_ACCOUNT);
            kvm_get_kvm(kvm);
            r = create_vcpu_fd(vcpu);
            if (r < 0) {
                    xa_erase(&kvm->vcpu_array, vcpu->vcpu_idx);
                    kvm_put_kvm_no_destroy(kvm);
```

```
                goto unlock_vcpu_destroy;
        }
        atomic_inc(&kvm->online_vcpus);
```

This results in a possible race condition when a reference to a vcpu is acquired (via kvm_get_vcpu() or kvm_for_each_vcpu()) moments before said vcpu is destroyed.

Signed-off-by: Michal Luczaj <mhal@rbox.co>
Message-Id: <20230510140410.1093987-2-mhal@rbox.co>
Cc: stable@vger.kernel.org
Fixes: c5b077549136 ("KVM: Convert the kvm->vcpus array to a xarray", 2021-12-08)
Signed-off-by: Paolo Bonzini <pbonzini@redhat.com>
Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 00e6cbf519f88582fee5bd22e477292aec0fb81a
Author: Oliver Upton <oliver.upton@linux.dev>
Date:   Fri Apr 21 07:16:05 2023 +0000

KVM: arm64: Infer the PA offset from IPA in stage-2 map walker

commit 1f0f4a2ef7a5693b135ce174e71f116db4bd684d upstream.

Until now, the page table walker counted increments to the PA and IPA of a walk in two separate places. While the PA is incremented as soon as a leaf PTE is installed in stage2_map_walker_try_leaf(), the IPA is actually bumped in the generic table walker context. Critically, __kvm_pgtable_visit() rereads the PTE after the LEAF callback returns to work out if a table or leaf was installed, and only bumps the IPA for a leaf PTE.

This arrangement worked fine when we handled faults behind the write lock, as the walker had exclusive access to the stage-2 page tables. However, commit 1577cb5823ce ("KVM: arm64: Handle stage-2 faults in parallel") started handling all stage-2 faults behind the read lock, opening up a race where a walker could increment the PA but not the IPA of a walk. Nothing good ensues, as the walker starts mapping with the incorrect IPA -> PA relationship.

For example, assume that two vCPUs took a data abort on the same IPA. One observes that dirty logging is disabled, and the other observed that it is enabled:

```
  vCPU attempting PMD mapping             vCPU attempting PTE mapping
  ====================================  ====================================
  /* install PMD */
  stage2_make_pte(ctx, leaf);
  data->phys += granule;
                                        /* replace PMD with a table */
                                        stage2_try_break_pte(ctx, data->mmu);
                                        stage2_make_pte(ctx, table);
  /* table is observed */
  ctx.old = READ_ONCE(*ptep);
  table = kvm_pte_table(ctx.old, level);

  /*
   * map walk continues w/o incrementing
   * IPA.
   */
   __kvm_pgtable_walk(..., level + 1);
```

Bring an end to the whole mess by using the IPA as the single source of truth for how far along a walk has gotten. Work out the correct PA to map by calculating the IPA offset from the beginning of the walk and add that to the starting physical address.

```
    Cc: stable@vger.kernel.org
    Fixes: 1577cb5823ce ("KVM: arm64: Handle stage-2 faults in parallel")
    Signed-off-by: Oliver Upton <oliver.upton@linux.dev>
    Signed-off-by: Marc Zyngier <maz@kernel.org>
    Link: https://lore.kernel.org/r/20230421071606.1603916-2-oliver.upton@linux.dev
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 0adcdc220fa555935bb37a273f08956616f8601a
Author: Chih-Yen Chang <cc85nod@gmail.com>
Date:   Sun May 14 12:05:05 2023 +0900

    ksmbd: fix global-out-of-bounds in smb2_find_context_vals

    commit 02f76c401d17e409ed45bf7887148fcc22c93c85 upstream.

    Add tag_len argument in smb2_find_context_vals() to avoid out-of-bound
    read when create_context's name_len is larger than tag length.

    [    7.995411] ==================================================================
    [    7.995866] BUG: KASAN: global-out-of-bounds in memcmp+0x83/0xa0
    [    7.996248] Read of size 8 at addr ffffffff8258d940 by task kworker/0:0/7
    ...
    [    7.998191] Call Trace:
    [    7.998358]  <TASK>
    [    7.998503]  dump_stack_lvl+0x33/0x50
    [    7.998743]  print_report+0xcc/0x620
    [    7.999458]  kasan_report+0xae/0xe0
    [    7.999895]  kasan_check_range+0x35/0x1b0
    [    8.000152]  memcmp+0x83/0xa0
    [    8.000347]  smb2_find_context_vals+0xf7/0x1e0
    [    8.000635]  smb2_open+0x1df2/0x43a0
    [    8.006398]  handle_ksmbd_work+0x274/0x810
    [    8.006666]  process_one_work+0x419/0x760
    [    8.006922]  worker_thread+0x2a2/0x6f0
    [    8.007429]  kthread+0x160/0x190
    [    8.007946]  ret_from_fork+0x1f/0x30
    [    8.008181]  </TASK>

    Cc: stable@vger.kernel.org
    Signed-off-by: Chih-Yen Chang <cc85nod@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 3df195fbddfae60ca24a9bbc209402d9fccdef68
Author: Chih-Yen Chang <cc85nod@gmail.com>
Date:   Sat May 6 00:01:54 2023 +0900

    ksmbd: fix wrong UserName check in session_user

    commit f0a96d1aafd8964e1f9955c830a3e5cb3c60a90f upstream.

    The offset of UserName is related to the address of security
    buffer. To ensure the validaty of UserName, we need to compare name_off
    + name_len with secbuf_len instead of auth_msg_len.

    [   27.096243] ==================================================================
    [   27.096890] BUG: KASAN: slab-out-of-bounds in smb_strndup_from_utf16+0x188/0x350
    [   27.097609] Read of size 2 at addr ffff888005e3b542 by task kworker/0:0/7
    ...
    [   27.099950] Call Trace:
    [   27.100194]  <TASK>
    [   27.100397]  dump_stack_lvl+0x33/0x50
    [   27.100752]  print_report+0xcc/0x620
```

```
      [   27.102305]  kasan_report+0xae/0xe0
      [   27.103072]  kasan_check_range+0x35/0x1b0
      [   27.103757]  smb_strndup_from_utf16+0x188/0x350
      [   27.105474]  smb2_sess_setup+0xaf8/0x19c0
      [   27.107935]  handle_ksmbd_work+0x274/0x810
      [   27.108315]  process_one_work+0x419/0x760
      [   27.108689]  worker_thread+0x2a2/0x6f0
      [   27.109385]  kthread+0x160/0x190
      [   27.110129]  ret_from_fork+0x1f/0x30
      [   27.110454]  </TASK>

      Cc: stable@vger.kernel.org
      Signed-off-by: Chih-Yen Chang <cc85nod@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 778aae5a513ea09aa5addfa352bd70a5b71dda85
  Author: Chih-Yen Chang <cc85nod@gmail.com>
  Date:   Sat May 6 00:03:54 2023 +0900

      ksmbd: allocate one more byte for implied bcc[0]

      commit 443d61d1fa9faa60ef925513d83742902390100f upstream.

      ksmbd_smb2_check_message allows client to return one byte more, so we
      need to allocate additional memory in ksmbd_conn_handler_loop to avoid
      out-of-bound access.

      Cc: stable@vger.kernel.org
      Signed-off-by: Chih-Yen Chang <cc85nod@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 69e576f03e2e4c41402d1673a96b0cdad25240e2
  Author: Gustav Johansson <gustajo@axis.com>
  Date:   Sat May 6 00:05:07 2023 +0900

      ksmbd: smb2: Allow messages padded to 8byte boundary

      commit e7b8b8ed9960bf699bf4029f482d9e869c094ed6 upstream.

      clc length is now accepted to <= 8 less than length,
      rather than < 8.

      Solve issues on some of Axis's smb clients which send
      messages where clc length is 8 bytes less than length.

      The specific client was running kernel 4.19.217 with
      smb dialect 3.0.2 on armv7l.

      Cc: stable@vger.kernel.org
      Signed-off-by: Gustav Johansson <gustajo@axis.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 849e96c74a3f1ff8c1aaeb40f387d00f24f0f3f8
  Author: Bharath SM <bharathsm@microsoft.com>
  Date:   Mon May 15 21:25:12 2023 +0000

      SMB3: drop reference to cfile before sending oplock break

      commit 59a556aebc43dded08535fe97d94ca3f657915e4 upstream.
```

```
    In cifs_oplock_break function we drop reference to a cfile at
    the end of function, due to which close command goes on wire
    after lease break acknowledgment even if file is already closed
    by application but we had deferred the handle close.
    If other client with limited file shareaccess waiting on lease
    break ack proceeds operation on that file as soon as first client
    sends ack, then we may encounter status sharing violation error
    because of open handle.
    Solution is to put reference to cfile(send close on wire if last ref)
    and then send oplock acknowledgment to server.

    Fixes: 9e31678fb403 ("SMB3: fix lease break timeout when multiple deferred close handles
for the same file.")
    Cc: stable@kernel.org
    Signed-off-by: Bharath SM <bharathsm@microsoft.com>
    Reviewed-by: Shyam Prasad N <sprasad@microsoft.com>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 545954f66de37ff6736e6f5f5c474822baa19e83
Author: Bharath SM <bharathsm@microsoft.com>
Date:   Wed May 3 14:38:35 2023 +0000

    SMB3: Close all deferred handles of inode in case of handle lease break

    commit 47592fa8eb03742048b096b4696ec133384c45eb upstream.

    Oplock break may occur for different file handle than the deferred
    handle. Check for inode deferred closes list, if it's not empty then
    close all the deferred handles of inode because we should not cache
    handles if we dont have handle lease.

    Eg: If openfilelist has one deferred file handle and another open file
    handle from app for a same file, then on a lease break we choose the
    first handle in openfile list. The first handle in list can be deferred
    handle or actual open file handle from app. In case if it is actual open
    handle then today, we don't close deferred handles if we lose handle lease
    on a file. Problem with this is, later if app decides to close the existing
    open handle then we still be caching deferred handles until deferred close
    timeout. Leaving open handle may result in sharing violation when windows
    client tries to open a file with limited file share access.

    So we should check for deferred list of inode and walk through the list of
    deferred files in inode and close all deferred files.

    Fixes: 9e31678fb403 ("SMB3: fix lease break timeout when multiple deferred close handles
for the same file.")
    Cc: stable@kernel.org
    Signed-off-by: Bharath SM <bharathsm@microsoft.com>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1c2fab552304f4b06f6ac0d395a7d580ea7c8fa1
Author: Ping-Ke Shih <pkshih@realtek.com>
Date:   Mon May 8 16:55:39 2023 +0800

    wifi: rtw88: correct qsel_to_ep[] type as int

    commit 8e4942db5f5ed7b7d9690d93235b3ca49c5c59ce upstream.

    qsel_to_ep[] can be assigned negative value, so change type from 'u8' to
    'int'. Otherwise, Smatch static checker warns:
      drivers/net/wireless/realtek/rtw88/usb.c:219 rtw_usb_parse() warn:
      assigning (-22) to unsigned variable 'rtwusb->qsel_to_ep[8]'
```

```
    Cc: stable@vger.kernel.org
    Fixes: a6f187f92bcc ("wifi: rtw88: usb: fix priority queue to endpoint mapping")
    Reported-by: Dan Carpenter <dan.carpenter@linaro.org>
    Link: https://lore.kernel.org/linux-wireless/c3f70197-829d-48ed-ae15-
66a9de80fa90@kili.mountain/
    Cc: Sascha Hauer <s.hauer@pengutronix.de>
    Signed-off-by: Ping-Ke Shih <pkshih@realtek.com>
    Acked-by: Sascha Hauer <s.hauer@pengutronix.de>
    Tested-by: Larry Finger <Larry.Finger@lwfinger.net>
    Signed-off-by: Kalle Valo <kvalo@kernel.org>
    Link: https://lore.kernel.org/r/20230508085539.46795-1-pkshih@realtek.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit dd3af22323e79a2ffabed366db20aab83716fe6f
Author: Ping-Ke Shih <pkshih@realtek.com>
Date:   Mon May 8 16:54:29 2023 +0800

    wifi: rtw88: use work to update rate to avoid RCU warning

    commit bcafcb959a57a6890e900199690c5fc47da1a304 upstream.

    The ieee80211_ops::sta_rc_update must be atomic, because
    ieee80211_chan_bw_change() holds rcu_read lock while calling
    drv_sta_rc_update(), so create a work to do original things.

     Voluntary context switch within RCU read-side critical section!
     WARNING: CPU: 0 PID: 4621 at kernel/rcu/tree_plugin.h:318
     rcu_note_context_switch+0x571/0x5d0
     CPU: 0 PID: 4621 Comm: kworker/u16:2 Tainted: G        W  OE
     Workqueue: phy3 ieee80211_chswitch_work [mac80211]
     RIP: 0010:rcu_note_context_switch+0x571/0x5d0
     Call Trace:
      <TASK>
      __schedule+0xb0/0x1460
      ? __mod_timer+0x116/0x360
      schedule+0x5a/0xc0
      schedule_timeout+0x87/0x150
      ? trace_raw_output_tick_stop+0x60/0x60
      wait_for_completion_timeout+0x7b/0x140
      usb_start_wait_urb+0x82/0x160 [usbcore
      usb_control_msg+0xe3/0x140 [usbcore
      rtw_usb_read+0x88/0xe0 [rtw_usb
      rtw_usb_read8+0xf/0x10 [rtw_usb
      rtw_fw_send_h2c_command+0xa0/0x170 [rtw_core
      rtw_fw_send_ra_info+0xc9/0xf0 [rtw_core
      drv_sta_rc_update+0x7c/0x160 [mac80211
      ieee80211_chan_bw_change+0xfb/0x110 [mac80211
      ieee80211_change_chanctx+0x38/0x130 [mac80211
      ieee80211_vif_use_reserved_switch+0x34e/0x900 [mac80211
      ieee80211_link_use_reserved_context+0x88/0xe0 [mac80211
      ieee80211_chswitch_work+0x95/0x170 [mac80211
      process_one_work+0x201/0x410
      worker_thread+0x4a/0x3b0
      ? process_one_work+0x410/0x410
      kthread+0xe1/0x110
      ? kthread_complete_and_exit+0x20/0x20
      ret_from_fork+0x1f/0x30
      </TASK>

    Cc: stable@vger.kernel.org
    Fixes: c1edc86472fc ("rtw88: add ieee80211:sta_rc_update ops")
    Reported-by: Larry Finger <Larry.Finger@lwfinger.net>
    Link: https://lore.kernel.org/linux-wireless/f1e31e8e-f84e-3791-50fb-
663a83c5c6e9@lwfinger.net/T/#t
```

```
    Signed-off-by: Ping-Ke Shih <pkshih@realtek.com>
    Tested-by: Larry Finger <Larry.Finger@lwfinger.net>
    Signed-off-by: Kalle Valo <kvalo@kernel.org>
    Link: https://lore.kernel.org/r/20230508085429.46653-1-pkshih@realtek.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 84766e77a5c35e2b60e34f570c62fc97adc05e09
Author: Hans de Goede <hdegoede@redhat.com>
Date:   Wed May 10 16:18:56 2023 +0200

    wifi: brcmfmac: Check for probe() id argument being NULL

    commit 60fc756fc8e6954a5618eecac73b255d651602e4 upstream.

    The probe() id argument may be NULL in 2 scenarios:

    1. brcmf_pcie_pm_leave_D3() calling brcmf_pcie_probe() to reprobe
       the device.

    2. If a user tries to manually bind the driver from sysfs then the sdio /
       pcie / usb probe() function gets called with NULL as id argument.

    1. Is being hit by users causing the following oops on resume and causing
    wifi to stop working:

    BUG: kernel NULL pointer dereference, address: 0000000000000018
    <snip>
    Hardware name: Dell Inc. XPS 13 9350/0PWNCR, BIDS 1.13.0 02/10/2020
    Workqueue: events_unbound async_run_entry_fn
    RIP: 0010:brcmf_pcie_probe+0x16b/0x7a0 [brcmfmac]
    <snip>
    Call Trace:
     <TASK>
     brcmf_pcie_pm_leave_D3+0xc5/0x1a0 [brcmfmac be3b4cefca451e190fa35be8f00db1bbec293887]
     ? pci_pm_resume+0x5b/0xf0
     ? pci_legacy_resume+0x80/0x80
     dpm_run_callback+0x47/0x150
     device_resume+0xa2/0x1f0
     async_resume+0x1d/0x30
    <snip>

    Fix this by checking for id being NULL.

    In the PCI and USB cases try a manual lookup of the id so that manually
    binding the driver through sysfs and more importantly brcmf_pcie_probe()
    on resume will work.

    For the SDIO case there is no helper to do a manual sdio_device_id lookup,
    so just directly error out on a NULL id there.

    Fixes: da6d9c8ecd00 ("wifi: brcmfmac: add firmware vendor info in driver info")
    Reported-by: Felix <nimrod4garoa@gmail.com>
    Link:
https://lore.kernel.org/regressions/4ef3f252ff530cbfa336f5a0d80710020fc5cb1e.camel@gmail.com
/
    Cc: stable@vger.kernel.org
    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Reviewed-by: Arend van Spriel <arend.vanspriel@broadcom.com>
    Signed-off-by: Kalle Valo <kvalo@kernel.org>
    Link: https://lore.kernel.org/r/20230510141856.46532-1-hdegoede@redhat.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f4741b0e34edced172f5b24de8f427b5b0480e20
Author: Jimmy Assarsson <extja@kvaser.com>
Date:   Tue May 16 15:43:18 2023 +0200
```

```
    can: kvaser_pciefd: Disable interrupts in probe error path

    commit 11164bc39459335ab93c6e99d53b7e4292fba38b upstream.

    Disable interrupts in error path of probe function.

    Fixes: 26ad340e582d ("can: kvaser_pciefd: Add driver for Kvaser PCIEcan devices")
    Cc: stable@vger.kernel.org
    Signed-off-by: Jimmy Assarsson <extja@kvaser.com>
    Link: https://lore.kernel.org/r/20230516134318.104279-7-extja@kvaser.com
    Signed-off-by: Marc Kleine-Budde <mkl@pengutronix.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c44254826c5cb1a1202a3cc2408b697d4137d362
Author: Jimmy Assarsson <extja@kvaser.com>
Date:   Tue May 16 15:43:17 2023 +0200

    can: kvaser_pciefd: Do not send EFLUSH command on TFD interrupt

    commit 262d7a52ba27525e3c1203230c9f0524e48bbb34 upstream.

    Under certain circumstances we send two EFLUSH commands, resulting in two
    EFLUSH ack packets, while only expecting a single EFLUSH ack.
    This can cause the driver Tx flush completion to get out of sync.

    To avoid this problem, don't enable the "Transmit buffer flush done" (TFD)
    interrupt and remove the code handling it.
    Now we only send EFLUSH command after receiving status packet with
    "Init detected" (IDET) bit set.

    Fixes: 26ad340e582d ("can: kvaser_pciefd: Add driver for Kvaser PCIEcan devices")
    Cc: stable@vger.kernel.org
    Signed-off-by: Jimmy Assarsson <extja@kvaser.com>
    Link: https://lore.kernel.org/r/20230516134318.104279-6-extja@kvaser.com
    Signed-off-by: Marc Kleine-Budde <mkl@pengutronix.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 6ea84e9328f4cf6d64b4e95ab6b7586df784fc42
Author: Jimmy Assarsson <extja@kvaser.com>
Date:   Tue May 16 15:43:14 2023 +0200

    can: kvaser_pciefd: Clear listen-only bit if not explicitly requested

    commit bf7ac55e991ca177f1ac16be51152f1ef291a4df upstream.

    The listen-only bit was never cleared, causing the controller to
    always use listen-only mode, if previously set.

    Fixes: 26ad340e582d ("can: kvaser_pciefd: Add driver for Kvaser PCIEcan devices")
    Cc: stable@vger.kernel.org
    Signed-off-by: Jimmy Assarsson <extja@kvaser.com>
    Link: https://lore.kernel.org/r/20230516134318.104279-3-extja@kvaser.com
    Signed-off-by: Marc Kleine-Budde <mkl@pengutronix.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 9cb35c6ec0c6ef76ee7fbb8e5356ae023380b690
Author: Jimmy Assarsson <extja@kvaser.com>
Date:   Tue May 16 15:43:16 2023 +0200

    can: kvaser_pciefd: Empty SRB buffer in probe

    commit c589557dd1426f5adf90c7a919d4fde5a3e4ef64 upstream.

    Empty the "Shared receive buffer" (SRB) in probe, to assure we start in a
```

```
    known state, and don't process any irrelevant packets.

    Fixes: 26ad340e582d ("can: kvaser_pciefd: Add driver for Kvaser PCIEcan devices")
    Cc: stable@vger.kernel.org
    Signed-off-by: Jimmy Assarsson <extja@kvaser.com>
    Link: https://lore.kernel.org/r/20230516134318.104279-5-extja@kvaser.com
    Signed-off-by: Marc Kleine-Budde <mkl@pengutronix.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 4a3cac2412328b0156c660a92c9dba6eb88771be
Author: Jimmy Assarsson <extja@kvaser.com>
Date:   Tue May 16 15:43:15 2023 +0200

    can: kvaser_pciefd: Call request_irq() before enabling interrupts

    commit 84762d8da89d29ba842317eb842973e628c27391 upstream.

    Make sure the interrupt handler is registered before enabling interrupts.

    Fixes: 26ad340e582d ("can: kvaser_pciefd: Add driver for Kvaser PCIEcan devices")
    Cc: stable@vger.kernel.org
    Signed-off-by: Jimmy Assarsson <extja@kvaser.com>
    Link: https://lore.kernel.org/r/20230516134318.104279-4-extja@kvaser.com
    Signed-off-by: Marc Kleine-Budde <mkl@pengutronix.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit d2af426a51a937a806de097ac6e7d6324a72365e
Author: Jimmy Assarsson <extja@kvaser.com>
Date:   Tue May 16 15:43:13 2023 +0200

    can: kvaser_pciefd: Set CAN_STATE_STOPPED in kvaser_pciefd_stop()

    commit aed0e6ca7dbb8fbea9bc69c9ac663d5533c8c5d8 upstream.

    Set can.state to CAN_STATE_STOPPED in kvaser_pciefd_stop().
    Without this fix, wrong CAN state was repported after the interface was
    brought down.

    Fixes: 26ad340e582d ("can: kvaser_pciefd: Add driver for Kvaser PCIEcan devices")
    Cc: stable@vger.kernel.org
    Signed-off-by: Jimmy Assarsson <extja@kvaser.com>
    Link: https://lore.kernel.org/r/20230516134318.104279-2-extja@kvaser.com
    Signed-off-by: Marc Kleine-Budde <mkl@pengutronix.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit bfa784e9cac28f4d9f28456a2f4eff4a428ea744
Author: Oliver Hartkopp <socketcan@hartkopp.net>
Date:   Thu Apr 6 13:08:45 2023 +0200

    can: isotp: recvmsg(): allow MSG_CMSG_COMPAT flag

    commit db2773d65b02aed319a93efdfb958087771d4e19 upstream.

    The control message provided by isotp support MSG_CMSG_COMPAT but
    blocked recvmsg() syscalls that have set this flag, i.e. on 32bit user
    space on 64 bit kernels.

    Link: https://github.com/hartkopp/can-isotp/issues/59
    Cc: Oleksij Rempel <o.rempel@pengutronix.de>
    Suggested-by: Marc Kleine-Budde <mkl@pengutronix.de>
    Signed-off-by: Oliver Hartkopp <socketcan@hartkopp.net>
    Fixes: 42bf50a1795a ("can: isotp: support MSG_TRUNC flag when reading from socket")
    Link: https://lore.kernel.org/20230505110308.81087-2-mkl@pengutronix.de
    Cc: stable@vger.kernel.org
    Signed-off-by: Marc Kleine-Budde <mkl@pengutronix.de>
```

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

commit 6f5292fddc5b8bac7553328a9c4c214cb4d66614
Author: Oliver Hartkopp <socketcan@hartkopp.net>
Date:   Thu Apr 6 13:08:45 2023 +0200

    can: j1939: recvmsg(): allow MSG_CMSG_COMPAT flag

    commit 1db080cbdbab28752bbb1c86d64daf96253a5da1 upstream.

    The control message provided by J1939 support MSG_CMSG_COMPAT but
    blocked recvmsg() syscalls that have set this flag, i.e. on 32bit user
    space on 64 bit kernels.

    Link: https://github.com/hartkopp/can-isotp/issues/59
    Cc: Oleksij Rempel <o.rempel@pengutronix.de>
    Suggested-by: Marc Kleine-Budde <mkl@pengutronix.de>
    Signed-off-by: Oliver Hartkopp <socketcan@hartkopp.net>
    Tested-by: Oleksij Rempel <o.rempel@pengutronix.de>
    Acked-by: Oleksij Rempel <o.rempel@pengutronix.de>
    Fixes: 9d71dd0c7009 ("can: add support of SAE J1939 protocol")
    Link: https://lore.kernel.org/20230505110308.81087-3-mkl@pengutronix.de
    Cc: stable@vger.kernel.org
    Signed-off-by: Marc Kleine-Budde <mkl@pengutronix.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 652dc4fa8468cd2feb3c51d02df51c4b90a963f3
Author: Kai-Heng Feng <kai.heng.feng@canonical.com>
Date:   Fri May 12 16:34:16 2023 +0800

    ALSA: hda/realtek: Fix mute and micmute LEDs for yet another HP laptop

    commit 9dc68a4fe70893b000fb3c92c68b9f72369cf448 upstream.

    There's yet another laptop that needs the fixup to enable mute and
    micmute LEDs. So do it accordingly.

    Signed-off-by: Kai-Heng Feng <kai.heng.feng@canonical.com>
    Cc: <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20230512083417.157127-1-kai.heng.feng@canonical.com
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 53f2e1860fdf56b024ed73a17a1a6a2b6e297cfb
Author: Vitaly Rodionov <vitalyr@opensource.cirrus.com>
Date:   Wed May 10 15:22:27 2023 +0100

    ALSA: hda/realtek: Add quirk for HP EliteBook G10 laptops

    commit 3e10f6ca76c4d00019badebd235c9d7f0068261e upstream.

    Add support for HP EliteBook 835/845/845W/865 G10 laptops
    with CS35L41 amplifiers on I2C/SPI bus connected to Realtek codec.

    Signed-off-by: Vitaly Rodionov <vitalyr@opensource.cirrus.com>
    Cc: <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20230510142227.32945-1-vitalyr@opensource.cirrus.com
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 2df932d52a73a2864eaff5f6f79387689ac9175a
Author: Luke D. Jones <luke@ljones.dev>
Date:   Sat May 6 11:58:24 2023 +1200

    ALSA: hda/realtek: Add quirk for 2nd ASUS GU603
```

```
    commit a4671b7fba59775845ee60cfbdfc4ba64300211b upstream.

    Add quirk for GU603 with 0x1c62 variant of codec.

    Signed-off-by: Luke D. Jones <luke@ljones.dev>
    Cc: <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20230505235824.49607-2-luke@ljones.dev
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 4102de2f89548f16cb7423cfb3a593ebf065b2fd
Author: Ai Chao <aichao@kylinos.cn>
Date:   Sat May 6 10:26:53 2023 +0800

    ALSA: hda/realtek: Add a quirk for HP EliteDesk 805

    commit 90670ef774a8b6700c38ce1222e6aa263be54d5f upstream.

    Add a quirk for HP EliteDesk 805 to fixup ALC3867 headset MIC no sound.

    Signed-off-by: Ai Chao <aichao@kylinos.cn>
    Cc: <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20230506022653.2074343-1-aichao@kylinos.cn
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3721a0cb58b7e246f78d9a6913cff468c99bd911
Author: Jeremy Soller <jeremy@system76.com>
Date:   Fri May 5 10:36:51 2023 -0600

    ALSA: hda/realtek: Add quirk for Clevo L140AU

    commit 0a6b36c5dc3dda0196f4fb65bdb34c38b8d060c3 upstream.

    Fixes headset detection on Clevo L140AU.

    Signed-off-by: Jeremy Soller <jeremy@system76.com>
    Signed-off-by: Tim Crawford <tcrawford@system76.com>
    Cc: <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20230505163651.21257-1-tcrawford@system76.com
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 6cd24510cb777a45a68b2485b293f42fd718b26a
Author: Nikhil Mahale <nmahale@nvidia.com>
Date:   Wed May 17 14:37:36 2023 +0530

    ALSA: hda: Add NVIDIA codec IDs a3 through a7 to patch table

    commit dc4f2ccaedddb489a83e7b12ebbdc347272aacc9 upstream.

    These IDs are for AD102, AD103, AD104, AD106, and AD107 gpus with
    audio functions that are largely similar to the existing ones.

    Tested audio using gnome-settings, over HDMI, DP-SST and DP-MST
    connections on AD106 gpu.

    Signed-off-by: Nikhil Mahale <nmahale@nvidia.com>
    Cc: <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20230517090736.15088-1-nmahale@nvidia.com
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit fcf637461019e9a5a0c12fc5c42a9db1779b0634
```

```
Author: Takashi Iwai <tiwai@suse.de>
Date:   Tue May 16 20:44:12 2023 +0200

    ALSA: hda: Fix Oops by 9.1 surround channel names

    commit 3b44ec8c5c44790a82f07e90db45643c762878c6 upstream.

    get_line_out_pfx() may trigger an Oops by overflowing the static array
    with more than 8 channels.  This was reported for MacBookPro 12,1 with
    Cirrus codec.

    As a workaround, extend for the 9.1 channels and also fix the
    potential Oops by unifying the code paths accessing the same array
    with the proper size check.

    Reported-by: Olliver Schinagl <oliver@schinagl.nl>
    Cc: <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/64d95eb0-dbdb-cff8-a8b1-988dc22b24cd@schinagl.nl
    Link: https://lore.kernel.org/r/20230516184412.24078-1-tiwai@suse.de
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c7cf2cec5160092f8c04f5c650df1020d4d7a8b0
Author: Mathias Nyman <mathias.nyman@linux.intel.com>
Date:   Mon May 15 16:40:59 2023 +0300

    xhci: Fix incorrect tracking of free space on transfer rings

    commit fe82f16aafdaf8002281d3b9524291d4a4a28460 upstream.

    This incorrect tracking caused unnecessary ring expansion in some
    usecases which over days of use consume a lot of memory.

    xhci driver tries to keep track of free transfer blocks (TRBs) on the
    ring buffer, but failed to add back some cancelled transfers that were
    turned into no-op operations instead of just moving past them.

    This can happen if there are several queued pending transfers which
    then are cancelled in reverse order.

    Solve this by counting the numer of steps we move the dequeue pointer
    once we complete a transfer, and add it to the number of free trbs
    instead of just adding the trb number of the current transfer.
    This way we ensure we count the no-op trbs on the way as well.

    Fixes: 55f6153d8cc8 ("xhci: remove extra loop in interrupt context")
    Cc: stable@vger.kernel.org
    Reported-by: Miller Hunter <MillerH@hearthnhome.com>
    Closes: https://bugzilla.kernel.org/show_bug.cgi?id=217242
    Tested-by: Miller Hunter <MillerH@hearthnhome.com>
    Signed-off-by: Mathias Nyman <mathias.nyman@linux.intel.com>
    Link: https://lore.kernel.org/r/20230515134059.161110-3-mathias.nyman@linux.intel.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 6e25ced4b8c8c4d2987da43858d4e65d508c8d19
Author: Mario Limonciello <mario.limonciello@amd.com>
Date:   Mon May 15 16:40:58 2023 +0300

    xhci-pci: Only run d3cold avoidance quirk for s2idle

    commit 2a821fc3136d5d99dcb9de152be8a052ca27d870 upstream.

    Donghun reports that a notebook that has an AMD Ryzen 5700U but supports
    S3 has problems with USB after resuming from suspend. The issue was
    bisected down to commit d1658268e439 ("usb: pci-quirks: disable D3cold on
```

```
    xhci suspend for s2idle on AMD Renoir").

    As this issue only happens on S3, narrow the broken D3cold quirk to only
    run in s2idle.

    Fixes: d1658268e439 ("usb: pci-quirks: disable D3cold on xhci suspend for s2idle on AMD
Renoir")
    Reported-and-tested-by: Donghun Yoon <donghun.yoon@lge.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Mario Limonciello <mario.limonciello@amd.com>
    Signed-off-by: Mathias Nyman <mathias.nyman@linux.intel.com>
    Link: https://lore.kernel.org/r/20230515134059.161110-2-mathias.nyman@linux.intel.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c93e346fa0ddfea4cc345ac09f610614c4121323
Author: Francesco Dolcini <francesco.dolcini@toradex.com>
Date:   Fri May 12 15:14:35 2023 +0200

    Revert "usb: gadget: udc: core: Invoke usb_gadget_connect only when started"

    commit f22e9b67f19ccc73de1ae04375d4b30684e261f8 upstream.

    This reverts commit 0db213ea8eed5534a5169e807f28103cbc9d23df.

    It introduces an issues with configuring the USB gadget hangs forever
    on multiple Qualcomm and NXP i.MX SoC at least.

    Cc: stable@vger.kernel.org
    Fixes: 0db213ea8eed ("usb: gadget: udc: core: Invoke usb_gadget_connect only when
started")
    Reported-by: Stephan Gerhold <stephan@gerhold.net>
    Reported-by: Francesco Dolcini <francesco.dolcini@toradex.com>
    Link: https://lore.kernel.org/all/ZF4BvgsOyoKxdPFF@francesco-nb.int.toradex.com/
    Signed-off-by: Francesco Dolcini <francesco.dolcini@toradex.com>
    Link: https://lore.kernel.org/r/20230512131435.205464-3-francesco@dolcini.it
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 94a4fa00c8198cee27c1d97b9e4bc365303cf098
Author: Francesco Dolcini <francesco.dolcini@toradex.com>
Date:   Fri May 12 15:14:34 2023 +0200

    Revert "usb: gadget: udc: core: Prevent redundant calls to pullup"

    commit 5e1617210aede9f1b91bb9819c93097b6da481f9 upstream.

    This reverts commit a3afbf5cc887fc3401f012fe629810998ed61859.

    This depends on commit 0db213ea8eed ("usb: gadget: udc: core: Invoke
    usb_gadget_connect only when started") that introduces a regression,
    revert it till the issue is fixed.

    Cc: stable@vger.kernel.org
    Reported-by: Stephan Gerhold <stephan@gerhold.net>
    Reported-by: Francesco Dolcini <francesco.dolcini@toradex.com>
    Link: https://lore.kernel.org/all/ZF4BvgsOyoKxdPFF@francesco-nb.int.toradex.com/
    Signed-off-by: Francesco Dolcini <francesco.dolcini@toradex.com>
    Link: https://lore.kernel.org/r/20230512131435.205464-2-francesco@dolcini.it
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 54ee23e4ab263a495ace1eed43d3883212ece17f
Author: Badhri Jagan Sridharan <badhri@google.com>
Date:   Mon May 8 21:44:43 2023 +0000

    usb: typec: altmodes/displayport: fix pin_assignment_show
```

```
    commit d8f28269dd4bf9b55c3fb376ae31512730a96fce upstream.

    This patch fixes negative indexing of buf array in pin_assignment_show
    when get_current_pin_assignments returns 0 i.e. no compatible pin
    assignments are found.

    BUG: KASAN: use-after-free in pin_assignment_show+0x26c/0x33c
    ...
    Call trace:
    dump_backtrace+0x110/0x204
    dump_stack_lvl+0x84/0xbc
    print_report+0x358/0x974
    kasan_report+0x9c/0xfc
    __do_kernel_fault+0xd4/0x2d4
    do_bad_area+0x48/0x168
    do_tag_check_fault+0x24/0x38
    do_mem_abort+0x6c/0x14c
    el1_abort+0x44/0x68
    el1h_64_sync_handler+0x64/0xa4
    el1h_64_sync+0x78/0x7c
    pin_assignment_show+0x26c/0x33c
    dev_attr_show+0x50/0xc0

    Fixes: 0e3bb7d6894d ("usb: typec: Add driver for DisplayPort alternate mode")
    Cc: stable@vger.kernel.org
    Signed-off-by: Badhri Jagan Sridharan <badhri@google.com>
    Reviewed-by: Heikki Krogerus <heikki.krogerus@linux.intel.com>
    Link: https://lore.kernel.org/r/20230508214443.893436-1-badhri@google.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f56ba18694b2a9a1bf412f6ef7f7e8a3eefcd721
Author: Konrad Gräfe <k.graefe@gateware.de>
Date:   Fri May 5 16:36:40 2023 +0200

    usb: gadget: u_ether: Fix host MAC address case

    commit 3c0f4f09c063e143822393d99cb2b19a85451c07 upstream.

    The CDC-ECM specification [1] requires to send the host MAC address as
    an uppercase hexadecimal string in chapter "5.4 Ethernet Networking
    Functional Descriptor":
        The Unicode character is chosen from the set of values 30h through
        39h and 41h through 46h (0-9 and A-F).

    However, snprintf(.., "%pm", ..) generates a lowercase MAC address
    string. While most host drivers are tolerant to this, UsbNcm.sys on
    Windows 10 is not. Instead it uses a different MAC address with all
    bytes set to zero including and after the first byte containing a
    lowercase letter. On Windows 11 Microsoft fixed it, but apparently they
    did not backport the fix.

    This change fixes the issue by upper-casing the MAC to comply with the
    specification.

    [1]: https://www.usb.org/document-library/class-definitions-communication-devices-12,
file ECM120.pdf

    Fixes: bcd4a1c40bee ("usb: gadget: u_ether: construct with default values and add
setters/getters")
    Cc: stable@vger.kernel.org
    Signed-off-by: Konrad Gräfe <k.graefe@gateware.de>
    Link: https://lore.kernel.org/r/20230505143640.443014-1-k.graefe@gateware.de
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 7a2feb6f2e6082ff65ae05c2c1f687d6b9fc1ca3
```

```
Author: Udipto Goswami <quic_ugoswami@quicinc.com>
Date:   Tue May 9 20:18:36 2023 +0530

    usb: dwc3: debugfs: Resume dwc3 before accessing registers

    commit 614ce6a2ea50068b45339257891e51e639ac9001 upstream.

    When the dwc3 device is runtime suspended, various required clocks are in
    disabled state and it is not guaranteed that access to any registers would
    work. Depending on the SoC glue, a register read could be as benign as
    returning 0 or be fatal enough to hang the system.

    In order to prevent such scenarios of fatal errors, make sure to resume
    dwc3 then allow the function to proceed.

    Fixes: 72246da40f37 ("usb: Introduce DesignWare USB3 DRD Driver")
    Cc: stable@vger.kernel.org #3.2: 30332eeefec8: debugfs: regset32: Add Runtime PM support
    Signed-off-by: Udipto Goswami <quic_ugoswami@quicinc.com>
    Reviewed-by: Johan Hovold <johan+linaro@kernel.org>
    Tested-by: Johan Hovold <johan+linaro@kernel.org>
    Acked-by: Thinh Nguyen <Thinh.Nguyen@synopsys.com>
    Link: https://lore.kernel.org/r/20230509144836.6803-1-quic_ugoswami@quicinc.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 06684c72b6b153dc434bb6ccebbb49f4cd812b5e
Author: Roger Quadros <rogerq@kernel.org>
Date:   Wed May 3 14:00:48 2023 +0300

    usb: dwc3: gadget: Improve dwc3_gadget_suspend() and dwc3_gadget_resume()

    commit c8540870af4ce6ddeb27a7bb5498b75fb29b643c upstream.

    Prevent -ETIMEDOUT error on .suspend().
    e.g. If gadget driver is loaded and we are connected to a USB host,
    all transfers must be stopped before stopping the controller else
    we will not get a clean stop i.e. dwc3_gadget_run_stop() will take
    several seconds to complete and will return -ETIMEDOUT.

    Handle error cases properly in dwc3_gadget_suspend().
    Simplify dwc3_gadget_resume() by using the introduced helper function.

    Fixes: 9f8a67b65a49 ("usb: dwc3: gadget: fix gadget suspend/resume")
    Cc: stable@vger.kernel.org
    Suggested-by: Thinh Nguyen <Thinh.Nguyen@synopsys.com>
    Signed-off-by: Roger Quadros <rogerq@kernel.org>
    Acked-by: Thinh Nguyen <Thinh.Nguyen@synopsys.com>
    Link: https://lore.kernel.org/r/20230503110048.30617-1-rogerq@kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 0b6bf9dd2b560a53e8b1760baa3d349aadad4596
Author: Weitao Wang <WeitaoWang-oc@zhaoxin.com>
Date:   Sun Apr 23 18:59:52 2023 +0800

    USB: UHCI: adjust zhaoxin UHCI controllers OverCurrent bit value

    commit dddb342b5b9e482bb213aecc08cbdb201ea4f8da upstream.

    OverCurrent condition is not standardized in the UHCI spec.
    Zhaoxin UHCI controllers report OverCurrent bit active off.
    In order to handle OverCurrent condition correctly, the uhci-hcd
    driver needs to be told to expect the active-off behavior.

    Suggested-by: Alan Stern <stern@rowland.harvard.edu>
    Cc: stable@vger.kernel.org
    Signed-off-by: Weitao Wang <WeitaoWang-oc@zhaoxin.com>
```

```
    Acked-by: Alan Stern <stern@rowland.harvard.edu>
    Link: https://lore.kernel.org/r/20230423105952.4526-1-WeitaoWang-oc@zhaoxin.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit d36dcf9dabde02f86a20ead6427e91a344c14cd0
Author: Maxime Bizon <mbizon@freebox.fr>
Date:   Fri May 5 13:47:59 2023 +0200

    usb-storage: fix deadlock when a scsi command timeouts more than once

    commit a398d5eac6984316e71474e25b975688f282379b upstream.

    With faulty usb-storage devices, read/write can timeout, in that case
    the SCSI layer will abort and re-issue the command. USB storage has no
    internal timeout, it relies on SCSI layer aborting commands via
    .eh_abort_handler() for non those responsive devices.

    After two consecutive timeouts of the same command, SCSI layer calls
    .eh_device_reset_handler(), without calling .eh_abort_handler() first.

    With usb-storage, this causes a deadlock:

      -> .eh_device_reset_handler
        -> device_reset
          -> mutex_lock(&(us->dev_mutex));

    mutex already by usb_stor_control_thread(), which is waiting for
    command completion:

      -> usb_stor_control_thread (mutex taken here)
        -> usb_stor_invoke_transport
          -> usb_stor_Bulk_transport
            -> usb_stor_bulk_srb
              -> usb_stor_bulk_transfer_sglist
                -> usb_sg_wait

    Make sure we cancel any pending command in .eh_device_reset_handler()
    to avoid this.

    Signed-off-by: Maxime Bizon <mbizon@freebox.fr>
    Cc: linux-usb@vger.kernel.org
    Cc: stable <stable@kernel.org>
    Link: https://lore.kernel.org/all/ZEllnjMKT8ulZbJh@sakura/
    Reviewed-by: Alan Stern <stern@rowland.harvard.edu>
    Acked-by: Alan Stern <stern@rowland.harvard.edu>
    Link: https://lore.kernel.org/r/20230505114759.1189741-1-mbizon@freebox.fr
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 50775a046c68e1d157d5e413cae2e5e00da1c463
Author: Alan Stern <stern@rowland.harvard.edu>
Date:   Mon May 1 14:22:35 2023 -0400

    USB: usbtmc: Fix direction for 0-length ioctl control messages

    commit 94d25e9128988c6a1fc9070f6e98215a95795bd8 upstream.

    The syzbot fuzzer found a problem in the usbtmc driver: When a user
    submits an ioctl for a 0-length control transfer, the driver does not
    check that the direction is set to OUT:

    ------------[ cut here ]------------
    usb 3-1: BOGUS control dir, pipe 80000b80 doesn't match bRequestType fd
    WARNING: CPU: 0 PID: 5100 at drivers/usb/core/urb.c:411 usb_submit_urb+0x14a7/0x1880
drivers/usb/core/urb.c:411
    Modules linked in:
```

```
    CPU: 0 PID: 5100 Comm: syz-executor428 Not tainted 6.3.0-syzkaller-12049-g58390c8ce1bd
 #0
    Hardware name: Google Google Compute Engine/Google Compute Engine, BIOS Google
 04/14/2023
    RIP: 0010:usb_submit_urb+0x14a7/0x1880 drivers/usb/core/urb.c:411
    Code: 7c 24 40 e8 1b 13 5c fb 48 8b 7c 24 40 e8 21 1d f0 fe 45 89 e8 44 89 f1 4c 89 e2
 48 89 c6 48 c7 c7 e0 b5 fc 8a e8 19 c8 23 fb <0f> 0b e9 9f ee ff ff e8 ed 12 5c fb 0f b6 1d
 12 8a 3c 08 31 ff 41
    RSP: 0018:ffffc90003d2fb00 EFLAGS: 00010282
    RAX: 0000000000000000 RBX: ffff8880789e9058 RCX: 0000000000000000
    RDX: ffff888029593b80 RSI: ffffffff814c1447 RDI: 0000000000000001
    RBP: ffff88801ea742f8 R08: 0000000000000001 R09: 0000000000000000
    R10: 0000000000000001 R11: 0000000000000001 R12: ffff88802915e528
    R13: 00000000000000fd R14: 0000000080000b80 R15: ffff8880222b3100
    FS:  0000555556ca63c0(0000) GS:ffff8880b9800000(0000) knlGS:0000000000000000
    CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    CR2: 00007f9ef4d18150 CR3: 0000000073e5b000 CR4: 00000000003506f0
    DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000
    DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400
    Call Trace:
     <TASK>
     usb_start_wait_urb+0x101/0x4b0 drivers/usb/core/message.c:58
     usb_internal_control_msg drivers/usb/core/message.c:102 [inline]
     usb_control_msg+0x320/0x4a0 drivers/usb/core/message.c:153
     usbtmc_ioctl_request drivers/usb/class/usbtmc.c:1954 [inline]
     usbtmc_ioctl+0x1b3d/0x2840 drivers/usb/class/usbtmc.c:2097

    To fix this, we must override the direction in the bRequestType field
    of the control request structure when the length is 0.

    Reported-and-tested-by: syzbot+ce77725b89b7bd52425c@syzkaller.appspotmail.com
    Signed-off-by: Alan Stern <stern@rowland.harvard.edu>
    Link: https://lore.kernel.org/linux-usb/000000000000716a3705f9adb8ee@google.com/
    CC: <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/ede1ee02-b718-49e7-a44c-51339fec706b@rowland.harvard.edu
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 22e0fba98f43f416b375ce6677b0a800850d18cf
Author: Takashi Iwai <tiwai@suse.de>
Date:   Fri May 12 09:58:58 2023 +0200

    ALSA: usb-audio: Add a sample rate workaround for Line6 Pod Go

    commit 359b4315471181f108723c61612d96e383e56179 upstream.

    Line6 Pod Go (0e41:424b) requires the similar workaround for the fixed
    48k sample rate like other Line6 models.  This patch adds the
    corresponding entry to line6_parse_audio_format_rate_quirk().

    Reported-by: John Humlick <john@humlick.org>
    Cc: <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20230512075858.22813-1-tiwai@suse.de
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit eac08fcd70ccb7a2d6d477f02c714cca07450721
Author: Arnd Bergmann <arnd@arndb.de>
Date:   Tue May 16 21:45:35 2023 +0200

    bridge: always declare tunnel functions

    [ Upstream commit 89dcd87ce534a3a7f267cfd58505803006f51301 ]

    When CONFIG_BRIDGE_VLAN_FILTERING is disabled, two functions are still
    defined but have no prototype or caller. This causes a W=1 warning for
```

```
    the missing prototypes:

    net/bridge/br_netlink_tunnel.c:29:6: error: no previous prototype for
'vlan_tunid_inrange' [-Werror=missing-prototypes]
    net/bridge/br_netlink_tunnel.c:199:5: error: no previous prototype for
'br_vlan_tunnel_info' [-Werror=missing-prototypes]

    The functions are already contitional on CONFIG_BRIDGE_VLAN_FILTERING,
    and I coulnd't easily figure out the right set of #ifdefs, so just
    move the declarations out of the #ifdef to avoid the warning,
    at a small cost in code size over a more elaborate fix.

    Fixes: 188c67dd1906 ("net: bridge: vlan options: add support for tunnel id dumping")
    Fixes: 569da0822808 ("net: bridge: vlan options: add support for tunnel mapping
set/del")
    Signed-off-by: Arnd Bergmann <arnd@arndb.de>
    Acked-by: Nikolay Aleksandrov <razor@blackwall.org>
    Link: https://lore.kernel.org/r/20230516194625.549249-3-arnd@kernel.org
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a337706c1fb35aac3f26b48aca80421bdbe1d33a
Author: Florian Westphal <fw@strlen.de>
Date:   Thu May 11 22:39:30 2023 +0200

    netfilter: nft_set_rbtree: fix null deref on element insertion

    [ Upstream commit 61ae320a29b0540c16931816299eb86bf2b66c08 ]

    There is no guarantee that rb_prev() will not return NULL in nft_rbtree_gc_elem():

    general protection fault, probably for non-canonical address 0xdffffc0000000003: 0000
[#1] PREEMPT SMP KASAN
    KASAN: null-ptr-deref in range [0x0000000000000018-0x000000000000001f]
     nft_add_set_elem+0x14b0/0x2990
      nf_tables_newsetelem+0x528/0xb30

    Furthermore, there is a possible use-after-free while iterating,
    'node' can be free'd so we need to cache the next value to use.

    Fixes: c9e6978e2725 ("netfilter: nft_set_rbtree: Switch to node list walk for overlap
detection")
    Signed-off-by: Florian Westphal <fw@strlen.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 110f8f59a822713ca5b4aa590f035c99c7b4900c
Author: Florian Westphal <fw@strlen.de>
Date:   Thu May 11 14:15:15 2023 +0200

    netfilter: nf_tables: fix nft_trans type confusion

    [ Upstream commit e3c361b8acd636f5fe80c02849ca175201edf10c ]

    nft_trans_FOO objects all share a common nft_trans base structure, but
    trailing fields depend on the real object size. Access is only safe after
    trans->msg_type check.

    Check for rule type first.  Found by code inspection.

    Fixes: 1a94e38d254b ("netfilter: nf_tables: add NFTA_RULE_ID attribute")
    Signed-off-by: Florian Westphal <fw@strlen.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d2e9490b755cb21d0c303122e77ca3d2073b5075
Author: Benjamin Poirier <bpoirier@nvidia.com>
```

```
Date:   Tue May 16 14:49:24 2023 -0400

    net: selftests: Fix optstring

    [ Upstream commit 9ba9485b87ac97fd159abdb4cbd53099bc9f01c6 ]

    The cited commit added a stray colon to the 'v' option. That makes the
    option work incorrectly.

    ex:
    tools/testing/selftests/net# ./fib_nexthops.sh -v
    (should enable verbose mode, instead it shows help text due to missing arg)

    Fixes: 5feba4727395 ("selftests: fib_nexthops: Make ping timeout configurable")
    Reviewed-by: Ido Schimmel <idosch@nvidia.com>
    Signed-off-by: Benjamin Poirier <bpoirier@nvidia.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9fe84a07f15ea58cf2041cb52f494932b9a76c4b
Author: Vladimir Oltean <vladimir.oltean@nxp.com>
Date:   Tue May 16 18:44:10 2023 +0300

    net: pcs: xpcs: fix C73 AN not getting enabled

    [ Upstream commit c46e78ba9a7a09da4f192dc8df15c4e8a07fb9e0 ]

    The XPCS expects clause 73 (copper backplane) autoneg to follow the
    ethtool autoneg bit. It actually did that until the blamed
    commit inaptly replaced state->an_enabled (coming from ethtool) with
    phylink_autoneg_inband() (coming from the device tree or struct
    phylink_config), as part of an unrelated phylink_pcs API conversion.

    Russell King suggests that state->an_enabled from the original code was
    just a proxy for the ethtool Autoneg bit, and that the correct way of
    restoring the functionality is to check for this bit in the advertising
    mask.

    Fixes: 11059740e616 ("net: pcs: xpcs: convert to phylink_pcs_ops")
    Link: https://lore.kernel.org/netdev/ZGNt2MFeRolKGFck@shell.armlinux.org.uk/
    Suggested-by: Russell King (Oracle) <linux@armlinux.org.uk>
    Signed-off-by: Vladimir Oltean <vladimir.oltean@nxp.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 862c6e3e26735247d8a4df41fa2421909c3f4d63
Author: M Chetan Kumar <m.chetan.kumar@linux.intel.com>
Date:   Tue May 16 21:09:46 2023 +0530

    net: wwan: iosm: fix NULL pointer dereference when removing device

    [ Upstream commit 60829145f1e2650b31ebe6a0ec70a9725b38fa2c ]

    In suspend and resume cycle, the removal and rescan of device ends
    up in NULL pointer dereference.

    During driver initialization, if the ipc_imem_wwan_channel_init()
    fails to get the valid device capabilities it returns an error and
    further no resource (wwan struct) will be allocated. Now in this
    situation if driver removal procedure is initiated it would result
    in NULL pointer exception since unallocated wwan struct is dereferenced
    inside ipc_wwan_deinit().

    ipc_imem_run_state_worker() to handle the called functions return value
    and to release the resource in failure case. It also reports the link
```

```
    down event in failure cases. The user space application can handle this
    event to do a device reset for restoring the device communication.

    Fixes: 3670970dd8c6 ("net: iosm: shared memory IPC interface")
    Reported-by: Samuel Wein PhD <sam@samwein.com>
    Closes: https://lore.kernel.org/netdev/20230427140819.1310f4bd@kernel.org/T/
    Signed-off-by: M Chetan Kumar <m.chetan.kumar@linux.intel.com>
    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 85c35386bc24eb501a175be95cb9a82941ec1823
Author: Eric Dumazet <edumazet@google.com>
Date:   Tue May 16 14:23:42 2023 +0000

    vlan: fix a potential uninit-value in vlan_dev_hard_start_xmit()

    [ Upstream commit dacab578c7c6cd06c50c89dfa36b0e0f10decd4e ]

    syzbot triggered the following splat [1], sending an empty message
    through pppoe_sendmsg().

    When VLAN_FLAG_REORDER_HDR flag is set, vlan_dev_hard_header()
    does not push extra bytes for the VLAN header, because vlan is offloaded.

    Unfortunately vlan_dev_hard_start_xmit() first reads veth->h_vlan_proto
    before testing (vlan->flags & VLAN_FLAG_REORDER_HDR).

    We need to swap the two conditions.

    [1]
    BUG: KMSAN: uninit-value in vlan_dev_hard_start_xmit+0x171/0x7f0
net/8021q/vlan_dev.c:111
    vlan_dev_hard_start_xmit+0x171/0x7f0 net/8021q/vlan_dev.c:111
    __netdev_start_xmit include/linux/netdevice.h:4883 [inline]
    netdev_start_xmit include/linux/netdevice.h:4897 [inline]
    xmit_one net/core/dev.c:3580 [inline]
    dev_hard_start_xmit+0x253/0xa20 net/core/dev.c:3596
    __dev_queue_xmit+0x3c7f/0x5ac0 net/core/dev.c:4246
    dev_queue_xmit include/linux/netdevice.h:3053 [inline]
    pppoe_sendmsg+0xa93/0xb80 drivers/net/ppp/pppoe.c:900
    sock_sendmsg_nosec net/socket.c:724 [inline]
    sock_sendmsg net/socket.c:747 [inline]
    ____sys_sendmsg+0xa24/0xe40 net/socket.c:2501
    ___sys_sendmsg+0x2a1/0x3f0 net/socket.c:2555
    __sys_sendmmsg+0x411/0xa50 net/socket.c:2641
    __do_sys_sendmmsg net/socket.c:2670 [inline]
    __se_sys_sendmmsg net/socket.c:2667 [inline]
    __x64_sys_sendmmsg+0xbc/0x120 net/socket.c:2667
    do_syscall_x64 arch/x86/entry/common.c:50 [inline]
    do_syscall_64+0x41/0xc0 arch/x86/entry/common.c:80
    entry_SYSCALL_64_after_hwframe+0x63/0xcd

    Uninit was created at:
    slab_post_alloc_hook+0x12d/0xb60 mm/slab.h:774
    slab_alloc_node mm/slub.c:3452 [inline]
    kmem_cache_alloc_node+0x543/0xab0 mm/slub.c:3497
    kmalloc_reserve+0x148/0x470 net/core/skbuff.c:520
    __alloc_skb+0x3a7/0x850 net/core/skbuff.c:606
    alloc_skb include/linux/skbuff.h:1277 [inline]
    sock_wmalloc+0xfe/0x1a0 net/core/sock.c:2583
    pppoe_sendmsg+0x3af/0xb80 drivers/net/ppp/pppoe.c:867
    sock_sendmsg_nosec net/socket.c:724 [inline]
    sock_sendmsg net/socket.c:747 [inline]
    ____sys_sendmsg+0xa24/0xe40 net/socket.c:2501
```

```
     ___sys_sendmsg+0x2a1/0x3f0 net/socket.c:2555
     __sys_sendmmsg+0x411/0xa50 net/socket.c:2641
     __do_sys_sendmmsg net/socket.c:2670 [inline]
     __se_sys_sendmmsg net/socket.c:2667 [inline]
     __x64_sys_sendmmsg+0xbc/0x120 net/socket.c:2667
     do_syscall_x64 arch/x86/entry/common.c:50 [inline]
     do_syscall_64+0x41/0xc0 arch/x86/entry/common.c:80
     entry_SYSCALL_64_after_hwframe+0x63/0xcd

     CPU: 0 PID: 29770 Comm: syz-executor.0 Not tainted 6.3.0-rc6-syzkaller-gc478e5b17829 #0
     Hardware name: Google Google Compute Engine/Google Compute Engine, BIOS Google
 03/30/2023

     Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
     Reported-by: syzbot <syzkaller@googlegroups.com>
     Signed-off-by: Eric Dumazet <edumazet@google.com>
     Signed-off-by: David S. Miller <davem@davemloft.net>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit 7d1bd401481344f8cef96068fdf5f9e00bff6af5
 Author: Aleksandr Loktionov <aleksandr.loktionov@intel.com>
 Date:   Tue May 16 10:41:46 2023 -0700

     igb: fix bit_shift to be in [1..8] range

     [ Upstream commit 60d758659f1fb49e0d5b6ac2691ede8c0958795b ]

     In igb_hash_mc_addr() the expression:
             "mc_addr[4] >> 8 - bit_shift", right shifting "mc_addr[4]"
     shift by more than 7 bits always yields zero, so hash becomes not so different.
     Add initialization with bit_shift = 1 and add a loop condition to ensure
     bit_shift will be always in [1..8] range.

     Fixes: 9d5c824399de ("igb: PCI-Express 82575 Gigabit Ethernet driver")
     Signed-off-by: Aleksandr Loktionov <aleksandr.loktionov@intel.com>
     Tested-by: Pucha Himasekhar Reddy <himasekharx.reddy.pucha@intel.com> (A Contingent
 worker at Intel)
     Signed-off-by: Tony Nguyen <anthony.l.nguyen@intel.com>
     Signed-off-by: David S. Miller <davem@davemloft.net>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit d8c20f0625732815038ca6be6b938a9406fc3c56
 Author: Marco Migliore <m.migliore@tiesse.com>
 Date:   Tue May 16 09:38:54 2023 +0200

     net: dsa: mv88e6xxx: Fix mv88e6393x EPC write command offset

     [ Upstream commit 1323e0c6e1d7e103d59384c3ac50f72b17a6936c ]

     According to datasheet, the command opcode must be specified
     into bits [14:12] of the Extended Port Control register (EPC).

     Fixes: de776d0d316f ("net: dsa: mv88e6xxx: add support for mv88e6393x family")
     Signed-off-by: Marco Migliore <m.migliore@tiesse.com>
     Reviewed-by: Andrew Lunn <andrew@lunn.ch>
     Signed-off-by: David S. Miller <davem@davemloft.net>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit b8b1a667744741fa7807b09a12797a27f14f3fac
 Author: Christophe JAILLET <christophe.jaillet@wanadoo.fr>
 Date:   Mon May 15 21:09:11 2023 +0200

     cassini: Fix a memory leak in the error handling path of cas_init_one()

     [ Upstream commit 412cd77a2c24b191c65ea53025222418db09817c ]
```

```
    cas_saturn_firmware_init() allocates some memory using vmalloc(). This
    memory is freed in the .remove() function but not it the error handling
    path of the probe.

    Add the missing vfree() to avoid a memory leak, should an error occur.

    Fixes: fcaa40669cd7 ("cassini: use request_firmware")
    Signed-off-by: Christophe JAILLET <christophe.jaillet@wanadoo.fr>
    Reviewed-by: Pavan Chebbi <pavan.chebbi@broadcom.com>
    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 0d20210a190f76db9ec35ee4e0fc77e6c7a148f5
Author: Kuniyuki Iwashima <kuniyu@amazon.com>
Date:   Mon May 15 11:42:04 2023 -0700

    tun: Fix memory leak for detached NAPI queue.

    [ Upstream commit 82b2bc279467c875ec36f8ef820f00997c2a4e8e ]

    syzkaller reported [0] memory leaks of sk and skb related to the TUN
    device with no repro, but we can reproduce it easily with:

      struct ifreq ifr = {}
      int fd_tun, fd_tmp;
      char buf[4] = {};

      fd_tun = openat(AT_FDCWD, "/dev/net/tun", O_WRONLY, 0);
      ifr.ifr_flags = IFF_TUN | IFF_NAPI | IFF_MULTI_QUEUE;
      ioctl(fd_tun, TUNSETIFF, &ifr);

      ifr.ifr_flags = IFF_DETACH_QUEUE;
      ioctl(fd_tun, TUNSETQUEUE, &ifr);

      fd_tmp = socket(AF_PACKET, SOCK_PACKET, 0);
      ifr.ifr_flags = IFF_UP;
      ioctl(fd_tmp, SIOCSIFFLAGS, &ifr);

      write(fd_tun, buf, sizeof(buf));
      close(fd_tun);

    If we enable NAPI and multi-queue on a TUN device, we can put skb into
    tfile->sk.sk_write_queue after the queue is detached.  We should prevent
    it by checking tfile->detached before queuing skb.

    Note this must be done under tfile->sk.sk_write_queue.lock because write()
    and ioctl(IFF_DETACH_QUEUE) can run concurrently.  Otherwise, there would
    be a small race window:

      write()                               ioctl(IFF_DETACH_QUEUE)
      `- tun_get_user                       `- __tun_detach
         |- if (tfile->detached)                |- tun_disable_queue
         |  `-> false                           |  `- tfile->detached = tun
         |                                      `- tun_queue_purge
         |- spin_lock_bh(&queue->lock)
         `- __skb_queue_tail(queue, skb)

    Another solution is to call tun_queue_purge() when closing and
    reattaching the detached queue, but it could paper over another
    problems.  Also, we do the same kind of test for IFF_NAPI_FRAGS.

    [0]:
    unreferenced object 0xffff88801edbc800 (size 2048):
```

```
        comm "syz-executor.1", pid 33269, jiffies 4295743834 (age 18.756s)
        hex dump (first 32 bytes):
          00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
          00 00 07 40 00 00 00 00 00 00 00 00 00 00 00 00  ...@............
        backtrace:
          [<000000008c16ea3d>] __do_kmalloc_node mm/slab_common.c:965 [inline]
          [<000000008c16ea3d>] __kmalloc+0x4a/0x130 mm/slab_common.c:979
          [<000000003addde56>] kmalloc include/linux/slab.h:563 [inline]
          [<000000003addde56>] sk_prot_alloc+0xef/0x1b0 net/core/sock.c:2035
          [<000000003e20621f>] sk_alloc+0x36/0x2f0 net/core/sock.c:2088
          [<0000000028e43843>] tun_chr_open+0x3d/0x190 drivers/net/tun.c:3438
          [<000000001b0f1f28>] misc_open+0x1a6/0x1f0 drivers/char/misc.c:165
          [<000000004376f706>] chrdev_open+0x111/0x300 fs/char_dev.c:414
          [<00000000614d379f>] do_dentry_open+0x2f9/0x750 fs/open.c:920
          [<000000008eb24774>] do_open fs/namei.c:3636 [inline]
          [<000000008eb24774>] path_openat+0x143f/0x1a30 fs/namei.c:3791
          [<00000000955077b5>] do_filp_open+0xce/0x1c0 fs/namei.c:3818
          [<00000000b78973b0>] do_sys_openat2+0xf0/0x260 fs/open.c:1356
          [<00000000057be699>] do_sys_open fs/open.c:1372 [inline]
          [<00000000057be699>] __do_sys_openat fs/open.c:1388 [inline]
          [<00000000057be699>] __se_sys_openat fs/open.c:1383 [inline]
          [<00000000057be699>] __x64_sys_openat+0x83/0xf0 fs/open.c:1383
          [<00000000a7d2182d>] do_syscall_x64 arch/x86/entry/common.c:50 [inline]
          [<00000000a7d2182d>] do_syscall_64+0x3c/0x90 arch/x86/entry/common.c:80
          [<000000004cc4e8c4>] entry_SYSCALL_64_after_hwframe+0x72/0xdc

      unreferenced object 0xffff88802f671700 (size 240):
        comm "syz-executor.1", pid 33269, jiffies 4295743854 (age 18.736s)
        hex dump (first 32 bytes):
          68 c9 db 1e 80 88 ff ff 68 c9 db 1e 80 88 ff ff  h.......h.......
          00 c0 7b 2f 80 88 ff ff 00 c8 db 1e 80 88 ff ff  ..{/............
        backtrace:
          [<00000000e9d9fdb6>] __alloc_skb+0x223/0x250 net/core/skbuff.c:644
          [<000000002c3e4e0b>] alloc_skb include/linux/skbuff.h:1288 [inline]
          [<000000002c3e4e0b>] alloc_skb_with_frags+0x6f/0x350 net/core/skbuff.c:6378
          [<00000000825f98d7>] sock_alloc_send_pskb+0x3ac/0x3e0 net/core/sock.c:2729
          [<00000000e9eb3df3>] tun_alloc_skb drivers/net/tun.c:1529 [inline]
          [<00000000e9eb3df3>] tun_get_user+0x5e1/0x1f90 drivers/net/tun.c:1841
          [<0000000053096912>] tun_chr_write_iter+0xac/0x120 drivers/net/tun.c:2035
          [<00000000b9282ae0>] call_write_iter include/linux/fs.h:1868 [inline]
          [<00000000b9282ae0>] new_sync_write fs/read_write.c:491 [inline]
          [<00000000b9282ae0>] vfs_write+0x40f/0x530 fs/read_write.c:584
          [<00000000524566e4>] ksys_write+0xa1/0x170 fs/read_write.c:637
          [<00000000a7d2182d>] do_syscall_x64 arch/x86/entry/common.c:50 [inline]
          [<00000000a7d2182d>] do_syscall_64+0x3c/0x90 arch/x86/entry/common.c:80
          [<000000004cc4e8c4>] entry_SYSCALL_64_after_hwframe+0x72/0xdc

      Fixes: cde8b15f1aab ("tuntap: add ioctl to attach or detach a file form tuntap device")
      Reported-by: syzkaller <syzkaller@googlegroups.com>
      Signed-off-by: Kuniyuki Iwashima <kuniyu@amazon.com>
      Signed-off-by: David S. Miller <davem@davemloft.net>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 450f79aad80be84b7e37558f8242a88af8cd2c30
  Author: Ido Schimmel <idosch@nvidia.com>
  Date:   Mon May 15 19:29:25 2023 +0300

      devlink: Fix crash with CONFIG_NET_NS=n

      [ Upstream commit d6352dae0903fe8beae4c007dc320e9e9f1fed45 ]

      '__net_initdata' becomes a no-op with CONFIG_NET_NS=y, but when this
      option is disabled it becomes '__initdata', which means the data can be
      freed after the initialization phase. This annotation is obviously
      incorrect for the devlink net device notifier block which is still
```

```
    registered after the initialization phase [1].

    Fix this crash by removing the '__net_initdata' annotation.

    [1]
    general protection fault, probably for non-canonical address 0xcccccccccccccccc: 0000
[#1] PREEMPT SMP
    CPU: 3 PID: 117 Comm: (udev-worker) Not tainted 6.4.0-rc1-custom-gdf0acdc59b09 #64
    Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS 1.16.2-1.fc37 04/01/2014
    RIP: 0010:notifier_call_chain+0x58/0xc0
    [...]
    Call Trace:
     <TASK>
     dev_set_mac_address+0x85/0x120
     dev_set_mac_address_user+0x30/0x50
     do_setlink+0x219/0x1270
     rtnl_setlink+0xf7/0x1a0
     rtnetlink_rcv_msg+0x142/0x390
     netlink_rcv_skb+0x58/0x100
     netlink_unicast+0x188/0x270
     netlink_sendmsg+0x214/0x470
     __sys_sendto+0x12f/0x1a0
     __x64_sys_sendto+0x24/0x30
     do_syscall_64+0x38/0x80
     entry_SYSCALL_64_after_hwframe+0x63/0xcd

    Fixes: e93c9378e33f ("devlink: change per-devlink netdev notifier to static one")
    Reported-by: Marek Szyprowski <m.szyprowski@samsung.com>
    Closes: https://lore.kernel.org/netdev/600ddf9e-589a-2aa0-7b69-a438f833ca10@samsung.com/
    Tested-by: Marek Szyprowski <m.szyprowski@samsung.com>
    Signed-off-by: Ido Schimmel <idosch@nvidia.com>
    Reviewed-by: Jiri Pirko <jiri@nvidia.com>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Link: https://lore.kernel.org/r/20230515162925.1144416-1-idosch@nvidia.com
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dd30e2172d081506041a0354ed952c3d61659e04
Author: Michael Kelley <mikelley@microsoft.com>
Date:   Mon May 15 10:20:41 2023 -0700

    scsi: storvsc: Don't pass unused PFNs to Hyper-V host

    [ Upstream commit 4e81a6cba517cb33584308a331f14f5e3fec369b ]

    In a SCSI request, storvsc pre-allocates space for up to
    MAX_PAGE_BUFFER_COUNT physical frame numbers to be passed to Hyper-V.  If
    the size of the I/O request requires more PFNs, a separate memory area of
    exactly the correct size is dynamically allocated.

    But when the pre-allocated area is used, current code always passes
    MAX_PAGE_BUFFER_COUNT PFNs to Hyper-V, even if fewer are needed.  While
    this doesn't break anything because the additional PFNs are always zero,
    more bytes than necessary are copied into the VMBus channel ring buffer.
    This takes CPU cycles and wastes space in the ring buffer. For a typical 4
    Kbyte I/O that requires only a single PFN, 248 unnecessary bytes are
    copied.

    Fix this by setting the payload_sz based on the actual number of PFNs
    required, not the size of the pre-allocated space.

    Reported-by: John Starks <jostarks@microsoft.com>
    Fixes: 8f43710543ef ("scsi: storvsc: Support PAGE_SIZE larger than 4K")
    Signed-off-by: Michael Kelley <mikelley@microsoft.com>
    Link: https://lore.kernel.org/r/1684171241-16209-1-git-send-email-mikelley@microsoft.com
```

```
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c176f03350954b795322de0bfe1d7b514db41f45
Author: Johannes Berg <johannes.berg@intel.com>
Date:   Sun May 14 12:15:53 2023 +0300

    wifi: iwlwifi: mvm: don't trust firmware n_channels

    [ Upstream commit 682b6dc29d98e857e6ca4bbc077c7dc2899b7473 ]

    If the firmware sends us a corrupted MCC response with
    n_channels much larger than the command response can be,
    we might copy far too much (uninitialized) memory and
    even crash if the n_channels is large enough to make it
    run out of the one page allocated for the FW response.

    Fix that by checking the lengths. Doing a < comparison
    would be sufficient, but the firmware should be doing
    it correctly, so check more strictly.

    Fixes: dcaf9f5ecb6f ("iwlwifi: mvm: add MCC update FW API")
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link:
https://lore.kernel.org/r/20230514120631.d7b233139eb4.I51fd319df8e9d41881fc8450e83d78049518a
79a@changeid
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3b7cd9b5ade5be8be47e7aa4d51c7623119a2e2d
Author: Alon Giladi <alon.giladi@intel.com>
Date:   Sun May 14 12:15:52 2023 +0300

    wifi: iwlwifi: mvm: fix OEM's name in the tas approved list

    [ Upstream commit d0246a0e49efee0f8649d0e4f2350614cdfe6565 ]

    Fix a spelling mistake.

    Fixes: 2856f623ce48 ("iwlwifi: mvm: Add list of OEMs allowed to use TAS")
    Signed-off-by: Alon Giladi <alon.giladi@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link:
https://lore.kernel.org/r/20230514120631.4090de6d1878.If9391ef6da78f1b2cc5eb6cb8f6965816bb7a
7f5@changeid
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 643aa17418ca875f0ae1396a26a822af6d3b20f3
Author: Alon Giladi <alon.giladi@intel.com>
Date:   Sun May 14 12:15:51 2023 +0300

    wifi: iwlwifi: fix OEM's name in the ppag approved list

    [ Upstream commit eca7296d9a671e9961834d2ace9cc0ce21fc15b3 ]

    Fix a spelling mistake.

    Fixes: e8e10a37c51c ("iwlwifi: acpi: move ppag code from mvm to fw/acpi")
    Signed-off-by: Alon Giladi <alon.giladi@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link:
https://lore.kernel.org/r/20230514120631.fdd07f36a8bf.I223e5fb16ab5c95d504c3fdaffd0bd70affad
1c2@changeid
```

```
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7bb1ebccfcb2513bc7a61d6c1fc322688efa4e71
Author: Johannes Berg <johannes.berg@intel.com>
Date:   Sun May 14 12:15:48 2023 +0300

    wifi: iwlwifi: fw: fix DBGI dump

    [ Upstream commit d3ae69180bbd74bcbc03a2b6d10ed7eccbe98c23 ]

    The DBGI dump is (unsurprisingly) of type DBGI, not SRAM.
    This leads to bad register accesses because the union is
    built differently, there's no allocation ID, and thus the
    allocation ID ends up being 0x8000.

    Note that this was already wrong for DRAM vs. SMEM since
    they use different parts of the union, but the allocation
    ID is at the same place, so it worked.

    Fix all of this but set the allocation ID in a way that
    the offset calculation ends up without any offset.

    Fixes: 34bc27783a31 ("iwlwifi: yoyo: fix DBGI_SRAM ini dump header.")
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link:
https://lore.kernel.org/r/20230514120631.19a302ae4c65.I12272599f7c1930666157b9d5e7f81fe9ec4c
421@changeid
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 22a7ee202b3af3f06528a3354d68345eb7cb04e1
Author: Johannes Berg <johannes.berg@intel.com>
Date:   Sun May 14 12:15:46 2023 +0300

    wifi: iwlwifi: mvm: fix cancel_delayed_work_sync() deadlock

    [ Upstream commit c2d8b7f257b2398f2d866205365895e038beca12 ]

    Lockdep points out that we can deadlock here by calling
    cancel_delayed_work_sync() because that might be already
    running and gotten interrupted by the NAPI soft-IRQ.
    Even just calling something that can sleep is wrong in
    this context though.

    Luckily, it doesn't even really matter since the things
    we need to do are idempotent, so just drop the _sync().

    Fixes: e5d153ec54f0 ("iwlwifi: mvm: fix CSA AP side")
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link:
https://lore.kernel.org/r/20230514120631.b1813c823b4d.I9d20cc06d24fa40b6774d3dd95ea5e2bf8dd0
15b@changeid
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 58945b97a3764503a8e6a0a7dabc7aecada47579
Author: Michael Lee <michael-cy.lee@mediatek.com>
Date:   Thu May 4 16:04:41 2023 +0800

    wifi: mac80211: Abort running color change when stopping the AP

    [ Upstream commit a23d7f5b2fbda114de60c4b53311e052281d7533 ]
```

```
    When stopping the AP, there might be a color change in progress. It
    should be deactivated here, or the driver might later finalize a color
    change on a stopped AP.

    Fixes: 5f9404abdf2a (mac80211: add support for BSS color change)
    Signed-off-by: Michael Lee <michael-cy.lee@mediatek.com>
    Link: https://lore.kernel.org/r/20230504080441.22958-1-michael-cy.lee@mediatek.com
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 684a15e5c0f8fd54ad1ac28ecbd024b0d5c6906e
Author: Johannes Berg <johannes.berg@intel.com>
Date:   Thu May 4 16:45:01 2023 +0300

    wifi: mac80211: fix min center freq offset tracing

    [ Upstream commit 248e4776514bf70236e6b1a54c65aa5324c8b1eb ]

    We need to set the correct trace variable, otherwise we're
    overwriting something else instead and the right one that
    we print later is not initialized.

    Fixes: b6011960f392 ("mac80211: handle channel frequency offset")
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link: https://lore.kernel.org/r/20230504134511.828474-2-gregory.greenman@intel.com
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5c9d1a46f79c8e75c416f8faa68e4a573f454e53
Author: Christophe JAILLET <christophe.jaillet@wanadoo.fr>
Date:   Mon Apr 24 19:42:04 2023 +0200

    wifi: mac80211: Fix puncturing bitmap handling in __ieee80211_csa_finalize()

    [ Upstream commit 13ad2b1eeacd48ec0f31f55964e6dc7dfc2c0299 ]

    'changed' can be OR'ed with BSS_CHANGED_EHT_PUNCTURING which is larger than
    an u32.
    So, turn 'changed' into an u64 and update ieee80211_set_after_csa_beacon()
    accordingly.

    In the commit in Fixes, only ieee80211_start_ap() was updated.

    Fixes: 2cc25e4b2a04 ("wifi: mac80211: configure puncturing bitmap")
    Signed-off-by: Christophe JAILLET <christophe.jaillet@wanadoo.fr>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Link:
https://lore.kernel.org/r/e84a3f80fe536787f7a2c7180507efc36cd14f95.1682358088.git.christophe
.jaillet@wanadoo.fr
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6df3eafa31b3ee4f0cba601ca857019964355034
Author: Mirsad Goran Todorovac <mirsad.todorovac@alu.unizg.hr>
Date:   Tue Apr 25 18:40:08 2023 +0200

    wifi: mac80211: fortify the spinlock against deadlock by interrupt

    [ Upstream commit ef6e1997da63ad0ac3fe33153fec9524c9ae56c9 ]

    In the function ieee80211_tx_dequeue() there is a particular locking
    sequence:
```

```
    begin:
            spin_lock(&local->queue_stop_reason_lock);
            q_stopped = local->queue_stop_reasons[q];
            spin_unlock(&local->queue_stop_reason_lock);

    However small the chance (increased by ftracetest), an asynchronous
    interrupt can occur in between of spin_lock() and spin_unlock(),
    and the interrupt routine will attempt to lock the same
    &local->queue_stop_reason_lock again.

    This will cause a costly reset of the CPU and the wifi device or an
    altogether hang in the single CPU and single core scenario.

    The only remaining spin_lock(&local->queue_stop_reason_lock) that
    did not disable interrupts was patched, which should prevent any
    deadlocks on the same CPU/core and the same wifi device.

    This is the probable trace of the deadlock:

    kernel: ================================
    kernel: WARNING: inconsistent lock state
    kernel: 6.3.0-rc6-mt-20230401-00001-gf86822a1170f #4 Tainted: G        W
    kernel: --------------------------------
    kernel: inconsistent {IN-SOFTIRQ-W} -> {SOFTIRQ-ON-W} usage.
    kernel: kworker/5:0/25656 [HC0[0]:SC0[0]:HE1:SE1] takes:
    kernel: ffff9d6190779478 (&local->queue_stop_reason_lock){+.?.}-{2:2}, at:
 return_to_handler+0x0/0x40
    kernel: {IN-SOFTIRQ-W} state was registered at:
    kernel:   lock_acquire+0xc7/0x2d0
    kernel:   _raw_spin_lock+0x36/0x50
    kernel:   ieee80211_tx_dequeue+0xb4/0x1330 [mac80211]
    kernel:   iwl_mvm_mac_itxq_xmit+0xae/0x210 [iwlmvm]
    kernel:   iwl_mvm_mac_wake_tx_queue+0x2d/0xd0 [iwlmvm]
    kernel:   ieee80211_queue_skb+0x450/0x730 [mac80211]
    kernel:   __ieee80211_xmit_fast.constprop.66+0x834/0xa50 [mac80211]
    kernel:   __ieee80211_subif_start_xmit+0x217/0x530 [mac80211]
    kernel:   ieee80211_subif_start_xmit+0x60/0x580 [mac80211]
    kernel:   dev_hard_start_xmit+0xb5/0x260
    kernel:   __dev_queue_xmit+0xdbe/0x1200
    kernel:   neigh_resolve_output+0x166/0x260
    kernel:   ip_finish_output2+0x216/0xb80
    kernel:   __ip_finish_output+0x2a4/0x4d0
    kernel:   ip_finish_output+0x2d/0xd0
    kernel:   ip_output+0x82/0x2b0
    kernel:   ip_local_out+0xec/0x110
    kernel:   igmpv3_sendpack+0x5c/0x90
    kernel:   igmp_ifc_timer_expire+0x26e/0x4e0
    kernel:   call_timer_fn+0xa5/0x230
    kernel:   run_timer_softirq+0x27f/0x550
    kernel:   __do_softirq+0xb4/0x3a4
    kernel:   irq_exit_rcu+0x9b/0xc0
    kernel:   sysvec_apic_timer_interrupt+0x80/0xa0
    kernel:   asm_sysvec_apic_timer_interrupt+0x1f/0x30
    kernel:   _raw_spin_unlock_irqrestore+0x3f/0x70
    kernel:   free_to_partial_list+0x3d6/0x590
    kernel:   __slab_free+0x1b7/0x310
    kernel:   kmem_cache_free+0x52d/0x550
    kernel:   putname+0x5d/0x70
    kernel:   do_sys_openat2+0x1d7/0x310
    kernel:   do_sys_open+0x51/0x80
    kernel:   __x64_sys_openat+0x24/0x30
    kernel:   do_syscall_64+0x5c/0x90
    kernel:   entry_SYSCALL_64_after_hwframe+0x72/0xdc
    kernel: irq event stamp: 5120729
    kernel: hardirqs last  enabled at (5120729): [<ffffffff9d149936>]
```

```
  trace_graph_return+0xd6/0x120
      kernel: hardirqs last disabled at (5120728): [<ffffffff9d149950>]
  trace_graph_return+0xf0/0x120
      kernel: softirqs last  enabled at (5069900): [<ffffffff9cf65b60>]
  return_to_handler+0x0/0x40
      kernel: softirqs last disabled at (5067555): [<ffffffff9cf65b60>]
  return_to_handler+0x0/0x40
      kernel:
              other info that might help us debug this:
      kernel:  Possible unsafe locking scenario:
      kernel:        CPU0
      kernel:        ----
      kernel:   lock(&local->queue_stop_reason_lock);
      kernel:   <Interrupt>
      kernel:     lock(&local->queue_stop_reason_lock);
      kernel:
               *** DEADLOCK ***
      kernel: 8 locks held by kworker/5:0/25656:
      kernel:  #0: ffff9d618009d138 ((wq_completion)events_freezable){+.+.}-{0:0}, at:
  process_one_work+0x1ca/0x530
      kernel:  #1: ffffb1ef4637fe68 ((work_completion)(&local->restart_work)){+.+.}-{0:0}, at:
  process_one_work+0x1ce/0x530
      kernel:  #2: ffffffff9f166548 (rtnl_mutex){+.+.}-{3:3}, at: return_to_handler+0x0/0x40
      kernel:  #3: ffff9d6190778728 (&rdev->wiphy.mtx){+.+.}-{3:3}, at:
  return_to_handler+0x0/0x40
      kernel:  #4: ffff9d619077b480 (&mvm->mutex){+.+.}-{3:3}, at: return_to_handler+0x0/0x40
      kernel:  #5: ffff9d61907bacd8 (&trans_pcie->mutex){+.+.}-{3:3}, at:
  return_to_handler+0x0/0x40
      kernel:  #6: ffffffff9ef9cda0 (rcu_read_lock){....}-{1:2}, at:
  iwl_mvm_queue_state_change+0x59/0x3a0 [iwlmvm]
      kernel:  #7: ffffffff9ef9cda0 (rcu_read_lock){....}-{1:2}, at:
  iwl_mvm_mac_itxq_xmit+0x42/0x210 [iwlmvm]
      kernel:
              stack backtrace:
      kernel: CPU: 5 PID: 25656 Comm: kworker/5:0 Tainted: G        W          6.3.0-rc6-mt-
  20230401-00001-gf86822a1170f #4
      kernel: Hardware name: LENOVO 82H8/LNVNB161216, BIOS GGCN51WW 11/16/2022
      kernel: Workqueue: events_freezable ieee80211_restart_work [mac80211]
      kernel: Call Trace:
      kernel:  <TASK>
      kernel:  ? ftrace_regs_caller_end+0x66/0x66
      kernel:  dump_stack_lvl+0x5f/0xa0
      kernel:  dump_stack+0x14/0x20
      kernel:  print_usage_bug.part.46+0x208/0x2a0
      kernel:  mark_lock.part.47+0x605/0x630
      kernel:  ? sched_clock+0xd/0x20
      kernel:  ? trace_clock_local+0x14/0x30
      kernel:  ? __rb_reserve_next+0x5f/0x490
      kernel:  ? _raw_spin_lock+0x1b/0x50
      kernel:  __lock_acquire+0x464/0x1990
      kernel:  ? mark_held_locks+0x4e/0x80
      kernel:  lock_acquire+0xc7/0x2d0
      kernel:  ? ftrace_regs_caller_end+0x66/0x66
      kernel:  ? ftrace_return_to_handler+0x8b/0x100
      kernel:  ? preempt_count_add+0x4/0x70
      kernel:  _raw_spin_lock+0x36/0x50
      kernel:  ? ftrace_regs_caller_end+0x66/0x66
      kernel:  ? ftrace_regs_caller_end+0x66/0x66
      kernel:  ieee80211_tx_dequeue+0xb4/0x1330 [mac80211]
      kernel:  ? prepare_ftrace_return+0xc5/0x190
      kernel:  ? ftrace_graph_func+0x16/0x20
      kernel:  ? 0xffffffffc02ab0b1
      kernel:  ? lock_acquire+0xc7/0x2d0
      kernel:  ? iwl_mvm_mac_itxq_xmit+0x42/0x210 [iwlmvm]
      kernel:  ? ieee80211_tx_dequeue+0x9/0x1330 [mac80211]
```

```
      kernel:  ? __rcu_read_lock+0x4/0x40
      kernel:  ? ftrace_regs_caller_end+0x66/0x66
      kernel:  iwl_mvm_mac_itxq_xmit+0xae/0x210 [iwlmvm]
      kernel:  ? ftrace_regs_caller_end+0x66/0x66
      kernel:  iwl_mvm_queue_state_change+0x311/0x3a0 [iwlmvm]
      kernel:  ? ftrace_regs_caller_end+0x66/0x66
      kernel:  iwl_mvm_wake_sw_queue+0x17/0x20 [iwlmvm]
      kernel:  ? ftrace_regs_caller_end+0x66/0x66
      kernel:  iwl_txq_gen2_unmap+0x1c9/0x1f0 [iwlwifi]
      kernel:  ? ftrace_regs_caller_end+0x66/0x66
      kernel:  iwl_txq_gen2_free+0x55/0x130 [iwlwifi]
      kernel:  ? ftrace_regs_caller_end+0x66/0x66
      kernel:  iwl_txq_gen2_tx_free+0x63/0x80 [iwlwifi]
      kernel:  ? ftrace_regs_caller_end+0x66/0x66
      kernel:  _iwl_trans_pcie_gen2_stop_device+0x3f3/0x5b0 [iwlwifi]
      kernel:  ? _iwl_trans_pcie_gen2_stop_device+0x9/0x5b0 [iwlwifi]
      kernel:  ? mutex_lock_nested+0x4/0x30
      kernel:  ? ftrace_regs_caller_end+0x66/0x66
      kernel:  iwl_trans_pcie_gen2_stop_device+0x5f/0x90 [iwlwifi]
      kernel:  ? ftrace_regs_caller_end+0x66/0x66
      kernel:  iwl_mvm_stop_device+0x78/0xd0 [iwlmvm]
      kernel:  ? ftrace_regs_caller_end+0x66/0x66
      kernel:  __iwl_mvm_mac_start+0x114/0x210 [iwlmvm]
      kernel:  ? ftrace_regs_caller_end+0x66/0x66
      kernel:  iwl_mvm_mac_start+0x76/0x150 [iwlmvm]
      kernel:  ? ftrace_regs_caller_end+0x66/0x66
      kernel:  drv_start+0x79/0x180 [mac80211]
      kernel:  ? ftrace_regs_caller_end+0x66/0x66
      kernel:  ieee80211_reconfig+0x1523/0x1ce0 [mac80211]
      kernel:  ? synchronize_net+0x4/0x50
      kernel:  ? ftrace_regs_caller_end+0x66/0x66
      kernel:  ieee80211_restart_work+0x108/0x170 [mac80211]
      kernel:  ? ftrace_regs_caller_end+0x66/0x66
      kernel:  process_one_work+0x250/0x530
      kernel:  ? ftrace_regs_caller_end+0x66/0x66
      kernel:  worker_thread+0x48/0x3a0
      kernel:  ? __pfx_worker_thread+0x10/0x10
      kernel:  kthread+0x10f/0x140
      kernel:  ? __pfx_kthread+0x10/0x10
      kernel:  ret_from_fork+0x29/0x50
      kernel:  </TASK>

      Fixes: 4444bc2116ae ("wifi: mac80211: Proper mark iTXQs for resumption")
      Link: https://lore.kernel.org/all/1f58a0d1-d2b9-d851-73c3-93fcc607501c@alu.unizg.hr/
      Reported-by: Mirsad Goran Todorovac <mirsad.todorovac@alu.unizg.hr>
      Cc: Gregory Greenman <gregory.greenman@intel.com>
      Cc: Johannes Berg <johannes.berg@intel.com>
      Link: https://lore.kernel.org/all/cdc80531-f25f-6f9d-b15f-25e16130b53a@alu.unizg.hr/
      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: Leon Romanovsky <leon@kernel.org>
      Cc: Alexander Wetzel <alexander@wetzel-home.de>
      Signed-off-by: Mirsad Goran Todorovac <mirsad.todorovac@alu.unizg.hr>
      Reviewed-by: Leon Romanovsky <leonro@nvidia.com>
      Reviewed-by: tag, or it goes automatically?
      Link: https://lore.kernel.org/r/20230425164005.25272-1-mirsad.todorovac@alu.unizg.hr
      Signed-off-by: Johannes Berg <johannes.berg@intel.com>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 1bdacca02d9c0ea2ca586951c41dc865960c565c
  Author: Ilan Peer <ilan.peer@intel.com>
  Date:   Mon Apr 24 10:32:24 2023 +0300
```

```
    wifi: cfg80211: Drop entries with invalid BSSIDs in RNR

    [ Upstream commit 1b6b4ed01493b7ea2205ab83c49198f7d13ca9d2 ]

    Ignore AP information for entries that include an invalid
    BSSID in the TBTT information field, e.g., all zeros BSSIDs.

    Fixes: c8cb5b854b40 ("nl80211/cfg80211: support 6 GHz scanning")
    Signed-off-by: Ilan Peer <ilan.peer@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link:
https://lore.kernel.org/r/20230424103224.5e65d04d1448.Ic10c8577ae4a85272c407106c9d0a2ecb5372
743@changeid
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dcc34cd93597b392c7f1e47a16d909c871b173b0
Author: Ahmed Zaki <ahmed.zaki@intel.com>
Date:   Mon Apr 17 12:09:39 2023 -0600

    iavf: send VLAN offloading caps once after VFR

    [ Upstream commit 7dcbdf29282fbcdb646dc785e8a57ed2c2fec8ba ]

    When the user disables rxvlan offloading and then changes the number of
    channels, all VLAN ports are unable to receive traffic.

    Changing the number of channels triggers a VFR reset. During re-init, when
    VIRTCHNL_OP_GET_OFFLOAD_VLAN_V2_CAPS is received, we do:
    1 - set the IAVF_FLAG_SETUP_NETDEV_FEATURES flag
    2 - call
        iavf_set_vlan_offload_features(adapter, 0, netdev->features);

    The second step sends to the PF the __default__ features, in this case
    aq_required |= IAVF_FLAG_AQ_ENABLE_CTAG_VLAN_STRIPPING

    While the first step forces the watchdog task to call
    netdev_update_features() ->  iavf_set_features() ->
    iavf_set_vlan_offload_features(adapter, netdev->features, features).
    Since the user disabled the "rxvlan", this sets:
    aq_required |= IAVF_FLAG_AQ_DISABLE_CTAG_VLAN_STRIPPING

    When we start processing the AQ commands, both flags are enabled. Since we
    process DISABLE_XTAG first then ENABLE_XTAG, this results in the PF
    enabling the rxvlan offload. This breaks all communications on the VLAN
    net devices.

    Fix by removing the call to iavf_set_vlan_offload_features() (second
    step). Calling netdev_update_features() from watchdog task is enough for
    both init and reset paths.

    Fixes: 7598f4b40bd6 ("iavf: Move netdev_update_features() into watchdog task")
    Signed-off-by: Ahmed Zaki <ahmed.zaki@intel.com>
    Tested-by: Rafal Romanowski <rafal.romanowski@intel.com>
    Reviewed-by: Leon Romanovsky <leonro@nvidia.com>
    Signed-off-by: Tony Nguyen <anthony.l.nguyen@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c362750362908598bfa16426bea542044089ca62
Author: Dawid Wesierski <dawidx.wesierski@intel.com>
Date:   Tue Apr 18 11:52:55 2023 +0200

    ice: Fix ice VF reset during iavf initialization

    [ Upstream commit 7255355a0636b4eff08d5e8139c77d98f151c4fc ]
```

```
    Fix the current implementation that causes ice_trigger_vf_reset()
    to start resetting the VF even when the VF-NIC is still initializing.

    When we reset NIC with ice driver it can interfere with
    iavf-vf initialization e.g. during consecutive resets induced by ice

    iavf                ice
      |                  |
      |<-----------------|
      |            ice resets vf
     iavf                |
     reset               |
     start               |
      |<-----------------|
      |             ice resets vf
      |             causing iavf
      |             initialization
      |             error
      |                  |
     iavf
     reset
     end

    This leads to a series of -53 errors
    (failed to init adminq) from the IAVF.

    Change the state of the vf_state field to be not active when the IAVF
    is still initializing. Make sure to wait until receiving the message on
    the message box to ensure that the vf is ready and initializded.

    In simple terms we use the ACTIVE flag to make sure that the ice
    driver knows if the iavf is ready for another reset

      iavf                  ice
        |                    |
        |                    |
        |<-------------- ice resets vf
      iavf           vf_state != ACTIVE
      reset                |
      start                |
        |                  |
        |                  |
      iavf                 |
      reset-------> vf_state == ACTIVE
      end              ice resets vf
        |                  |
        |                  |

    Fixes: c54d209c78b8 ("ice: Wait for VF to be reset/ready before configuration")
    Signed-off-by: Dawid Wesierski <dawidx.wesierski@intel.com>
    Signed-off-by: Kamil Maziarz <kamil.maziarz@intel.com>
    Acked-by: Jacob Keller <Jacob.e.keller@intel.com>
    Tested-by: Rafal Romanowski <rafal.romanowski@intel.com>
    Signed-off-by: Tony Nguyen <anthony.l.nguyen@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dcb01b0b39e71b788a092438af7d1f7ef21d025a
Author: Ahmed Zaki <ahmed.zaki@intel.com>
Date:   Mon Apr 17 17:44:45 2023 -0600

    ice: Fix stats after PF reset

    [ Upstream commit ab7470bc6d8fb5f3004ccc8e4dfd49aab0f27561 ]
```

```
    After a core PF reset, the VFs were showing wrong Rx/Tx stats. This is a
    regression in commit 6624e780a577 ("ice: split ice_vsi_setup into smaller
    functions") caused by missing to set "stat_offsets_loaded = false" in the
    ice_vsi_rebuild() path.

    Fixes: 6624e780a577 ("ice: split ice_vsi_setup into smaller functions")
    Signed-off-by: Ahmed Zaki <ahmed.zaki@intel.com>
    Reviewed-by: Alexander Lobakin <aleksander.lobakin@intel.com>
    Tested-by: Rafal Romanowski <rafal.romanowski@intel.com>
    Reviewed-by: Leon Romanovsky <leonro@nvidia.com>
    Signed-off-by: Tony Nguyen <anthony.l.nguyen@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1e4b5e722ba5a358d6964de16fec455229aaf360
Author: Florian Fainelli <f.fainelli@gmail.com>
Date:   Sun May 14 19:56:07 2023 -0700

    net: bcmgenet: Restore phy_stop() depending upon suspend/close

    [ Upstream commit 225c657945c4a6307741cb3cc89467eadcc26e9b ]

    Removing the phy_stop() from bcmgenet_netif_stop() ended up causing
    warnings from the PHY library that phy_start() is called from the
    RUNNING state since we are no longer stopping the PHY state machine
    during bcmgenet_suspend().

    Restore the call to phy_stop() but make it conditional on being called
    from the close or suspend path.

    Fixes: c96e731c93ff ("net: bcmgenet: connect and disconnect from the PHY state machine")
    Fixes: 93e0401e0fc0 ("net: bcmgenet: Remove phy_stop() from bcmgenet_netif_stop()")
    Signed-off-by: Florian Fainelli <f.fainelli@gmail.com>
    Reviewed-by: Pavan Chebbi <pavan.chebbi@broadcom.com>
    Link: https://lore.kernel.org/r/20230515025608.2587012-1-f.fainelli@gmail.com
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 970ea2284d32292572ea2c3f68c3089656088043
Author: Florian Fainelli <f.fainelli@gmail.com>
Date:   Thu May 4 16:07:27 2023 -0700

    net: bcmgenet: Remove phy_stop() from bcmgenet_netif_stop()

    [ Upstream commit 93e0401e0fc0c54b0ac05b687cd135c2ac38187c ]

    The call to phy_stop() races with the later call to phy_disconnect(),
    resulting in concurrent phy_suspend() calls being run from different
    CPUs. The final call to phy_disconnect() ensures that the PHY is
    stopped and suspended, too.

    Fixes: c96e731c93ff ("net: bcmgenet: connect and disconnect from the PHY state machine")
    Signed-off-by: Florian Fainelli <f.fainelli@gmail.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Stable-dep-of: 225c657945c4 ("net: bcmgenet: Restore phy_stop() depending upon
suspend/close")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit eb33ceffab94de72d74c13ee2c443ae133854d44
Author: Oliver Hartkopp <socketcan@hartkopp.net>
Date:   Sat May 6 20:45:15 2023 +0200

    can: dev: fix missing CAN XL support in can_put_echo_skb()

    [ Upstream commit 6bffdc38f9935bae49f980448f3f6be2dada0564 ]
```

```
    can_put_echo_skb() checks for the enabled IFF_ECHO flag and the
    correct ETH_P type of the given skbuff. When implementing the CAN XL
    support the new check for ETH_P_CANXL has been forgotten.

    Fixes: fb08cba12b52 ("can: canxl: update CAN infrastructure for CAN XL frames")
    Signed-off-by: Oliver Hartkopp <socketcan@hartkopp.net>
    Link: https://lore.kernel.org/all/20230506184515.39241-1-socketcan@hartkopp.net
    Signed-off-by: Marc Kleine-Budde <mkl@pengutronix.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2dc9701eb0cbcde251a71038f318e79090bf0c8d
Author: Vineeth Vijayan <vneethv@linux.ibm.com>
Date:   Tue May 2 11:12:42 2023 +0200

    s390/cio: include subchannels without devices also for evaluation

    [ Upstream commit b1b0d5aec1cf9f9a900a14964f869c68688d923e ]

    Currently when the new channel-path is enabled, we do evaluation only
    on the subchannels with a device connected on it. This is because,
    in the past, if the device in the subchannel is not working or not
    available, we used to unregister the subchannels. But, from the 'commit
    2297791c92d0 ("s390/cio: dont unregister subchannel from child-drivers")'
    we allow subchannels with or without an active device connected
    on it. So, when we do the io_subchannel_verify, make sure that,
    we are evaluating the subchannels without any device too.

    Fixes: 2297791c92d0 ("s390/cio: dont unregister subchannel from child-drivers")
    Reported-by: Boris Fiuczynski <fiuczy@linux.ibm.com>
    Signed-off-by: Vineeth Vijayan <vneethv@linux.ibm.com>
    Reviewed-by: Peter Oberparleiter <oberpar@linux.ibm.com>
    Signed-off-by: Alexander Gordeev <agordeev@linux.ibm.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f32d1fa401b314e2d3ccf6a2eff1d02ccc3b7a44
Author: Xin Long <lucien.xin@gmail.com>
Date:   Sun May 14 15:52:29 2023 -0400

    tipc: check the bearer min mtu properly when setting it by netlink

    [ Upstream commit 35a089b5d793d2bfd2cc7cfa6104545184de2ce7 ]

    Checking the bearer min mtu with tipc_udp_mtu_bad() only works for
    IPv4 UDP bearer, and IPv6 UDP bearer has a different value for the
    min mtu. This patch checks with encap_hlen + TIPC_MIN_BEARER_MTU
    for min mtu, which works for both IPv4 and IPv6 UDP bearer.

    Note that tipc_udp_mtu_bad() is still used to check media min mtu
    in __tipc_nl_media_set(), as m->mtu currently is only used by the
    IPv4 UDP bearer as its default mtu value.

    Fixes: 682cd3cf946b ("tipc: confgiure and apply UDP bearer MTU on running links")
    Signed-off-by: Xin Long <lucien.xin@gmail.com>
    Acked-by: Jon Maloy <jmaloy@redhat.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1dd7ae5e0cf5a56e513f7ab7ab9570b7496281d2
Author: Xin Long <lucien.xin@gmail.com>
Date:   Sun May 14 15:52:28 2023 -0400

    tipc: do not update mtu if msg_max is too small in mtu negotiation

    [ Upstream commit 56077b56cd3fb78e1c8619e29581ba25a5c55e86 ]
```

```
    When doing link mtu negotiation, a malicious peer may send Activate msg
    with a very small mtu, e.g. 4 in Shuang's testing, without checking for
    the minimum mtu, l->mtu will be set to 4 in tipc_link_proto_rcv(), then
    n->links[bearer_id].mtu is set to 4294967228, which is a overflow of
    '4 - INT_H_SIZE - EMSG_OVERHEAD' in tipc_link_mss().

    With tipc_link.mtu = 4, tipc_link_xmit() kept printing the warning:

     tipc: Too large msg, purging xmit list 1 5 0 40 4!
     tipc: Too large msg, purging xmit list 1 15 0 60 4!

    And with tipc_link_entry.mtu 4294967228, a huge skb was allocated in
    named_distribute(), and when purging it in tipc_link_xmit(), a crash
    was even caused:

      general protection fault, probably for non-canonical address 0x2100001011000dd: 0000
[#1] PREEMPT SMP PTI
      CPU: 0 PID: 0 Comm: swapper/0 Kdump: loaded Not tainted 6.3.0.neta #19
      RIP: 0010:kfree_skb_list_reason+0x7e/0x1f0
      Call Trace:
       <IRQ>
       skb_release_data+0xf9/0x1d0
       kfree_skb_reason+0x40/0x100
       tipc_link_xmit+0x57a/0x740 [tipc]
       tipc_node_xmit+0x16c/0x5c0 [tipc]
       tipc_named_node_up+0x27f/0x2c0 [tipc]
       tipc_node_write_unlock+0x149/0x170 [tipc]
       tipc_rcv+0x608/0x740 [tipc]
       tipc_udp_recv+0xdc/0x1f0 [tipc]
       udp_queue_rcv_one_skb+0x33e/0x620
       udp_unicast_rcv_skb.isra.72+0x75/0x90
       __udp4_lib_rcv+0x56d/0xc20
       ip_protocol_deliver_rcu+0x100/0x2d0

    This patch fixes it by checking the new mtu against tipc_bearer_min_mtu(),
    and not updating mtu if it is too small.

    Fixes: ed193ece2649 ("tipc: simplify link mtu negotiation")
    Reported-by: Shuang Li <shuali@redhat.com>
    Signed-off-by: Xin Long <lucien.xin@gmail.com>
    Acked-by: Jon Maloy <jmaloy@redhat.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 625fa0142d0648e0d654cb0b545769a282bdb761
Author: Xin Long <lucien.xin@gmail.com>
Date:   Sun May 14 15:52:27 2023 -0400

    tipc: add tipc_bearer_min_mtu to calculate min mtu

    [ Upstream commit 3ae6d66b605be604644d4bb5708a7ffd9cf1abe8 ]

    As different media may requires different min mtu, and even the
    same media with different net family requires different min mtu,
    add tipc_bearer_min_mtu() to calculate min mtu accordingly.

    This API will be used to check the new mtu when doing the link
    mtu negotiation in the next patch.

    Signed-off-by: Xin Long <lucien.xin@gmail.com>
    Acked-by: Jon Maloy <jmaloy@redhat.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Stable-dep-of: 56077b56cd3f ("tipc: do not update mtu if msg_max is too small in mtu
negotiation")
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 037768b28e3752c07d63d1c72a651a6775b080bb
Author: Feng Liu <feliu@nvidia.com>
Date:   Fri May 12 11:18:12 2023 -0400

    virtio_net: Fix error unwinding of XDP initialization

    [ Upstream commit 5306623a9826aa7d63b32c6a3803c798a765474d ]

    When initializing XDP in virtnet_open(), some rq xdp initialization
    may hit an error causing net device open failed. However, previous
    rqs have already initialized XDP and enabled NAPI, which is not the
    expected behavior. Need to roll back the previous rq initialization
    to avoid leaks in error unwinding of init code.

    Also extract helper functions of disable and enable queue pairs.
    Use newly introduced disable helper function in error unwinding and
    virtnet_close. Use enable helper function in virtnet_open.

    Fixes: 754b8a21a96d ("virtio_net: setup xdp_rxq_info")
    Signed-off-by: Feng Liu <feliu@nvidia.com>
    Reviewed-by: Jiri Pirko <jiri@nvidia.com>
    Reviewed-by: William Tu <witu@nvidia.com>
    Acked-by: Michael S. Tsirkin <mst@redhat.com>
    Acked-by: Jason Wang <jasowang@redhat.com>
    Reviewed-by: Xuan Zhuo <xuanzhuo@linux.alibaba.com>
    Acked-by: Michael S. Tsirkin <mst@redhat.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ac7106a35e3c620076024164693b34cfdfb352ea
Author: Shenwei Wang <shenwei.wang@nxp.com>
Date:   Fri May 12 08:38:43 2023 -0500

    net: fec: remove the xdp_return_frame when lack of tx BDs

    [ Upstream commit 6ead9c98cafcbc6992cf35f0ca393df2c03e3316 ]

    In the implementation, the sent_frame count does not increment when
    transmit errors occur. Therefore, bq_xmit_all() will take care of
    returning the XDP frames.

    Fixes: 26312c685ae0 ("net: fec: correct the counting of XDP sent frames")
    Signed-off-by: Shenwei Wang <shenwei.wang@nxp.com>
    Reviewed-by: Horatiu Vultur <horatiu.vultur@microchip.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cb38e62922aa3991793344b5a5870e7291c74a44
Author: Dong Chenchen <dongchenchen2@huawei.com>
Date:   Thu May 11 20:54:40 2023 +0800

    net: nsh: Use correct mac_offset to unwind gso skb in nsh_gso_segment()

    [ Upstream commit c83b49383b595be50647f0c764a48c78b5f3c4f8 ]

    As the call trace shows, skb_panic was caused by wrong skb->mac_header
    in nsh_gso_segment():

    invalid opcode: 0000 [#1] PREEMPT SMP KASAN PTI
    CPU: 3 PID: 2737 Comm: syz Not tainted 6.3.0-next-20230505 #1
    RIP: 0010:skb_panic+0xda/0xe0
    call Trace:
     skb_push+0x91/0xa0
     nsh_gso_segment+0x4f3/0x570
```

```
       skb_mac_gso_segment+0x19e/0x270
       __skb_gso_segment+0x1e8/0x3c0
       validate_xmit_skb+0x452/0x890
       validate_xmit_skb_list+0x99/0xd0
       sch_direct_xmit+0x294/0x7c0
       __dev_queue_xmit+0x16f0/0x1d70
       packet_xmit+0x185/0x210
       packet_snd+0xc15/0x1170
       packet_sendmsg+0x7b/0xa0
       sock_sendmsg+0x14f/0x160

     The root cause is:
     nsh_gso_segment() use skb->network_header - nhoff to reset mac_header
     in skb_gso_error_unwind() if inner-layer protocol gso fails.
     However, skb->network_header may be reset by inner-layer protocol
     gso function e.g. mpls_gso_segment. skb->mac_header reset by the
     inaccurate network_header will be larger than skb headroom.

     nsh_gso_segment
         nhoff = skb->network_header - skb->mac_header;
         __skb_pull(skb,nsh_len)
         skb_mac_gso_segment
             mpls_gso_segment
                 skb_reset_network_header(skb);//skb->network_header+=nsh_len
                 return -EINVAL;
         skb_gso_error_unwind
             skb_push(skb, nsh_len);
             skb->mac_header = skb->network_header - nhoff;
             // skb->mac_header > skb->headroom, cause skb_push panic

     Use correct mac_offset to restore mac_header and get rid of nhoff.

     Fixes: c411ed854584 ("nsh: add GSO support")
     Reported-by: syzbot+632b5d9964208bfef8c0@syzkaller.appspotmail.com
     Suggested-by: Eric Dumazet <edumazet@google.com>
     Signed-off-by: Dong Chenchen <dongchenchen2@huawei.com>
     Signed-off-by: David S. Miller <davem@davemloft.net>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 364284b20567c5f7b87bf1b1915ffcd1ce2c4c22
  Author: Arnd Bergmann <arnd@arndb.de>
  Date:   Mon Apr 17 23:04:11 2023 +0200

     drm/exynos: fix g2d_open/close helper function definitions

     [ Upstream commit 2ef0785b30bd6549ddbc124979f1b6596e065ae2 ]

     The empty stub functions are defined as global functions, which
     causes a warning because of missing prototypes:

     drivers/gpu/drm/exynos/exynos_drm_g2d.h:37:5: error: no previous prototype for
  'g2d_open'
     drivers/gpu/drm/exynos/exynos_drm_g2d.h:42:5: error: no previous prototype for
  'g2d_close'

     Mark them as 'static inline' to avoid the warning and to make
     them behave as intended.

     Fixes: eb4d9796fa34 ("drm/exynos: g2d: Convert to driver component API")
     Signed-off-by: Arnd Bergmann <arnd@arndb.de>
     Reviewed-by: Andi Shyti <andi.shyti@kernel.org>
     Signed-off-by: Inki Dae <inki.dae@samsung.com>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 0f2b855c9d71afb4d0197b56bac20e49717dd44f
```

```
Author: Ranjani Sridharan <ranjani.sridharan@linux.intel.com>
Date:   Fri May 12 14:46:30 2023 +0300

    ASoC: SOF: topology: Fix logic for copying tuples

    [ Upstream commit 41c5305cc3d827d2ea686533777a285176ae01a0 ]

    Topology could have more instances of the tokens being searched for than
    the number of sets that need to be copied. Stop copying token after the
    limit of number of token instances has been reached. This worked before
    only by chance as we had allocated more size for the tuples array than
    the number of actual tokens being parsed.

    Fixes: 7006d20e5e9d ("ASoC: SOF: Introduce IPC3 ops")
    Signed-off-by: Ranjani Sridharan <ranjani.sridharan@linux.intel.com
    Reviewed-by: Péter Ujfalusi <peter.ujfalusi@linux.intel.com
    Reviewed-by: Bard Liao <yung-chuan.liao@linux.intel.com
    Reviewed-by: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com
    Signed-off-by: Peter Ujfalusi <peter.ujfalusi@linux.intel.com
    Link: https://lore.kernel.org/r/20230512114630.24439-1-peter.ujfalusi@linux.intel.com
    Signed-off-by: Mark Brown <broonie@kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dffd9e2b57cb845930fa885aa634a847ba2130dd
Author: Douglas Anderson <dianders@chromium.org>
Date:   Thu May 11 09:25:12 2023 -0700

    ASoC: mediatek: mt8186: Fix use-after-free in driver remove path

    [ Upstream commit a93d2afd3f77a7331271a0f25c6a11003db69b3c ]

    When devm runs function in the "remove" path for a device it runs them
    in the reverse order. That means that if you have parts of your driver
    that aren't using devm or are using "roll your own" devm w/
    devm_add_action_or_reset() you need to keep that in mind.

    The mt8186 audio driver didn't quite get this right. Specifically, in
    mt8186_init_clock() it called mt8186_audsys_clk_register() and then
    went on to call a bunch of other devm function. The caller of
    mt8186_init_clock() used devm_add_action_or_reset() to call
    mt8186_deinit_clock() but, because of the intervening devm functions,
    the order was wrong.

    Specifically at probe time, the order was:
    1. mt8186_audsys_clk_register()
    2. afe_priv->clk = devm_kcalloc(...)
    3. afe_priv->clk[i] = devm_clk_get(...)

    At remove time, the order (which should have been 3, 2, 1) was:
    1. mt8186_audsys_clk_unregister()
    3. Free all of afe_priv->clk[i]
    2. Free afe_priv->clk

    The above seemed to be causing a use-after-free. Luckily, it's easy to
    fix this by simply using devm more correctly. Let's move the
    devm_add_action_or_reset() to the right place. In addition to fixing
    the use-after-free, code inspection shows that this fixes a leak
    (missing call to mt8186_audsys_clk_unregister()) that would have
    happened if any of the syscon_regmap_lookup_by_phandle() calls in
    mt8186_init_clock() had failed.

    Fixes: 55b423d5623c ("ASoC: mediatek: mt8186: support audio clock control in platform
driver")
    Signed-off-by: Douglas Anderson <dianders@chromium.org
    Link:
```

```
https://lore.kernel.org/r/20230511092437.1.I31cceffc8c45bb1af16eb613e197b3df92cdc19e@changei
d
    Signed-off-by: Mark Brown <broonie@kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 15b71ad25ea41612b4d1cfd8cdf8267eb047d11e
Author: Peter Ujfalusi <peter.ujfalusi@linux.intel.com>
Date:   Fri May 12 14:03:17 2023 +0300

    ASoC: SOF: ipc3-topology: Make sure that only one cmd is sent in dai_config

    [ Upstream commit 4708449eafe607423346061689269857 98c9c9b8 ]

    The commands in sof_ipc_dai_config.flags are encoded as bits:
    1 (bit0) - hw_params
    2 (bit1) - hw_free
    4 (bit2) - pause

    These are commands, they cannot be combined as one would assume, for
    example
    3 (bit0 | bit1) is invalid.

    This can happen right at the second start of a stream as at the end of the
    first stream we set the hw_free command (bit1) and on the second start we
    would OR on top of it the hw_params (bit0).

    Fixes: b66bfc3a9810 ("ASoC: SOF: sof-audio: Fix broken early bclk feature for SSP")
    Signed-off-by: Peter Ujfalusi <peter.ujfalusi@linux.intel.com
    Reviewed-by: Daniel Baluta <daniel.baluta@nxp.com
    Reviewed-by: Bard Liao <yung-chuan.liao@linux.intel.com
    Reviewed-by: Ranjani Sridharan <ranjani.sridharan@linux.intel.com
    Reviewed-by: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com
    Link: https://lore.kernel.org/r/20230512110317.5180-1-peter.ujfalusi@linux.intel.com
    Signed-off-by: Mark Brown <broonie@kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6afdb52f3b58c0fa21b0b53cb9cd03fa42b90f90
Author: Chuck Lever <chuck.lever@oracle.com>
Date:   Sun May 14 15:51:48 2023 -0400

    SUNRPC: Fix trace_svc_register() call site

    [ Upstream commit 07a27305938559fb35f7a46fb90a5e37728bdee6 ]

    The trace event recorded incorrect values for the registered family,
    protocol, and port because the arguments are in the wrong order.

    Fixes: b4af59328c25 ("SUNRPC: Trace server-side rpcbind registration events")
    Signed-off-by: Chuck Lever <chuck.lever@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 27ad3d50668f5ac7f561323c4ec089b787758572
Author: NeilBrown <neilb@suse.de>
Date:   Tue May 9 09:42:47 2023 +1000

    SUNRPC: always free ctxt when freeing deferred request

    [ Upstream commit 948f072ada23e0a504c5e4d7d71d4c83bd0785ec ]

    Since the ->xprt_ctxt pointer was added to svc_deferred_req, it has not
    been sufficient to use kfree() to free a deferred request.  We may need
    to free the ctxt as well.

    As freeing the ctxt is all that ->xpo_release_rqst() does, we repurpose
    it to explicit do that even when the ctxt is not stored in an rqst.
```

```
    So we now have ->xpo_release_ctxt() which is given an xprt and a ctxt,
    which may have been taken either from an rqst or from a dreq.  The
    caller is now responsible for clearing that pointer after the call to
    ->xpo_release_ctxt.

    We also clear dr->xprt_ctxt when the ctxt is moved into a new rqst when
    revisiting a deferred request.  This ensures there is only one pointer
    to the ctxt, so the risk of double freeing in future is reduced.  The
    new code in svc_xprt_release which releases both the ctxt and any
    rq_deferred depends on this.

    Fixes: 773f91b2cf3f ("SUNRPC: Fix NFSD's request deferral on RDMA transports")
    Signed-off-by: NeilBrown <neilb@suse.de>
    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 e0c648627322a4c7e018e5c7f837c3c03e297dbb
Author: NeilBrown <neilb@suse.de>
Date:   Tue May 9 09:41:49 2023 +1000

    SUNRPC: double free xprt_ctxt while still in use

    [ Upstream commit eb8d3a2c809abd73ab0a060fe971d6b9019aa3c1 ]

    When an RPC request is deferred, the rq_xprt_ctxt pointer is moved out
    of the svc_rqst into the svc_deferred_req.
    When the deferred request is revisited, the pointer is copied into
    the new svc_rqst - and also remains in the svc_deferred_req.

    In the (rare?) case that the request is deferred a second time, the old
    svc_deferred_req is reused - it still has all the correct content.
    However in that case the rq_xprt_ctxt pointer is NOT cleared so that
    when xpo_release_xprt is called, the ctxt is freed (UDP) or possible
    added to a free list (RDMA).
    When the deferred request is revisited for a second time, it will
    reference this ctxt which may be invalid, and the free the object a
    second time which is likely to oops.

    So change svc_defer() to *always* clear rq_xprt_ctxt, and assert that
    the value is now stored in the svc_deferred_req.

    Fixes: 773f91b2cf3f ("SUNRPC: Fix NFSD's request deferral on RDMA transports")
    Signed-off-by: NeilBrown <neilb@suse.de>
    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 c8f9c05e1ebcc9c7bc211cc8b74d8fb86a8756fc
Author: Duoming Zhou <duoming@zju.edu.cn>
Date:   Wed Mar 8 12:55:14 2023 +0000

    media: netup_unidvb: fix use-after-free at del_timer()

    [ Upstream commit 0f5bb36bf9b39a2a96e730bf4455095b50713f63 ]

    When Universal DVB card is detaching, netup_unidvb_dma_fini()
    uses del_timer() to stop dma->timeout timer. But when timer
    handler netup_unidvb_dma_timeout() is running, del_timer()
    could not stop it. As a result, the use-after-free bug could
    happen. The process is shown below:

        (cleanup routine)          |        (timer routine)
                                   | mod_timer(&dev->tx_sim_timer, ..)
    netup_unidvb_finidev()         | (wait a time)
```

```
       netup_unidvb_dma_fini()      | netup_unidvb_dma_timeout()
         del_timer(&dma->timeout);  |
                                    |   ndev->pci_dev->dev //USE
```

    Fix by changing del_timer() to del_timer_sync().

    Link: https://lore.kernel.org/linux-media/20230308125514.4208-1-duoming@zju.edu.cn
    Fixes: 52b1eaf4c59a ("[media] netup_unidvb: NetUP Universal DVB-S/S2/T/T2/C PCI-E card driver")
    Signed-off-by: Duoming Zhou <duoming@zju.edu.cn>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6d588ddc1c4caa25a03711e0d5da4e46f7f734be
Author: Jijie Shao <shaojijie@huawei.com>
Date:   Fri May 12 18:00:14 2023 +0800

    net: hns3: fix reset timeout when enable full VF

    [ Upstream commit 6b45d5ff8c2c61baddd67d7510075ae121c5e704 ]

    The timeout of the cmdq reset command has been increased to
    resolve the reset timeout issue in the full VF scenario.
    The timeout of other cmdq commands remains unchanged.

    Fixes: 8d307f8e8cf1 ("net: hns3: create new set of unified hclge_comm_cmd_send APIs")
    Signed-off-by: Jijie Shao <shaojijie@huawei.com>
    Signed-off-by: Hao Lan <lanhao@huawei.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 89070115f589181aa76b87e2b611b25dc1384ed4
Author: Jie Wang <wangjie125@huawei.com>
Date:   Fri May 12 18:00:13 2023 +0800

    net: hns3: fix reset delay time to avoid configuration timeout

    [ Upstream commit 814d0c786068e858d889ada3153bff82f64223ad ]

    Currently the hns3 vf function reset delays 5000ms before vf rebuild
    process. In product applications, this delay is too long for application
    configurations and causes configuration timeout.

    According to the tests, 500ms delay is enough for reset process except PF
    FLR. So this patch modifies delay to 500ms in these scenarios.

    Fixes: 6988eb2a9b77 ("net: hns3: Add support to reset the enet/ring mgmt layer")
    Signed-off-by: Jie Wang <wangjie125@huawei.com>
    Signed-off-by: Hao Lan <lanhao@huawei.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 999c2b7502190ba4896868d964d24cb26ca9c08d
Author: Jijie Shao <shaojijie@huawei.com>
Date:   Fri May 12 18:00:12 2023 +0800

    net: hns3: fix sending pfc frames after reset issue

    [ Upstream commit f14db07064727dd3bc0906c77a6d2759c1bbb395 ]

    To prevent the system from abnormally sending PFC frames after an
    abnormal reset. The hns3 driver notifies the firmware to disable pfc
    before reset.

    Fixes: 35d93a30040c ("net: hns3: adjust the process of PF reset")

```
    Signed-off-by: Jijie Shao <shaojijie@huawei.com>
    Signed-off-by: Hao Lan <lanhao@huawei.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 94a17fff6e60323fd6bfe2e9f5a33a8fa82cf538
Author: Jie Wang <wangjie125@huawei.com>
Date:   Fri May 12 18:00:11 2023 +0800

    net: hns3: fix output information incomplete for dumping tx queue info with debugfs

    [ Upstream commit 89f6bfb071182f05d7188c255b0e7251c3806f16 ]

    In function hns3_dump_tx_queue_info, The print buffer is not enough when
    the tx BD number is configured to 32760. As a result several BD
    information wouldn't be displayed.

    So fix it by increasing the tx queue print buffer length.

    Fixes: 630a6738da82 ("net: hns3: adjust string spaces of some parameters of tx bd info
in debugfs")
    Signed-off-by: Jie Wang <wangjie125@huawei.com>
    Signed-off-by: Hao Lan <lanhao@huawei.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6cb2dfabe538def57f7f9b221a1c2759d8a67e92
Author: Clément Léger <clement.leger@bootlin.com>
Date:   Fri May 12 09:27:12 2023 +0200

    net: dsa: rzn1-a5psw: disable learning for standalone ports

    [ Upstream commit ec52b69c046a6219011af780aca155a96719637b ]

    When ports are in standalone mode, they should have learning disabled to
    avoid adding new entries in the MAC lookup table which might be used by
    other bridge ports to forward packets. While adding that, also make sure
    learning is enabled for CPU port.

    Fixes: 888cdb892b61 ("net: dsa: rzn1-a5psw: add Renesas RZ/N1 advanced 5 port switch
driver")
    Signed-off-by: Clément Léger <clement.leger@bootlin.com>
    Signed-off-by: Alexis Lothoré <alexis.lothore@bootlin.com>
    Reviewed-by: Piotr Raczynski <piotr.raczynski@intel.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit fc61ef099e259229126491d10ad5060b8047b71c
Author: Alexis Lothoré <alexis.lothore@bootlin.com>
Date:   Fri May 12 09:27:11 2023 +0200

    net: dsa: rzn1-a5psw: fix STP states handling

    [ Upstream commit ebe9bc50952757b4b25eaf514da7c464196c9606 ]

    stp_set_state() should actually allow receiving BPDU while in LEARNING
    mode which is not the case. Additionally, the BLOCKEN bit does not
    actually forbid sending forwarded frames from that port. To fix this, add
    a5psw_port_tx_enable() function which allows to disable TX. However, while
    its name suggest that TX is totally disabled, it is not and can still
    allow to send BPDUs even if disabled. This can be done by using forced
    forwarding with the switch tagging mechanism but keeping "filtering"
    disabled (which is already the case in the rzn1-a5sw tag driver). With
    these fixes, STP support is now functional.
```

```
    Fixes: 888cdb892b61 ("net: dsa: rzn1-a5psw: add Renesas RZ/N1 advanced 5 port switch
driver")
    Signed-off-by: Clément Léger <clement.leger@bootlin.com>
    Signed-off-by: Alexis Lothoré <alexis.lothore@bootlin.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 04b06ac2d524e6e65117970063746a818cf39e41
Author: Clément Léger <clement.leger@bootlin.com>
Date:   Fri May 12 09:27:10 2023 +0200

    net: dsa: rzn1-a5psw: enable management frames for CPU port

    [ Upstream commit 9e4b45f20c5aac786c728619e5ee746bffce1798 ]

    Currently, management frame were discarded before reaching the CPU port due
    to a misconfiguration of the MGMT_CONFIG register. Enable them by setting
    the correct value in this register in order to correctly receive management
    frame and handle STP.

    Fixes: 888cdb892b61 ("net: dsa: rzn1-a5psw: add Renesas RZ/N1 advanced 5 port switch
driver")
    Signed-off-by: Clément Léger <clement.leger@bootlin.com>
    Signed-off-by: Alexis Lothoré <alexis.lothore@bootlin.com>
    Reviewed-by: Piotr Raczynski <piotr.raczynski@intel.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c594f3ad3219ebd3f1336b50449736f85c5a4b44
Author: Xin Long <lucien.xin@gmail.com>
Date:   Thu May 11 19:22:11 2023 -0400

    erspan: get the proto with the md version for collect_md

    [ Upstream commit d80fc101d2eb9b3188c228d61223890aeea480a4 ]

    In commit 20704bd1633d ("erspan: build the header with the right proto
    according to erspan_ver"), it gets the proto with t->parms.erspan_ver,
    but t->parms.erspan_ver is not used by collect_md branch, and instead
    it should get the proto with md->version for collect_md.

    Thanks to Kevin for pointing this out.

    Fixes: 20704bd1633d ("erspan: build the header with the right proto according to
erspan_ver")
    Fixes: 94d7d8f29287 ("ip6_gre: add erspan v2 support")
    Reported-by: Kevin Traynor <ktraynor@redhat.com>
    Signed-off-by: Xin Long <lucien.xin@gmail.com>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Reviewed-by: William Tu <u9012063@gmail.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 56a81445b8e4b8906d557518c5dae3ddbb447d1e
Author: Doug Berger <opendmb@gmail.com>
Date:   Thu Apr 27 11:19:16 2023 -0700

    serial: 8250_bcm7271: fix leak in `brcmuart_probe`

    [ Upstream commit f264f2f6f4788dc031cef60a0cf2881902736709 ]

    Smatch reports:
    drivers/tty/serial/8250/8250_bcm7271.c:1120 brcmuart_probe() warn:
    'baud_mux_clk' from clk_prepare_enable() not released on lines: 1032.
```

```
    The issue is fixed by using a managed clock.

    Fixes: 41a469482de2 ("serial: 8250: Add new 8250-core based Broadcom STB driver")
    Reported-by: XuDong Liu <m202071377@hust.edu.cn>
    Link: https://lore.kernel.org/lkml/20230424125100.4783-1-m202071377@hust.edu.cn/
    Signed-off-by: Doug Berger <opendmb@gmail.com>
    Acked-by: Florian Fainelli <f.fainelli@gmail.com>
    Link: https://lore.kernel.org/r/20230427181916.2983697-3-opendmb@gmail.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5ea03de0199a3efa25724241624f0b17a738f436
Author: Doug Berger <opendmb@gmail.com>
Date:   Thu Apr 27 11:19:15 2023 -0700

    serial: 8250_bcm7271: balance clk_enable calls

    [ Upstream commit 8a3b5477256a54ae4a470dcebbcf8cdc18e4696d ]

    The sw_baud clock must be disabled when the device driver is not
    connected to the device. This now occurs when probe fails and
    upon remove.

    Fixes: 41a469482de2 ("serial: 8250: Add new 8250-core based Broadcom STB driver")
    Reported-by: XuDong Liu <m202071377@hust.edu.cn>
    Link: https://lore.kernel.org/lkml/20230424125100.4783-1-m202071377@hust.edu.cn/
    Signed-off-by: Doug Berger <opendmb@gmail.com>
    Acked-by: Florian Fainelli <f.fainelli@gmail.com>
    Link: https://lore.kernel.org/r/20230427181916.2983697-2-opendmb@gmail.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 40a462313ba4f337a2b419e7fb4a670f3dd95e14
Author: Ke Zhang <m202171830@hust.edu.cn>
Date:   Fri Apr 28 11:16:36 2023 +0800

    serial: arc_uart: fix of_iomap leak in `arc_serial_probe`

    [ Upstream commit 8ab5fc55d7f65d58a3c3aeadf11bdf60267cd2bd ]

    Smatch reports:

    drivers/tty/serial/arc_uart.c:631 arc_serial_probe() warn:
    'port->membase' from of_iomap() not released on lines: 631.

    In arc_serial_probe(), if uart_add_one_port() fails,
    port->membase is not released, which would cause a resource leak.

    To fix this, I replace of_iomap with devm_platform_ioremap_resource.

    Fixes: 8dbe1d5e09a7 ("serial/arc: inline the probe helper")
    Signed-off-by: Ke Zhang <m202171830@hust.edu.cn>
    Reviewed-by: Dongliang Mu <dzm91@hust.edu.cn>
    Link: https://lore.kernel.org/r/20230428031636.44642-1-m202171830@hust.edu.cn
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 36c6c629b77d3b2327d70b1a7c2ce879e90e392b
Author: Arnd Bergmann <arnd@arndb.de>
Date:   Tue Jan 17 17:10:16 2023 +0000

    media: pvrusb2: fix DVB_CORE dependency

    [ Upstream commit 53558de2b5c4f4ee6bfcfbe34e27071c2d0073d5 ]
```

```
    Now that DVB_CORE can be a loadable module, pvrusb2 can run into
    a link error:

    ld.lld: error: undefined symbol: dvb_module_probe
    >>> referenced by pvrusb2-devattr.c
    >>>               drivers/media/usb/pvrusb2/pvrusb2-devattr.o:(pvr2_lgdt3306a_attach) in
 archive vmlinux.a
    ld.lld: error: undefined symbol: dvb_module_release
    >>> referenced by pvrusb2-devattr.c
    >>>               drivers/media/usb/pvrusb2/pvrusb2-devattr.o:(pvr2_dual_fe_attach) in
 archive vmlinux.a

    Refine the Kconfig dependencies to avoid this case.

    Link: https://lore.kernel.org/linux-media/20230117171055.2714621-1-arnd@kernel.org
    Fixes: 7655c342dbc4 ("media: Kconfig: Make DVB_CORE=m possible when MEDIA_SUPPORT=y")
    Signed-off-by: Arnd Bergmann <arnd@arndb.de>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit e077f8ff76e725d826954a15f9b239ddc72b9d63
 Author: Eric Dumazet <edumazet@google.com>
 Date:   Thu May 11 11:47:49 2023 +0000

    tcp: fix possible sk_priority leak in tcp_v4_send_reset()

    [ Upstream commit 1e306ec49a1f206fd2cc89a42fac6e6f592a8cc1 ]

    When tcp_v4_send_reset() is called with @sk == NULL,
    we do not change ctl_sk->sk_priority, which could have been
    set from a prior invocation.

    Change tcp_v4_send_reset() to set sk_priority and sk_mark
    fields before calling ip_send_unicast_reply().

    This means tcp_v4_send_reset() and tcp_v4_send_ack()
    no longer have to clear ctl_sk->sk_mark after
    their call to ip_send_unicast_reply().

    Fixes: f6c0f5d209fa ("tcp: honor SO_PRIORITY in TIME_WAIT state")
    Signed-off-by: Eric Dumazet <edumazet@google.com>
    Cc: Antoine Tenart <atenart@kernel.org>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit 859a3997a2679df933b81c73f8e94120927485d7
 Author: Zhuang Shengen <zhuangshengen@huawei.com>
 Date:   Thu May 11 19:34:30 2023 +0800

    vsock: avoid to close connected socket after the timeout

    [ Upstream commit 6d4486efe9c69626cab423456169e250a5cd3af5 ]

    When client and server establish a connection through vsock,
    the client send a request to the server to initiate the connection,
    then start a timer to wait for the server's response. When the server's
    RESPONSE message arrives, the timer also times out and exits. The
    server's RESPONSE message is processed first, and the connection is
    established. However, the client's timer also times out, the original
    processing logic of the client is to directly set the state of this vsock
    to CLOSE and return ETIMEDOUT. It will not notify the server when the port
    is released, causing the server port remain.
    when client's vsock_connect timeout，it should check sk state is
    ESTABLISHED or not. if sk state is ESTABLISHED, it means the connection
    is established, the client should not set the sk state to CLOSE
```

```
    Note: I encountered this issue on kernel-4.18, which can be fixed by
    this patch. Then I checked the latest code in the community
    and found similar issue.

    Fixes: d021c344051a ("VSOCK: Introduce VM Sockets")
    Signed-off-by: Zhuang Shengen <zhuangshengen@huawei.com>
    Reviewed-by: Stefano Garzarella <sgarzare@redhat.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit da3a31a145e38dafb5cbd3c4d87ab427d100bef0
Author: Pieter Jansen van Vuuren <pieter.jansen-van-vuuren@amd.com>
Date:   Thu May 11 10:43:33 2023 +0100

    sfc: disable RXFCS and RXALL features by default

    [ Upstream commit 134120b066044399ef59564ff3ba66ab344cfc5b ]

    By default we would not want RXFCS and RXALL features enabled as they are
    mainly intended for debugging purposes. This does not stop users from
    enabling them later on as needed.

    Fixes: 8e57daf70671 ("sfc_ef100: RX path for EF100")
    Signed-off-by: Pieter Jansen van Vuuren <pieter.jansen-van-vuuren@amd.com>
    Co-developed-by: Edward Cree <ecree.xilinx@gmail.com>
    Signed-off-by: Edward Cree <ecree.xilinx@gmail.com>
    Reviewed-by: Martin Habets <habetsm.xilinx@gmail.com>
    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 7cdc413df9555fe70bae3b1ef6e7c7ca8679977c
Author: Jan Sokolowski <jan.sokolowski@intel.com>
Date:   Thu May 11 08:53:19 2023 -0700

    ice: Fix undersized tx_flags variable

    [ Upstream commit 9113302bb43cf7a6d5a414d49b29478e57451c86 ]

    As not all ICE_TX_FLAGS_* fit in current 16-bit limited
    tx_flags field that was introduced in the Fixes commit,
    VLAN-related information would be discarded completely.
    As such, creating a vlan and trying to run ping through
    would result in no traffic passing.

    Fix that by refactoring tx_flags variable into flags only and
    a separate variable that holds VLAN ID. As there is some space left,
    type variable can fit between those two. Pahole reports no size
    change to ice_tx_buf struct.

    Fixes: aa1d3faf71a6 ("ice: Robustify cleaning/completing XDP Tx buffers")
    Signed-off-by: Jan Sokolowski <jan.sokolowski@intel.com>
    Reviewed-by: Alexander Lobakin <aleksander.lobakin@intel.com>
    Signed-off-by: Tony Nguyen <anthony.l.nguyen@intel.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 47a5d0413f174091d8af89f3dfe45d86b6c7f92e
Author: Ryan C. Underwood <nemesis@icequake.net>
Date:   Thu May 11 12:32:21 2023 -0500

    ALSA: hda/realtek: Apply HP B&O top speaker profile to Pavilion 15

    [ Upstream commit 92553ee03166ef8fa978e7683f9f4af30c9c4e6b ]
```

```
    The Pavilion 15 line has B&O top speakers similar to the x360 and
    applying the same profile produces good sound.  Without this, the
    sound would be tinny and underpowered without either applying
    model=alc295-hp-x360 or booting another OS first.

    Signed-off-by: Ryan Underwood <nemesis@icequake.net>
    Fixes: 563785edfcef ("ALSA: hda/realtek - Add quirk entry for HP Pavilion 15")
    Link: https://lore.kernel.org/r/ZF0mpcMz3ezP9KQw@icequake.net
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 86658be1115e083f81e779fe0a6d85e7386aacd1
Author: Ryder Lee <ryder.lee@mediatek.com>
Date:   Mon Apr 24 05:39:06 2023 +0800

    wifi: mt76: connac: fix stats->tx_bytes calculation

    [ Upstream commit c7ab7a29ef5c0779574120d922256ce4651555d3 ]

    The stats->tx_bytes shall subtract retry byte from tx byte.

    Fixes: 43eaa3689507 ("wifi: mt76: add PPDU based TxS support for WED device")
    Signed-off-by: Ryder Lee <ryder.lee@mediatek.com>
    Signed-off-by: Kalle Valo <kvalo@kernel.org>
    Link:
https://lore.kernel.org/r/b3cd45596943cf5a06b2e08e2fe732ab0b51311b.1682285873.git.ryder.lee@
mediatek.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 67148395efa2c1fb20e98fca359b20e7a6c81fe4
Author: Dan Carpenter <dan.carpenter@linaro.org>
Date:   Tue May 9 12:07:11 2023 +0300

    ALSA: firewire-digi00x: prevent potential use after free

    [ Upstream commit c0e72058d5e21982e61a29de6b098f7c1f0db498 ]

    This code was supposed to return an error code if init_stream()
    failed, but it instead freed dg00x->rx_stream and returned success.
    This potentially leads to a use after free.

    Fixes: 9a08067ec318 ("ALSA: firewire-digi00x: support AMDTP domain")
    Signed-off-by: Dan Carpenter <dan.carpenter@linaro.org>
    Link: https://lore.kernel.org/r/c224cbd5-d9e2-4cd4-9bcf-2138eb1d35c6@kili.mountain
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ce4e77f2dcf0eb30afabeaa0e41aab2bc3f6aacf
Author: Grygorii Strashko <grygorii.strashko@ti.com>
Date:   Wed May 10 18:21:39 2023 +0530

    net: phy: dp83867: add w/a for packet errors seen with short cables

    [ Upstream commit 0b01db274028f5acd207332686ffc92ac77491ac ]

    Introduce the W/A for packet errors seen with short cables (<1m) between
    two DP83867 PHYs.

    The W/A recommended by DM requires FFE Equalizer Configuration tuning by
    writing value 0x0E81 to DSP_FFE_CFG register (0x012C), surrounded by hard
    and soft resets as follows:

    write_reg(0x001F, 0x8000); //hard reset
    write_reg(DSP_FFE_CFG, 0x0E81);
```

```
    write_reg(0x001F, 0x4000); //soft reset

    Since  DP83867 PHY DM says "Changing this register to 0x0E81, will not
    affect Long Cable performance.", enable the W/A by default.

    Fixes: 2a10154abcb7 ("net: phy: dp83867: Add TI dp83867 phy")
    Signed-off-by: Grygorii Strashko <grygorii.strashko@ti.com>
    Signed-off-by: Siddharth Vadapalli <s-vadapalli@ti.com>
    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 e02d8d5b1602689b98d9b91550a11b9b57baedbe
Author: Uwe Kleine-König <u.kleine-koenig@pengutronix.de>
Date:   Wed May 10 22:00:20 2023 +0200

    net: fec: Better handle pm_runtime_get() failing in .remove()

    [ Upstream commit f816b9829b19394d318e01953aa3b2721bca040d ]

    In the (unlikely) event that pm_runtime_get() (disguised as
    pm_runtime_resume_and_get()) fails, the remove callback returned an
    error early. The problem with this is that the driver core ignores the
    error value and continues removing the device. This results in a
    resource leak. Worse the devm allocated resources are freed and so if a
    callback of the driver is called later the register mapping is already
    gone which probably results in a crash.

    Fixes: a31eda65ba21 ("net: fec: fix clock count mis-match")
    Signed-off-by: Uwe Kleine-König <u.kleine-koenig@pengutronix.de>
    Reviewed-by: Andrew Lunn <andrew@lunn.ch>
    Link: https://lore.kernel.org/r/20230510200020.1534610-1-u.kleine-koenig@pengutronix.de
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9643174ca5f1df70527a000a4937a9c4f5afe85f
Author: Jiri Pirko <jiri@resnulli.us>
Date:   Wed May 10 16:46:21 2023 +0200

    devlink: change per-devlink netdev notifier to static one

    [ Upstream commit e93c9378e33f68b61ea9318580d841caa22fb9ea ]

    The commit 565b4824c39f ("devlink: change port event netdev notifier
    from per-net to global") changed original per-net notifier to be
    per-devlink instance. That fixed the issue of non-receiving events
    of netdev uninit if that moved to a different namespace.
    That worked fine in -net tree.

    However, later on when commit ee75f1fc44dd ("net/mlx5e: Create
    separate devlink instance for ethernet auxiliary device") and
    commit 72ed5d5624af ("net/mlx5: Suspend auxiliary devices only in
    case of PCI device suspend") were merged, a deadlock was introduced
    when removing a namespace with devlink instance with another nested
    instance.

    Here there is the bad flow example resulting in deadlock with mlx5:
    net_cleanup_work -> cleanup_net (takes down_read(&pernet_ops_rwsem) ->
    devlink_pernet_pre_exit() -> devlink_reload() ->
    mlx5_devlink_reload_down() -> mlx5_unload_one_devl_locked() ->
    mlx5_detach_device() -> del_adev() -> mlx5e_remove() ->
    mlx5e_destroy_devlink() -> devlink_free() ->
    unregister_netdevice_notifier() (takes down_write(&pernet_ops_rwsem)

    Steps to reproduce:
```

```
    $ modprobe mlx5_core
    $ ip netns add ns1
    $ devlink dev reload pci/0000:08:00.0 netns ns1
    $ ip netns del ns1

    Resolve this by converting the notifier from per-devlink instance to
    a static one registered during init phase and leaving it registered
    forever. Use this notifier for all devlink port instances created
    later on.

    Note what a tree needs this fix only in case all of the cited fixes
    commits are present.

    Reported-by: Moshe Shemesh <moshe@nvidia.com>
    Fixes: 565b4824c39f ("devlink: change port event netdev notifier from per-net to
global")
    Fixes: ee75f1fc44dd ("net/mlx5e: Create separate devlink instance for ethernet auxiliary
device")
    Fixes: 72ed5d5624af ("net/mlx5: Suspend auxiliary devices only in case of PCI device
suspend")
    Signed-off-by: Jiri Pirko <jiri@nvidia.com>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Link: https://lore.kernel.org/r/20230510144621.932017-1-jiri@resnulli.us
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit eb67cc136e073f37dc4240128d656df8d19b1468
Author: Andrea Mayer <andrea.mayer@uniroma2.it>
Date:   Wed May 10 13:16:38 2023 +0200

    selftets: seg6: disable rp_filter by default in srv6_end_dt4_l3vpn_test

    [ Upstream commit f97b8401e0deb46ad1e4245c21f651f64f55aaa6 ]

    On some distributions, the rp_filter is automatically set (=1) by
    default on a netdev basis (also on VRFs).
    In an SRv6 End.DT4 behavior, decapsulated IPv4 packets are routed using
    the table associated with the VRF bound to that tunnel. During lookup
    operations, the rp_filter can lead to packet loss when activated on the
    VRF.
    Therefore, we chose to make this selftest more robust by explicitly
    disabling the rp_filter during tests (as it is automatically set by some
    Linux distributions).

    Fixes: 2195444e09b4 ("selftests: add selftest for the SRv6 End.DT4 behavior")
    Reported-by: Hangbin Liu <liuhangbin@gmail.com>
    Signed-off-by: Andrea Mayer <andrea.mayer@uniroma2.it>
    Tested-by: Hangbin Liu <liuhangbin@gmail.com>
    Reviewed-by: David Ahern <dsahern@kernel.org>
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9918bdef624aa4cb87ba8aaee78da628d032f9a9
Author: Andrea Mayer <andrea.mayer@uniroma2.it>
Date:   Wed May 10 13:16:37 2023 +0200

    selftests: seg6: disable DAD on IPv6 router cfg for srv6_end_dt4_l3vpn_test

    [ Upstream commit 21a933c79a33add3612808f3be4ad65dd4dc026b ]

    The srv6_end_dt4_l3vpn_test instantiates a virtual network consisting of
    several routers (rt-1, rt-2) and hosts.
    When the IPv6 addresses of rt-{1,2} routers are configured, the Deduplicate
    Address Detection (DAD) kicks in when enabled in the Linux distros running
    the selftests. DAD is used to check whether an IPv6 address is already
```

```
    assigned in a network. Such a mechanism consists of sending an ICMPv6 Echo
    Request and waiting for a reply.
    As the DAD process could take too long to complete, it may cause the
    failing of some tests carried out by the srv6_end_dt4_l3vpn_test script.

    To make the srv6_end_dt4_l3vpn_test more robust, we disable DAD on routers
    since we configure the virtual network manually and do not need any address
    deduplication mechanism at all.

    Fixes: 2195444e09b4 ("selftests: add selftest for the SRv6 End.DT4 behavior")
    Signed-off-by: Andrea Mayer <andrea.mayer@uniroma2.it>
    Reviewed-by: David Ahern <dsahern@kernel.org>
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e304ff331f7a88d8d138536c8511f4ca73804acd
Author: Rob Clark <robdclark@chromium.org>
Date:   Tue May 9 13:30:41 2023 -0700

    drm/msm: Fix submit error-path leaks

    [ Upstream commit 68dc6c2d5eec45515855cce99256162f45651a0b ]

    For errors after msm_submitqueue_get(), we need to drop the submitqueue
    reference.  Additionally after get_unused_fd() we need to drop the fd.
    The ordering for dropping the queue lock and put_unused_fd() is not
    important, so just move this all into out_post_unlock.

    v2: Only drop queue ref if submit doesn't take it
    v3: Fix unitialized submit ref in error path
    v4: IS_ERR_OR_NULL()

    Reported-by: pinkperfect2021@gmail.com
    Fixes: f0de40a131d9 drm/msm: ("Reorder lock vs submit alloc")
    Signed-off-by: Rob Clark <robdclark@chromium.org>
    Patchwork: https://patchwork.freedesktop.org/patch/536073/
    Link: https://lore.kernel.org/r/20230509203041.440619-1-robdclark@gmail.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 79f19d3480866d716be15fe93aa5b96900fa6b7c
Author: Tobias Brunner <tobias@strongswan.org>
Date:   Tue May 9 11:00:06 2023 +0200

    af_key: Reject optional tunnel/BEET mode templates in outbound policies

    [ Upstream commit cf3128a7aca55b2eefb68281d44749c683bdc96f ]

    xfrm_state_find() uses `encap_family` of the current template with
    the passed local and remote addresses to find a matching state.
    If an optional tunnel or BEET mode template is skipped in a mixed-family
    scenario, there could be a mismatch causing an out-of-bounds read as
    the addresses were not replaced to match the family of the next template.

    While there are theoretical use cases for optional templates in outbound
    policies, the only practical one is to skip IPComp states in inbound
    policies if uncompressed packets are received that are handled by an
    implicitly created IPIP state instead.

    Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
    Signed-off-by: Tobias Brunner <tobias@strongswan.org>
    Acked-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Steffen Klassert <steffen.klassert@secunet.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 43faf151daf5d2323e9cd3a3f2f5410b7eb8c618
```

```
Author: Tobias Brunner <tobias@strongswan.org>
Date:   Tue May 9 10:59:58 2023 +0200

    xfrm: Reject optional tunnel/BEET mode templates in outbound policies

    [ Upstream commit 3d776e31c841ba2f69895d2255a49320bec7cea6 ]

    xfrm_state_find() uses `encap_family` of the current template with
    the passed local and remote addresses to find a matching state.
    If an optional tunnel or BEET mode template is skipped in a mixed-family
    scenario, there could be a mismatch causing an out-of-bounds read as
    the addresses were not replaced to match the family of the next template.

    While there are theoretical use cases for optional templates in outbound
    policies, the only practical one is to skip IPComp states in inbound
    policies if uncompressed packets are received that are handled by an
    implicitly created IPIP state instead.

    Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
    Signed-off-by: Tobias Brunner <tobias@strongswan.org>
    Acked-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Steffen Klassert <steffen.klassert@secunet.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ba175763f780cf259bc3dbea406fa963c296f8fb
Author: Wyes Karny <wyes.karny@amd.com>
Date:   Thu May 4 06:25:44 2023 +0000

    cpupower: Make TSC read per CPU for Mperf monitor

    [ Upstream commit c2adb1877b76fc81ae041e1db1a6ed2078c6746b ]

    System-wide TSC read could cause a drift in C0 percentage calculation.
    Because if first TSC is read and then one by one mperf is read for all
    cpus, this introduces drift between mperf reading of later CPUs and TSC
    reading.  To lower this drift read TSC per CPU and also just after mperf
    read.  This technique improves C0 percentage calculation in Mperf monitor.

    Before fix: (System 100% busy)

                 | Mperf              || RAPL        || Idle_Stats
     PKG|CORE| CPU| C0   | Cx   | Freq  || pack | core  || POLL | C1   | C2
       0|   0|   0| 87.15| 12.85|  2695||168659003|3970468||  0.00|  0.00| 0.00
       0|   0| 256| 84.62| 15.38|  2695||168659003|3970468||  0.00|  0.00| 0.00
       0|   1|   1| 87.15| 12.85|  2695||168659003|3970468||  0.00|  0.00| 0.00
       0|   1| 257| 84.08| 15.92|  2695||168659003|3970468||  0.00|  0.00| 0.00
       0|   2|   2| 86.61| 13.39|  2695||168659003|3970468||  0.00|  0.00| 0.00
       0|   2| 258| 83.26| 16.74|  2695||168659003|3970468||  0.00|  0.00| 0.00
       0|   3|   3| 86.61| 13.39|  2695||168659003|3970468||  0.00|  0.00| 0.00
       0|   3| 259| 83.60| 16.40|  2695||168659003|3970468||  0.00|  0.00| 0.00
       0|   4|   4| 86.33| 13.67|  2695||168659003|3970468||  0.00|  0.00| 0.00
       0|   4| 260| 83.33| 16.67|  2695||168659003|3970468||  0.00|  0.00| 0.00
       0|   5|   5| 86.06| 13.94|  2695||168659003|3970468||  0.00|  0.00| 0.00
       0|   5| 261| 83.05| 16.95|  2695||168659003|3970468||  0.00|  0.00| 0.00
       0|   6|   6| 85.51| 14.49|  2695||168659003|3970468||  0.00|  0.00| 0.00

    After fix: (System 100% busy)

                 | Mperf              || RAPL        || Idle_Stats
     PKG|CORE| CPU| C0   | Cx   | Freq  || pack | core  || POLL | C1   | C2
       0|   0|   0| 98.03|  1.97|  2415||163295480|3811189||  0.00|  0.00| 0.00
       0|   0| 256| 98.50|  1.50|  2394||163295480|3811189||  0.00|  0.00| 0.00
       0|   1|   1| 99.99|  0.01|  2401||163295480|3811189||  0.00|  0.00| 0.00
       0|   1| 257| 99.99|  0.01|  2375||163295480|3811189||  0.00|  0.00| 0.00
       0|   2|   2| 99.99|  0.01|  2401||163295480|3811189||  0.00|  0.00| 0.00
```

```
        0|   2| 258|100.00|  0.00|  2401||163295480|3811189||  0.00|  0.00| 0.00
        0|   3|   3|100.00|  0.00|  2401||163295480|3811189||  0.00|  0.00| 0.00
        0|   3| 259| 99.99|  0.01|  2435||163295480|3811189||  0.00|  0.00| 0.00
        0|   4|   4|100.00|  0.00|  2401||163295480|3811189||  0.00|  0.00| 0.00
        0|   4| 260|100.00|  0.00|  2435||163295480|3811189||  0.00|  0.00| 0.00
        0|   5|   5| 99.99|  0.01|  2401||163295480|3811189||  0.00|  0.00| 0.00
        0|   5| 261|100.00|  0.00|  2435||163295480|3811189||  0.00|  0.00| 0.00
        0|   6|   6|100.00|  0.00|  2401||163295480|3811189||  0.00|  0.00| 0.00
        0|   6| 262|100.00|  0.00|  2435||163295480|3811189||  0.00|  0.00| 0.00

    Cc: Thomas Renninger <trenn@suse.com>
    Cc: Shuah Khan <shuah@kernel.org>
    Cc: Dominik Brodowski <linux@dominikbrodowski.net>

    Fixes: 7fe2f6399a84 ("cpupowerutils - cpufrequtils extended with quite some features")
    Signed-off-by: Wyes Karny <wyes.karny@amd.com>
    Signed-off-by: Shuah Khan <skhan@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cec323486a8c0dee9707e0acc879c94a0aad5bfb
Author: Shengjiu Wang <shengjiu.wang@nxp.com>
Date:   Mon May 8 18:16:36 2023 +0800

    ASoC: fsl_micfil: Fix error handler with pm_runtime_enable

    [ Upstream commit 17955aba7877a4494d8093ae5498e19469b01d57 ]

    There is error message when defer probe happens:

    fsl-micfil-dai 30ca0000.micfil: Unbalanced pm_runtime_enable!

    Fix the error handler with pm_runtime_enable and add
    fsl_micfil_remove() for pm_runtime_disable.

    Fixes: 47a70e6fc9a8 ("ASoC: Add MICFIL SoC Digital Audio Interface driver.")
    Signed-off-by: Shengjiu Wang <shengjiu.wang@nxp.com
    Link: https://lore.kernel.org/r/1683540996-6136-1-git-send-email-shengjiu.wang@nxp.com
    Signed-off-by: Mark Brown <broonie@kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9b59f5c4911e87264507e0934cd2bb277390c560
Author: Chuck Lever <chuck.lever@oracle.com>
Date:   Tue May 2 14:59:10 2023 -0400

    SUNRPC: Fix encoding of accepted but unsuccessful RPC replies

    [ Upstream commit 29cd2927fb914cc53b5ba4f67d2b74695c994ba4 ]

    Jiri Slaby says:
    > I bisected to this ... as it breaks nfs3-only servers in 6.3.
    > I.e. /etc/nfs.conf containing:
    > [nfsd]
    > vers4=no
    >
    > The client sees:
    >  mount("10.0.2.15:/tmp", "/mnt", "nfs", 0, "vers=4.2,addr=10.0.2.15,clientad"...) = -1
EIO (Input/output error)
    >  write(2, "mount.nfs: mount system call fai"..., 45
    >  mount.nfs: mount system call failed for /mnt
    >
    > And the kernel says:
    >  nfs4_discover_server_trunking unhandled error -5. Exiting with error EIO

    Reported-by: Jiri Slaby <jirislaby@kernel.org>
    Link: https://bugzilla.suse.com/show_bug.cgi?id=1210995
```

```
    Fixes: 4bcf0343e8a6 ("SUNRPC: Set rq_accept_statp inside ->accept methods")
    Tested-by: Jiri Slaby <jirislaby@kernel.org>
    Signed-off-by: Chuck Lever <chuck.lever@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a978c6449266b05d0c92aa9ff51c92f8ca31ee62
Author: Jianhua Lu <lujianhua000@gmail.com>
Date:   Thu Apr 27 20:21:32 2023 +0800

    dt-bindings: display/msm: dsi-controller-main: Document qcom, master-dsi and qcom, sync-
dual-dsi

    [ Upstream commit ca29699a57ecee6084a4056f5bfd6f11dd359a71 ]

    This fixes warning:
      sm8250-xiaomi-elish-csot.dtb: dsi@ae94000: Unevaluated properties are not allowed
('qcom,master-dsi', 'qcom,sync-dual-dsi' were unexpected)

    Reviewed-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Acked-by: Rob Herring <robh@kernel.org>
    Signed-off-by: Jianhua Lu <lujianhua000@gmail.com>
    Fixes: 4dbe55c97741 ("dt-bindings: msm: dsi: add yaml schemas for DSI bindings")
    Reviewed-by: Abhinav Kumar <quic_abhinavk@quicinc.com>
    Patchwork: https://patchwork.freedesktop.org/patch/534306/
    Link: https://lore.kernel.org/r/20230427122132.24840-1-lujianhua000@gmail.com
    Signed-off-by: Abhinav Kumar <quic_abhinavk@quicinc.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 878d4ff2ae73e13f096604b28d9cf48324c23ccc
Author: Marijn Suijten <marijn.suijten@somainline.org>
Date:   Thu Apr 27 00:37:22 2023 +0200

    drm/msm/dpu: Remove duplicate register defines from INTF

    [ Upstream commit 202c044203ac5860e3025169105368d99f9bc6a2 ]

    The INTF_FRAME_LINE_COUNT_EN, INTF_FRAME_COUNT and INTF_LINE_COUNT
    registers are already defined higher up, in the right place when sorted
    numerically.

    Fixes: 25fdd5933e4c ("drm/msm: Add SDM845 DPU support")
    Signed-off-by: Marijn Suijten <marijn.suijten@somainline.org>
    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Reviewed-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Reviewed-by: Abhinav Kumar <quic_abhinavk@quicinc.com>
    Patchwork: https://patchwork.freedesktop.org/patch/534231/
    Link: https://lore.kernel.org/r/20230411-dpu-intf-te-v4-8-27ce1a5ab5c6@somainline.org
    Signed-off-by: Abhinav Kumar <quic_abhinavk@quicinc.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9e1a191e8890d2af0b0fc5409430705e97bcb71a
Author: Marijn Suijten <marijn.suijten@somainline.org>
Date:   Thu Apr 27 00:37:19 2023 +0200

    drm/msm/dpu: Fix PP_BLK_DIPHER -> DITHER typo

    [ Upstream commit 701f69183d4d52533fb2af0d6948b7d1b00d1a09 ]

    SM8550 exclusively has a DITHER sub-block inside the PINGPONG block and
    no other registers, hence the DITHER name of the macro and a
    corresponding PINGPONG block length of zero.  However, the PP_BLK_ macro
    name was typo'd to DIPHER rather than DITHER.

    Fixes: efcd0107727c ("drm/msm/dpu: add support for SM8550")
    Signed-off-by: Marijn Suijten <marijn.suijten@somainline.org>
```

```
    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Reviewed-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Reviewed-by: Abhinav Kumar <quic_abhinavk@quicinc.com>
    Patchwork: https://patchwork.freedesktop.org/patch/534214/
    Link: https://lore.kernel.org/r/20230411-dpu-intf-te-v4-5-27ce1a5ab5c6@somainline.org
    Signed-off-by: Abhinav Kumar <quic_abhinavk@quicinc.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 97ae8728e7d250adf519595b9f90df5f8b325dcc
Author: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
Date:   Tue Apr 4 16:05:47 2023 +0300

    drm/msm/dpu: split SM8550 catalog entry to the separate file

    [ Upstream commit 9cc547933636fa0656b52691f24373ccd7ec61cb ]

    Reviewed-by: Konrad DYbcio <konrad.dybcio@linaro.org>
    Signed-off-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Patchwork: https://patchwork.freedesktop.org/patch/530824/
    Link: https://lore.kernel.org/r/20230404130622.509628-8-dmitry.baryshkov@linaro.org
    Signed-off-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Stable-dep-of: 701f69183d4d ("drm/msm/dpu: Fix PP_BLK_DIPHER -> DITHER typo")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2edefb97c30a31b5ed71e75b2cde345e64f5fad3
Author: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
Date:   Tue Apr 4 16:05:46 2023 +0300

    drm/msm/dpu: move UBWC/memory configuration to separate struct

    [ Upstream commit fbbd8cce803a2ca86ae20fe37b1642571e9dd971 ]

    UBWC and highest bank settings differ slightly between different DPU
    units of the same generation, while the dpu_caps and dpu_mdp_cfg are
    much more stable. To ease configuration reuse move ubwc_swizzle and
    highest_bank_bit data to separate structure.

    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Reviewed-by: Abhinav Kumar <quic_abhinavk@quicinc.com>
    Signed-off-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Patchwork: https://patchwork.freedesktop.org/patch/530820/
    Link: https://lore.kernel.org/r/20230404130622.509628-7-dmitry.baryshkov@linaro.org
    Signed-off-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Stable-dep-of: 701f69183d4d ("drm/msm/dpu: Fix PP_BLK_DIPHER -> DITHER typo")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e67bb5f5c7a228574c0927629a06386af4162681
Author: Konrad Dybcio <konrad.dybcio@linaro.org>
Date:   Tue Apr 4 16:05:43 2023 +0300

    drm/msm/dpu: Allow variable INTF_BLK size

    [ Upstream commit 8399a5ff18dc634d3245d993ee8a906b5c6ffba3 ]

    These blocks are of variable length on different SoCs. Set the
    correct values where I was able to retrieve it from downstream
    DTs and leave the old defaults (0x280) otherwise.

    Signed-off-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    [DB: fixed some lengths, split the INTF changes away]
    Signed-off-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Reviewed-by: Abhinav Kumar <quic_abhinavk@quicinc.com>
    Patchwork: https://patchwork.freedesktop.org/patch/530816/
    Link: https://lore.kernel.org/r/20230404130622.509628-4-dmitry.baryshkov@linaro.org
    Signed-off-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
```

```
    Stable-dep-of: 701f69183d4d ("drm/msm/dpu: Fix PP_BLK_DIPHER -> DITHER typo")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d1c896a0a6e53cccfb46bc26ec35e546cf0f66bf
Author: Konrad Dybcio <konrad.dybcio@linaro.org>
Date:   Tue Apr 4 16:05:42 2023 +0300

    drm/msm/dpu: Allow variable SSPP_BLK size

    [ Upstream commit 8f940ddbc4f1b6cc454d9487f9cd735310fb0cfa ]

    These blocks are of variable length on different SoCs. Set the
    correct values where I was able to retrieve it from downstream
    DTs and leave the old defaults (0x1c8) otherwise.

    Signed-off-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    [DB: fixed some of lengths, split the INTF changes away]
    Signed-off-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Patchwork: https://patchwork.freedesktop.org/patch/530814/
    Link: https://lore.kernel.org/r/20230404130622.509628-3-dmitry.baryshkov@linaro.org
    Signed-off-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Stable-dep-of: 701f69183d4d ("drm/msm/dpu: Fix PP_BLK_DIPHER -> DITHER typo")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dd8acdc717fac5afd485a848c604f9a27e4fcea8
Author: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
Date:   Thu Mar 16 19:16:51 2023 +0300

    drm/msm/dpu: drop smart_dma_rev from dpu_caps

    [ Upstream commit dcb3f7c9042d1c1aa637b58d47bd45c00b2ac153 ]

    The code doesn't use dpu_caps::smart_dma_rev field. It checks if the
    corresponding feature is enabled in the SSPP features. Drop the
    smart_dma_rev field completely.

    Reviewed-by: Abhinav Kumar <quic_abhinavk@quicinc.com>
    Signed-off-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Patchwork: https://patchwork.freedesktop.org/patch/527369/
    Link: https://lore.kernel.org/r/20230316161653.4106395-31-dmitry.baryshkov@linaro.org
    Signed-off-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Stable-dep-of: 701f69183d4d ("drm/msm/dpu: Fix PP_BLK_DIPHER -> DITHER typo")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1d39a7bf72a5ad0ffce23452bc5107a7e3e19147
Author: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
Date:   Thu Mar 16 19:16:49 2023 +0300

    drm/msm/dpu: populate SmartDMA features in hw catalog

    [ Upstream commit 8b409996ebdce009777dfb17e542c06f749b02d5 ]

    Downstream driver uses dpu->caps->smart_dma_rev to update
    sspp->cap->features with the bit corresponding to the supported SmartDMA
    version. Upstream driver does not do this, resulting in SSPP subdriver
    not enabling setup_multirect callback. Add corresponding SmartDMA SSPP
    feature bits to dpu hw catalog.

    Per Abhinav's request enable the SmartDMA features only on the platforms
    where the multirect was actually verified visually (sdm845 and sm8250).
    An (untested) enablement on the rest of the platforms comes in the next
    patch.

    Reviewed-by: Abhinav Kumar <quic_abhinavk@quicinc.com>
    Signed-off-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
```

```
    Patchwork: https://patchwork.freedesktop.org/patch/527362/
    Link: https://lore.kernel.org/r/20230316161653.4106395-29-dmitry.baryshkov@linaro.org
    Signed-off-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Stable-dep-of: 701f69183d4d ("drm/msm/dpu: Fix PP_BLK_DIPHER -> DITHER typo")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 899a8fd75b60560c3cc03afbe51a775ed74976bb
Author: Marijn Suijten <marijn.suijten@somainline.org>
Date:   Thu Apr 27 00:37:18 2023 +0200

    drm/msm/dpu: Reindent REV_7xxx interrupt masks with tabs

    [ Upstream commit 85340c0256f9b85b47c5867e411df37d76df5858 ]

    Use tabs for consistency with the other interrupt register definitions,
    rather than spaces.

    Fixes: ed6154a136e4 ("drm/msm/disp/dpu1: add intf offsets for SC7280 target")
    Fixes: 89688e2119b2 ("drm/msm/dpu: Add more of the INTF interrupt regions")
    Fixes: 4a352c2fc15a ("drm/msm/dpu: Introduce SC8280XP")
    Signed-off-by: Marijn Suijten <marijn.suijten@somainline.org>
    Reviewed-by: Abhinav Kumar <quic_abhinavk@quicinc.com>
    Reviewed-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Patchwork: https://patchwork.freedesktop.org/patch/534212/
    Link: https://lore.kernel.org/r/20230411-dpu-intf-te-v4-4-27ce1a5ab5c6@somainline.org
    Signed-off-by: Abhinav Kumar <quic_abhinavk@quicinc.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1eafafcf09e6af809273278aa4aa0a921b558e46
Author: Marijn Suijten <marijn.suijten@somainline.org>
Date:   Thu Apr 27 00:37:17 2023 +0200

    drm/msm/dpu: Move non-MDP_TOP INTF_INTR offsets out of hwio header

    [ Upstream commit e9d9ce5462fecdeefec87953de71df4d025cbc72 ]

    These offsets do not fall under the MDP TOP block and do not fit the
    comment right above.  Move them to dpu_hw_interrupts.c next to the
    repsective MDP_INTF_x_OFF interrupt block offsets.

    Fixes: 25fdd5933e4c ("drm/msm: Add SDM845 DPU support")
    Signed-off-by: Marijn Suijten <marijn.suijten@somainline.org>
    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Reviewed-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Reviewed-by: Abhinav Kumar <quic_abhinavk@quicinc.com>
    Patchwork: https://patchwork.freedesktop.org/patch/534203/
    Link: https://lore.kernel.org/r/20230411-dpu-intf-te-v4-3-27ce1a5ab5c6@somainline.org
    Signed-off-by: Abhinav Kumar <quic_abhinavk@quicinc.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 392940353eb2a38be6f60dabcd0627d31e9ab43c
Author: Marijn Suijten <marijn.suijten@somainline.org>
Date:   Wed Apr 26 01:11:09 2023 +0200

    drm/msm/dpu: Assign missing writeback log_mask

    [ Upstream commit a432fc31f03db2546a48bcf5dd69ca28ceb732bf ]

    The WB debug log mask ended up never being assigned, leading to writes
    to this block to never be logged even if the mask is enabled in
    dpu_hw_util_log_mask via debugfs.

    Fixes: 84a33d0fd921 ("drm/msm/dpu: add dpu_hw_wb abstraction for writeback blocks")
    Signed-off-by: Marijn Suijten <marijn.suijten@somainline.org>
    Reviewed-by: Abhinav Kumar <quic_abhinavk@quicinc.com>
```

```
    Reviewed-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Patchwork: https://patchwork.freedesktop.org/patch/533860/
    Link: https://lore.kernel.org/r/20230418-dpu-drop-useless-for-lookup-v3-1-
e8d869eea455@somainline.org
    Signed-off-by: Abhinav Kumar <quic_abhinavk@quicinc.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 56aadce80466dd18162701ad14de7d66f0e9b8a5
Author: Srinivas Kandagatla <srinivas.kandagatla@linaro.org>
Date:   Fri Apr 21 15:56:57 2023 +0100

    drm/msm/dp: unregister audio driver during unbind

    [ Upstream commit 85c636284cb63b7740b4ae98881ace92158068d3 ]

    while binding the code always registers a audio driver, however there
    is no corresponding unregistration done in unbind. This leads to multiple
    redundant audio platform devices if dp_display_bind and dp_display_unbind
    happens multiple times during startup. On X13s platform this resulted in
    6 to 9 audio codec device instead of just 3 codec devices for 3 dp ports.

    Fix this by unregistering codecs on unbind.

    Signed-off-by: Srinivas Kandagatla <srinivas.kandagatla@linaro.org>
    Fixes: d13e36d7d222 ("drm/msm/dp: add audio support for Display Port on MSM")
    Reviewed-by: Abhinav Kumar <quic_abhinavk@quicinc.com>
    Patchwork: https://patchwork.freedesktop.org/patch/533324/
    Link: https://lore.kernel.org/r/20230421145657.12186-1-srinivas.kandagatla@linaro.org
    Signed-off-by: Abhinav Kumar <quic_abhinavk@quicinc.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8b2e9aaf5736a8f0d351a5f7de1ee760ed59724b
Author: Martin Willi <martin@strongswan.org>
Date:   Tue Apr 25 09:46:18 2023 +0200

    Revert "Fix XFRM-I support for nested ESP tunnels"

    [ Upstream commit 5fc46f94219d1d103ffb5f0832be9da674d85a73 ]

    This reverts commit b0355dbbf13c0052931dd14c38c789efed64d3de.

    The reverted commit clears the secpath on packets received via xfrm interfaces
    to support nested IPsec tunnels. This breaks Netfilter policy matching using
    xt_policy in the FORWARD chain, as the secpath is missing during forwarding.
    Additionally, Benedict Wong reports that it breaks Transport-in-Tunnel mode.

    Fix this regression by reverting the commit until we have a better approach
    for nested IPsec tunnels.

    Fixes: b0355dbbf13c ("Fix XFRM-I support for nested ESP tunnels")
    Link: https://lore.kernel.org/netdev/20230412085615.124791-1-martin@strongswan.org/
    Signed-off-by: Martin Willi <martin@strongswan.org>
    Signed-off-by: Steffen Klassert <steffen.klassert@secunet.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7d16c515059b3746f2d6a24a74c3ba786a68c2a1
Author: Leon Romanovsky <leon@kernel.org>
Date:   Wed Apr 19 15:19:08 2023 +0300

    xfrm: Fix leak of dev tracker

    [ Upstream commit ec8f32ad9a65a8cbb465b69e154aaec9d2fe45c4 ]

    At the stage of direction checks, the netdev reference tracker is
    already initialized, but released with wrong *_put() call.
```

```
    Fixes: 919e43fad516 ("xfrm: add an interface to offload policy")
    Signed-off-by: Leon Romanovsky <leonro@nvidia.com>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Reviewed-by: Eric Dumazet <edumazet@google.com>
    Signed-off-by: Steffen Klassert <steffen.klassert@secunet.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit fff0a3bfbcf89b58aa14bf020fe410d98c30a09d
Author: Leon Romanovsky <leon@kernel.org>
Date:   Wed Apr 19 15:19:07 2023 +0300

    xfrm: release all offloaded policy memory

    [ Upstream commit 94b95dfaa814f565d92f5a65f0ff12a483095522 ]

    Failure to add offloaded policy will cause to the following
    error once user will try to reload driver.

    Unregister_netdevice: waiting for eth3 to become free. Usage count = 2

    This was caused by xfrm_dev_policy_add() which increments reference
    to net_device. That reference was supposed to be decremented
    in xfrm_dev_policy_free(). However the latter wasn't called.

     unregister_netdevice: waiting for eth3 to become free. Usage count = 2
     leaked reference.
      xfrm_dev_policy_add+0xff/0x3d0
      xfrm_policy_construct+0x352/0x420
      xfrm_add_policy+0x179/0x320
      xfrm_user_rcv_msg+0x1d2/0x3d0
      netlink_rcv_skb+0xe0/0x210
      xfrm_netlink_rcv+0x45/0x50
      netlink_unicast+0x346/0x490
      netlink_sendmsg+0x3b0/0x6c0
      sock_sendmsg+0x73/0xc0
      sock_write_iter+0x13b/0x1f0
      vfs_write+0x528/0x5d0
      ksys_write+0x120/0x150
      do_syscall_64+0x3d/0x90
      entry_SYSCALL_64_after_hwframe+0x46/0xb0

    Fixes: 919e43fad516 ("xfrm: add an interface to offload policy")
    Signed-off-by: Leon Romanovsky <leonro@nvidia.com>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Reviewed-by: Eric Dumazet <edumazet@google.com>
    Signed-off-by: Steffen Klassert <steffen.klassert@secunet.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f494ef925bbff082ace8a78dd83a487693f94f1d
Author: Sabrina Dubroca <sd@queasysnail.net>
Date:   Tue Apr 4 15:12:16 2023 +0200

    xfrm: don't check the default policy if the policy allows the packet

    [ Upstream commit 430cac487400494c19a8b85299e979bb07b4671f ]

    The current code doesn't let a simple "allow" policy counteract a
    default policy blocking all incoming packets:

        ip x p setdefault in block
        ip x p a src 192.168.2.1/32 dst 192.168.2.2/32 dir in action allow

    At this stage, we have an allow policy (with or without transforms)
    for this packet. It doesn't matter what the default policy says, since
```

```
    the policy we looked up lets the packet through. The case of a
    blocking policy is already handled separately, so we can remove this
    check.

    Fixes: 2d151d39073a ("xfrm: Add possibility to set the default to block if we have no
policy")
    Signed-off-by: Sabrina Dubroca <sd@queasysnail.net>
    Signed-off-by: Steffen Klassert <steffen.klassert@secunet.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c5ece3426c8da7d38fb6d76ad39679c54a88cb46
Author: Guilherme G. Piccoli <gpiccoli@igalia.com>
Date:   Mon Feb 20 18:11:05 2023 -0300

    parisc: Replace regular spinlock with spin_trylock on panic path

    [ Upstream commit 829632dae8321787525ee37dc4828bbe6edafdae ]

    The panic notifiers' callbacks execute in an atomic context, with
    interrupts/preemption disabled, and all CPUs not running the panic
    function are off, so it's very dangerous to wait on a regular
    spinlock, there's a risk of deadlock.

    Refactor the panic notifier of parisc/power driver to make use
    of spin_trylock - for that, we've added a second version of the
    soft-power function. Also, some comments were reorganized and
    trailing white spaces, useless header inclusion and blank lines
    were removed.

    Cc: "James E.J. Bottomley" <James.Bottomley@HansenPartnership.com>
    Cc: Jeroen Roovers <jer@xs4all.nl>
    Acked-by: Helge Deller <deller@gmx.de> # parisc
    Signed-off-by: Guilherme G. Piccoli <gpiccoli@igalia.com>
    Signed-off-by: Helge Deller <deller@gmx.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a84e588d2527d2a43dd9c6c175f2def3a5300323
Author: Jarkko Nikula <jarkko.nikula@linux.intel.com>
Date:   Thu Mar 30 16:26:18 2023 +0300

    mfd: intel-lpss: Add Intel Meteor Lake PCH-S LPSS PCI IDs

    [ Upstream commit 72d4a1683741ee578da0e265886e6a7f3d42266c ]

    Add Intel Meteor Lake PCH-S also called as Meteor Point-S LPSS PCI IDs.

    Signed-off-by: Jarkko Nikula <jarkko.nikula@linux.intel.com>
    Acked-by: Andy Shevchenko <andriy.shevchenko@linux.intel.com>
    Signed-off-by: Lee Jones <lee@kernel.org>
    Link: https://lore.kernel.org/r/20230330132618.4108665-1-jarkko.nikula@linux.intel.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit fa045c911f0bfc0305c71618ab5630153faf86a4
Author: Qiang Ning <qning0106@126.com>
Date:   Thu Mar 30 10:43:53 2023 +0800

    mfd: dln2: Fix memory leak in dln2_probe()

    [ Upstream commit 96da8f148396329ba769246cb8ceaa35f1ddfc48 ]

    When dln2_setup_rx_urbs() in dln2_probe() fails, error out_free forgets
    to call usb_put_dev() to decrease the refcount of dln2->usb_dev.

    Fix this by adding usb_put_dev() in the error handling code of
    dln2_probe().
```

```
    Signed-off-by: Qiang Ning <qning0106@126.com>
    Signed-off-by: Lee Jones <lee@kernel.org>
    Link: https://lore.kernel.org/r/20230330024353.4503-1-qning0106@126.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7f87ef8411af75b85f9a7d1bfc42fece5bdc165d
Author: Hans de Goede <hdegoede@redhat.com>
Date:   Wed Mar 1 10:54:02 2023 +0100

    mfd: intel_soc_pmic_chtwc: Add Lenovo Yoga Book X90F to intel_cht_wc_models

    [ Upstream commit ded99b89d25fd73a1d7bd910378e0339fd9d4c4a ]

    The Android Lenovo Yoga Book X90F / X90L uses the same charger / fuelgauge
    setup as the already supported Windows Lenovo Yoga Book X91F/L, add
    a DMI match for this to intel_cht_wc_models with driver_data
    set to INTEL_CHT_WC_LENOVO_YOGABOOK1.

    When the quirk for the X91F/L was initially added it was written to
    also apply to the X90F/L but this does not work because the Android
    version of the Yoga Book uses completely different DMI strings.
    Also adjust the X91F/L quirk to reflect that it only applies to
    the X91F/L models.

    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Reviewed-by: Andy Shevchenko <andy.shevchenko@gmail.com>
    Signed-off-by: Lee Jones <lee@kernel.org>
    Link: https://lore.kernel.org/r/20230301095402.28582-1-hdegoede@redhat.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 203aa4374c433159f163acde2d0bd4118f23bbaf
Author: Richard Fitzgerald <rf@opensource.cirrus.com>
Date:   Thu Apr 6 14:46:40 2023 +0100

    soundwire: bus: Fix unbalanced pm_runtime_put() causing usage count underflow

    [ Upstream commit e9537962519e88969f5f69cd0571eb4f6984403c ]

    This reverts commit
    443a98e649b4 ("soundwire: bus: use pm_runtime_resume_and_get()")

    Change calls to pm_runtime_resume_and_get() back to pm_runtime_get_sync().
    This fixes a usage count underrun caused by doing a pm_runtime_put() even
    though pm_runtime_resume_and_get() returned an error.

    The three affected functions ignore -EACCES error from trying to get
    pm_runtime, and carry on, including a put at the end of the function.
    But pm_runtime_resume_and_get() does not increment the usage count if it
    returns an error. So in the -EACCES case you must not call
    pm_runtime_put().

    The documentation for pm_runtime_get_sync() says:
     "Consider using pm_runtime_resume_and_get() ...  as this is likely to
     result in cleaner code."

    In this case I don't think it results in cleaner code because the
    pm_runtime_put() at the end of the function would have to be conditional on
    the return value from pm_runtime_resume_and_get() at the top of the
    function.

    pm_runtime_get_sync() doesn't have this problem because it always
    increments the count, so always needs a put. The code can just flow through
    and do the pm_runtime_put() unconditionally.
```

```
    Signed-off-by: Richard Fitzgerald <rf@opensource.cirrus.com>
    Reviewed-by: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com>
    Link: https://lore.kernel.org/r/20230406134640.8582-1-rf@opensource.cirrus.com
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dea96ca9dc050682c7335846095cd66f687d2b66
Author: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
Date:   Wed Feb 22 15:44:12 2023 +0100

    soundwire: qcom: gracefully handle too many ports in DT

    [ Upstream commit 2367e0ecb498764e95cfda691ff0828f7d25f9a4 ]

    There are two issues related to the number of ports coming from
    Devicetree when exceeding in total QCOM_SDW_MAX_PORTS.  Both lead to
    incorrect memory accesses:
    1. With DTS having too big value of input or output ports, the driver,
       when copying port parameters from local/stack arrays into 'pconfig'
       array in 'struct qcom_swrm_ctrl', will iterate over their sizes.

    2. If DTS also has too many parameters for these ports (e.g.
       qcom,ports-sinterval-low), the driver will overflow buffers on the
       stack when reading these properties from DTS.

    Add a sanity check so incorrect DTS will not cause kernel memory
    corruption.

    Signed-off-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Reviewed-by: Srinivas Kandagatla <srinivas.kandagatla@linaro.org>
    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Link: https://lore.kernel.org/r/20230222144412.237832-2-krzysztof.kozlowski@linaro.org
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b91020efb1479bfd3f20892cccb922af8ac225dd
Author: Alain Volmat <avolmat@me.com>
Date:   Fri Feb 10 23:43:08 2023 +0100

    phy: st: miphy28lp: use _poll_timeout functions for waits

    [ Upstream commit e3be4dd2c8d8aabfd2c3127d0e2e5754d3ae82d6 ]

    This commit introduces _poll_timeout functions usage instead of
    wait loops waiting for a status bit.

    Signed-off-by: Alain Volmat <avolmat@me.com>
    Reviewed-by: Patrice Chotard <patrice.chotard@foss.st.com>
    Link: https://lore.kernel.org/r/20230210224309.98452-1-avolmat@me.com
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 39b1b92817ea8e66849233bfd381f3b6f35c7b79
Author: Eugene Huang <eugene.huang99@gmail.com>
Date:   Tue Mar 14 17:06:18 2023 +0800

    soundwire: dmi-quirks: add remapping for Intel 'Rooks County' NUC M15

    [ Upstream commit 01b33e284ca28cc977bdcfb23be2c719f2139175 ]

    Same DSDT problem as the HP Omen 16-k0005TX, except rt1316 amp is on
    link2.

    Link: https://github.com/thesofproject/linux/issues/4088
    Signed-off-by: Eugene Huang <eugene.huang99@gmail.com>
```

```
    Reviewed-by: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com>
    Reviewed-by: Péter Ujfalusi <peter.ujfalusi@linux.intel.com>
    Signed-off-by: Bard Liao <yung-chuan.liao@linux.intel.com>
    Link: https://lore.kernel.org/r/20230314090618.498716-1-yung-chuan.liao@linux.intel.com
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 07c9ed563a69f22470a9e909ee3a5f5b9c2cdebf
Author: Andy Shevchenko <andriy.shevchenko@linux.intel.com>
Date:   Wed Feb 15 15:42:38 2023 +0200

    pinctrl: at91: use devm_kasprintf() to avoid potential leaks (part 2)

    [ Upstream commit f494c1913cbb34b9e2078b7b045c87c1ca6df791 ]

    Use devm_kasprintf() instead of kasprintf() to avoid any potential
    leaks. At the moment drivers have no remove functionality hence
    there is no need for fixes tag.

    While at it, switch to use devm_kasprintf_strarray().

    Signed-off-by: Andy Shevchenko <andriy.shevchenko@linux.intel.com>
    Reviewed-by: Claudiu Beznea <claudiu.beznea@microchip.com>
    Tested-by: Claudiu Beznea <claudiu.beznea@microchip.com>
    Link: https://lore.kernel.org/r/20230215134242.37618-2-andriy.shevchenko@linux.intel.com
    Signed-off-by: Linus Walleij <linus.walleij@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9fd5f48429b004591069b95df022dbe4d6480387
Author: Steve French <stfrench@microsoft.com>
Date:   Wed Apr 26 22:01:31 2023 -0500

    cifs: missing lock when updating session status

    [ Upstream commit 943fb67b090212f1d3789eb7796b1c9045c62fd6 ]

    Coverity noted a place where we were not grabbing
    the ses_lock when setting (and checking) ses_status.

    Addresses-Coverity: 1536833 ("Data race condition (MISSING_LOCK)")
    Reviewed-by: Paulo Alcantara (SUSE) <pc@manguebit.com>
    Reviewed-by: Bharath SM <bharathsm@microsoft.com>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 98016c2b9b4888a51c0a2b6aa3b2382b69c1d06e
Author: Sebastian Reichel <sebastian.reichel@collabora.com>
Date:   Mon Apr 3 21:32:49 2023 +0200

    clk: rockchip: rk3588: make gate linked clocks critical

    [ Upstream commit 64042c28c3bb6729df8e2fda89bc7ebbe3790907 ]

    RK3588 has a couple of hardware blocks called Native Interface Unit
    (NIU) that gate the clocks to devices behind them. Effectively this
    means that some clocks require two parent clocks being enabled.
    Downstream implemented this by using a separate clock driver
    ("clk-link") for them, which enables the second clock using PM
    framework.

    In the upstream kernel we are currently missing support for the second
    parent. The information about it is in the GATE_LINK() macro as
    linkname, but that is not used. Thus the second parent clock is not
    properly enabled. So far this did not really matter, since these clocks
    are mostly required for the more advanced IP blocks, that are not yet
```

```
    supported upstream. As this is about to change we need a fix. There
    are three options available:

    1. Properly implement support for having two parent clocks in the
       clock framework.
    2. Mark the affected clocks CLK_IGNORE_UNUSED, so that they are not
       disabled. This wastes some power, but keeps the hack contained
       within the clock driver. Going from this to the first solution
       is easy once that has been implemented.
    3. Enabling the extra clock in the consumer driver. This leaks some
       implementation details into DT.

    This patch implements the second option as an intermediate solution
    until the first one is available. I used an alias for CLK_IS_CRITICAL,
    so that it's easy to see which clocks are not really critical once
    the clock framework supports a better way to implement this.

    Tested-by: Vincent Legoll <vincent.legoll@gmail.com>
    Signed-off-by: Sebastian Reichel <sebastian.reichel@collabora.com>
    Link: https://lore.kernel.org/r/20230403193250.108693-2-sebastian.reichel@collabora.com
    Signed-off-by: Heiko Stuebner <heiko@sntech.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0e8d2abe1e18f0513bb699c8b2b7700b369cffca
Author: Avihai Horon <avihaih@nvidia.com>
Date:   Mon Apr 10 16:07:50 2023 +0300

    RDMA/mlx5: Remove pcie_relaxed_ordering_enabled() check for RO write

    [ Upstream commit ed4b0661cce119870edb1994fd06c9cbc1dc05c3 ]

    pcie_relaxed_ordering_enabled() check was added to avoid a syndrome when
    creating a MKey with relaxed ordering (RO) enabled when the driver's
    relaxed_ordering_{read,write} HCA capabilities are out of sync with FW.

    While this can happen with relaxed_ordering_read, it can't happen with
    relaxed_ordering_write as it's set if the device supports RO write,
    regardless of RO in PCI config space, and thus can't change during
    runtime.

    Therefore, drop the pcie_relaxed_ordering_enabled() check for
    relaxed_ordering_write while keeping it for relaxed_ordering_read.
    Doing so will also allow the usage of RO write in VFs and VMs (where RO
    in PCI config space is not reported/emulated properly).

    Signed-off-by: Avihai Horon <avihaih@nvidia.com>
    Reviewed-by: Shay Drory <shayd@nvidia.com>
    Link:
https://lore.kernel.org/r/7e8f55e31572c1702d69cae015a395d3a824a38a.1681131553.git.leon@kerne
l.org
    Reviewed-by: Jacob Keller <jacob.e.keller@intel.com>
    Signed-off-by: Leon Romanovsky <leon@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit db96d284030f3714e6c46722c64a38ddc0e0a905
Author: Vicki Pfau <vi@endrift.com>
Date:   Thu Apr 13 23:57:42 2023 -0700

    Input: xpad - add constants for GIP interface numbers

    [ Upstream commit f9b2e603c6216824e34dc9a67205d98ccc9a41ca ]

    Wired GIP devices present multiple interfaces with the same USB identification
    other than the interface number. This adds constants for differentiating two of
    them and uses them where appropriate
```

```
    Signed-off-by: Vicki Pfau <vi@endrift.com>
    Link: https://lore.kernel.org/r/20230411031650.960322-2-vi@endrift.com
    Signed-off-by: Dmitry Torokhov <dmitry.torokhov@gmail.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b5d1ed4334b3ff22b9194ce88f5d3e5bfd806201
Author: Hans de Goede <hdegoede@redhat.com>
Date:   Sat Apr 1 17:06:51 2023 +0200

    power: supply: axp288_charger: Use alt usb-id extcon on some x86 android tablets

    [ Upstream commit ce38f3fc0f87a358a9560a3815265a94f1b38c37 ]

    x86 ACPI boards which ship with only Android as their factory image may
    have pretty broken ACPI tables. This includes broken _AEI ACPI GPIO event
    handlers, which are normally used to listen to the micro-USB ID pin and:

    1. Switch the USB-mux to the host / device USB controllers
    2. Disable Vbus path before enabling the 5V boost (AXP reg 0x30 bit 7)
    3. Turn 5V Vboost on / off

    On non broken systems where this is not done through an ACPI GPIO event
    handler, there is an ACPI INT3496 device describing the involved GPIOs
    which are handled by the extcon-intel-int3496 driver; and axp288-charger.ko
    listens to this extcon-device and disables the Vbus path when necessary.

    On x86 Android boards, with broken ACPI GPIO event handlers, these are
    disabled by acpi_quirk_skip_gpio_event_handlers() and an intel-int3496
    extcon device is manually instantiated by x86-android-tablets.ko .

    Add support to the axp288-charger code for this setup, so that it
    properly disables the Vbus path when necessary. Note this uses
    acpi_quirk_skip_gpio_event_handlers() to identify these systems,
    to avoid the need to add a separate DMI match table for this.

    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Sebastian Reichel <sebastian.reichel@collabora.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d0a917fd5e3b3ed9d9306b4260ba684b982da9f3
Author: Chunyan Zhang <chunyan.zhang@unisoc.com>
Date:   Fri Mar 31 11:31:23 2023 +0800

    iommu/sprd: Release dma buffer to avoid memory leak

    [ Upstream commit 9afea57384d4ae7b2034593eac7fa76c7122762a ]

    When attaching to a domain, the driver would alloc a DMA buffer which
    is used to store address mapping table, and it need to be released
    when the IOMMU domain is freed.

    Signed-off-by: Chunyan Zhang <chunyan.zhang@unisoc.com>
    Link: https://lore.kernel.org/r/20230331033124.864691-2-zhang.lyra@gmail.com
    Signed-off-by: Joerg Roedel <jroedel@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d1afdab5682c070d8f96c1c4c2ceb3c8ec23c7fc
Author: Tomas Krcka <krckatom@amazon.de>
Date:   Wed Mar 29 12:34:19 2023 +0000

    iommu/arm-smmu-v3: Acknowledge pri/event queue overflow if any

    [ Upstream commit 67ea0b7ce41844eae7c10bb04dfe66a23318c224 ]
```

```
    When an overflow occurs in the PRI queue, the SMMU toggles the overflow
    flag in the PROD register. To exit the overflow condition, the PRI thread
    is supposed to acknowledge it by toggling this flag in the CONS register.
    Unacknowledged overflow causes the queue to stop adding anything new.

    Currently, the priq thread always writes the CONS register back to the
    SMMU after clearing the queue.

    The writeback is not necessary if the OVFLG in the PROD register has not
    been changed, no overflow has occured.

    This commit checks the difference of the overflow flag between CONS and
    PROD register. If it's different, toggles the OVACKFLG flag in the CONS
    register and write it to the SMMU.

    The situation is similar for the event queue.
    The acknowledge register is also toggled after clearing the event
    queue but never propagated to the hardware. This would only be done the
    next time when executing evtq thread.

    Unacknowledged event queue overflow doesn't affect the event
    queue, because the SMMU still adds elements to that queue when the
    overflow condition is active.
    But it feel nicer to keep SMMU in sync when possible, so use the same
    way here as well.

    Signed-off-by: Tomas Krcka <krckatom@amazon.de>
    Link: https://lore.kernel.org/r/20230329123420.34641-1-tomas.krcka@gmail.com
    Signed-off-by: Will Deacon <will@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1a370ae100dcb4641743062b680cc0e9750a2991
Author: Arnd Bergmann <arnd@arndb.de>
Date:   Mon Feb 27 09:59:10 2023 +0100

    clk: tegra20: fix gcc-7 constant overflow warning

    [ Upstream commit b4a2adbf3586efa12fe78b9dec047423e01f3010 ]

    Older gcc versions get confused by comparing a u32 value to a negative
    constant in a switch()/case block:

    drivers/clk/tegra/clk-tegra20.c: In function 'tegra20_clk_measure_input_freq':
    drivers/clk/tegra/clk-tegra20.c:581:2: error: case label does not reduce to an integer
constant
      case OSC_CTRL_OSC_FREQ_12MHZ:
      ^~~~
    drivers/clk/tegra/clk-tegra20.c:593:2: error: case label does not reduce to an integer
constant
      case OSC_CTRL_OSC_FREQ_26MHZ:

    Make the constants unsigned instead.

    Signed-off-by: Arnd Bergmann <arnd@arndb.de>
    Link: https://lore.kernel.org/r/20230227085914.2560984-1-arnd@kernel.org
    Signed-off-by: Stephen Boyd <sboyd@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e7675f85a92233136c630000a0b7cf97826705da
Author: Jia-Ju Bai <baijiaju1990@gmail.com>
Date:   Wed Jan 11 16:59:43 2023 +0800

    fs/ntfs3: Fix a possible null-pointer dereference in ni_clear()

    [ Upstream commit ec275bf9693d19cc0fdce8436f4c425ced86f6e7 ]
```

```
    In a previous commit c1006bd13146, ni->mi.mrec in ni_write_inode()
    could be NULL, and thus a NULL check is added for this variable.

    However, in the same call stack, ni->mi.mrec can be also dereferenced
    in ni_clear():

    ntfs_evict_inode(inode)
      ni_write_inode(inode, ...)
        ni = ntfs_i(inode);
        is_rec_inuse(ni->mi.mrec) -> Add a NULL check by previous commit
      ni_clear(ntfs_i(inode))
        is_rec_inuse(ni->mi.mrec) -> No check

    Thus, a possible null-pointer dereference may exist in ni_clear().
    To fix it, a NULL check is added in this function.

    Signed-off-by: Jia-Ju Bai <baijiaju1990@gmail.com>
    Reported-by: TOTE Robot <oslab@tsinghua.edu.cn>
    Signed-off-by: Konstantin Komarov <almaz.alexandrovich@paragon-software.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit eb5b59931d20f3b02076fae49e85282310b12012
Author: Edward Lo <edward.lo@ambergroup.io>
Date:   Tue Oct 4 23:15:06 2022 +0800

    fs/ntfs3: Add length check in indx_get_root

    [ Upstream commit 08e8cf5f2d9ec383a2e339a2711b62a54ff3fba0 ]

    This adds a length check to guarantee the retrieved index root is legit.

    [  162.459513] BUG: KASAN: use-after-free in hdr_find_e.isra.0+0x10c/0x320
    [  162.460176] Read of size 2 at addr ffff8880037bca99 by task mount/243
    [  162.460851]
    [  162.461252] CPU: 0 PID: 243 Comm: mount Not tainted 6.0.0-rc7 #42
    [  162.461744] Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS rel-1.14.0-0-
g155821a1990b-prebuilt.qemu.org 04/01/2014
    [  162.462609] Call Trace:
    [  162.462954]  <TASK>
    [  162.463276]  dump_stack_lvl+0x49/0x63
    [  162.463822]  print_report.cold+0xf5/0x689
    [  162.464608]  ? unwind_get_return_address+0x3a/0x60
    [  162.465766]  ? hdr_find_e.isra.0+0x10c/0x320
    [  162.466975]  kasan_report+0xa7/0x130
    [  162.467506]  ? _raw_spin_lock_irq+0xc0/0xf0
    [  162.467998]  ? hdr_find_e.isra.0+0x10c/0x320
    [  162.468536]  __asan_load2+0x68/0x90
    [  162.468923]  hdr_find_e.isra.0+0x10c/0x320
    [  162.469282]  ? cmp_uints+0xe0/0xe0
    [  162.469557]  ? cmp_sdh+0x90/0x90
    [  162.469864]  ? ni_find_attr+0x214/0x300
    [  162.470217]  ? ni_load_mi+0x80/0x80
    [  162.470479]  ? entry_SYSCALL_64_after_hwframe+0x63/0xcd
    [  162.470931]  ? ntfs_bread_run+0x190/0x190
    [  162.471307]  ? indx_get_root+0xe4/0x190
    [  162.471556]  ? indx_get_root+0x140/0x190
    [  162.471833]  ? indx_init+0x1e0/0x1e0
    [  162.472069]  ? fnd_clear+0x115/0x140
    [  162.472363]  ? _raw_spin_lock_irqsave+0x100/0x100
    [  162.472731]  indx_find+0x184/0x470
    [  162.473461]  ? sysvec_apic_timer_interrupt+0x57/0xc0
    [  162.474429]  ? indx_find_buffer+0x2d0/0x2d0
    [  162.474704]  ? do_syscall_64+0x3b/0x90
    [  162.474962]  dir_search_u+0x196/0x2f0
```

```
     [  162.475381]  ? ntfs_nls_to_utf16+0x450/0x450
     [  162.475661]  ? ntfs_security_init+0x3d6/0x440
     [  162.475906]  ? is_sd_valid+0x180/0x180
     [  162.476191]  ntfs_extend_init+0x13f/0x2c0
     [  162.476496]  ? ntfs_fix_post_read+0x130/0x130
     [  162.476861]  ? iput.part.0+0x286/0x320
     [  162.477325]  ntfs_fill_super+0x11e0/0x1b50
     [  162.477709]  ? put_ntfs+0x1d0/0x1d0
     [  162.477970]  ? vsprintf+0x20/0x20
     [  162.478258]  ? set_blocksize+0x95/0x150
     [  162.478538]  get_tree_bdev+0x232/0x370
     [  162.478789]  ? put_ntfs+0x1d0/0x1d0
     [  162.479038]  ntfs_fs_get_tree+0x15/0x20
     [  162.479374]  vfs_get_tree+0x4c/0x130
     [  162.479729]  path_mount+0x654/0xfe0
     [  162.480124]  ? putname+0x80/0xa0
     [  162.480484]  ? finish_automount+0x2e0/0x2e0
     [  162.480894]  ? putname+0x80/0xa0
     [  162.481467]  ? kmem_cache_free+0x1c4/0x440
     [  162.482280]  ? putname+0x80/0xa0
     [  162.482714]  do_mount+0xd6/0xf0
     [  162.483264]  ? path_mount+0xfe0/0xfe0
     [  162.484782]  ? __kasan_check_write+0x14/0x20
     [  162.485593]  __x64_sys_mount+0xca/0x110
     [  162.486024]  do_syscall_64+0x3b/0x90
     [  162.486543]  entry_SYSCALL_64_after_hwframe+0x63/0xcd
     [  162.487141] RIP: 0033:0x7f9d374e948a
     [  162.488324] Code: 48 8b 0d 11 fa 2a 00 f7 d8 64 89 01 48 83 c8 ff c3 66 2e 0f 1f 84
 00 00 00 00 00 0f 1f 44 00 00 49 89 ca b8 a5 00 00 008
     [  162.489728] RSP: 002b:00007ffe30e73d18 EFLAGS: 00000206 ORIG_RAX: 00000000000000a5
     [  162.490971] RAX: ffffffffffffffda RBX: 0000561cdb43a060 RCX: 00007f9d374e948a
     [  162.491669] RDX: 0000561cdb43a260 RSI: 0000561cdb43a2e0 RDI: 0000561cdb442af0
     [  162.492050] RBP: 0000000000000000 R08: 0000561cdb43a280 R09: 0000000000000020
     [  162.492459] R10: 00000000c0ed0000 R11: 0000000000000206 R12: 0000561cdb442af0
     [  162.493183] R13: 0000561cdb43a260 R14: 0000000000000000 R15: 00000000ffffffff
     [  162.493644]  </TASK>
     [  162.493908]
     [  162.494214] The buggy address belongs to the physical page:
     [  162.494761] page:000000003e38a3d5 refcount:0 mapcount:0 mapping:0000000000000000
 index:0x0 pfn:0x37bc
     [  162.496064] flags: 0xfffffc0000000(node=0|zone=1|lastcpupid=0x1fffff)
     [  162.497278] raw: 000fffffc0000000 ffffea00000df1c8 ffffea00000df008 0000000000000000
     [  162.498928] raw: 0000000000000000 0000000000240000 00000000ffffffff 0000000000000000
     [  162.500542] page dumped because: kasan: bad access detected
     [  162.501057]
     [  162.501242] Memory state around the buggy address:
     [  162.502230]  ffff8880037bc980: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff
     [  162.502977]  ffff8880037bca00: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff
     [  162.503522] >ffff8880037bca80: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff
     [  162.503963]                                ^
     [  162.504370]  ffff8880037bcb00: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff
     [  162.504766]  ffff8880037bcb80: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff

    Signed-off-by: Edward Lo <edward.lo@ambergroup.io>
    Signed-off-by: Konstantin Komarov <almaz.alexandrovich@paragon-software.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit e6f4b1c32d6d6047958d7700d12fed6d91f441e7
 Author: Edward Lo <edward.lo@ambergroup.io>
 Date:   Sat Nov 5 23:39:44 2022 +0800

    fs/ntfs3: Validate MFT flags before replaying logs

    [ Upstream commit 98bea253aa28ad8be2ce565a9ca21beb4a9419e5 ]
```

```
    Log load and replay is part of the metadata handle flow during mount
    operation. The $MFT record will be loaded and used while replaying logs.
    However, a malformed $MFT record, say, has RECORD_FLAG_DIR flag set and
    contains an ATTR_ROOT attribute will misguide kernel to treat it as a
    directory, and try to free the allocated resources when the
    corresponding inode is freed, which will cause an invalid kfree because
    the memory hasn't actually been allocated.

    [  101.368647] BUG: KASAN: invalid-free in kvfree+0x2c/0x40
    [  101.369457]
    [  101.369986] CPU: 0 PID: 198 Comm: mount Not tainted 6.0.0-rc7+ #5
    [  101.370529] Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS rel-1.14.0-0-
g155821a1990b-prebuilt.qemu.org 04/01/2014
    [  101.371362] Call Trace:
    [  101.371795]  <TASK>
    [  101.372157]  dump_stack_lvl+0x49/0x63
    [  101.372658]  print_report.cold+0xf5/0x689
    [  101.373022]  ? ni_write_inode+0x754/0xd90
    [  101.373378]  ? kvfree+0x2c/0x40
    [  101.373698]  kasan_report_invalid_free+0x77/0xf0
    [  101.374058]  ? kvfree+0x2c/0x40
    [  101.374352]  ? kvfree+0x2c/0x40
    [  101.374668]  __kasan_slab_free+0x189/0x1b0
    [  101.374992]  ? kvfree+0x2c/0x40
    [  101.375271]  kfree+0x168/0x3b0
    [  101.375717]  kvfree+0x2c/0x40
    [  101.376002]  indx_clear+0x26/0x60
    [  101.376316]  ni_clear+0xc5/0x290
    [  101.376661]  ntfs_evict_inode+0x45/0x70
    [  101.377001]  evict+0x199/0x280
    [  101.377432]  iput.part.0+0x286/0x320
    [  101.377819]  iput+0x32/0x50
    [  101.378166]  ntfs_loadlog_and_replay+0x143/0x320
    [  101.378656]  ? ntfs_bio_fill_1+0x510/0x510
    [  101.378968]  ? iput.part.0+0x286/0x320
    [  101.379367]  ntfs_fill_super+0xecb/0x1ba0
    [  101.379729]  ? put_ntfs+0x1d0/0x1d0
    [  101.380046]  ? vsprintf+0x20/0x20
    [  101.380542]  ? mutex_unlock+0x81/0xd0
    [  101.380914]  ? set_blocksize+0x95/0x150
    [  101.381597]  get_tree_bdev+0x232/0x370
    [  101.382254]  ? put_ntfs+0x1d0/0x1d0
    [  101.382699]  ntfs_fs_get_tree+0x15/0x20
    [  101.383094]  vfs_get_tree+0x4c/0x130
    [  101.383675]  path_mount+0x654/0xfe0
    [  101.384203]  ? putname+0x80/0xa0
    [  101.384540]  ? finish_automount+0x2e0/0x2e0
    [  101.384943]  ? putname+0x80/0xa0
    [  101.385362]  ? kmem_cache_free+0x1c4/0x440
    [  101.385968]  ? putname+0x80/0xa0
    [  101.386666]  do_mount+0xd6/0xf0
    [  101.387228]  ? path_mount+0xfe0/0xfe0
    [  101.387585]  ? __kasan_check_write+0x14/0x20
    [  101.387979]  __x64_sys_mount+0xca/0x110
    [  101.388436]  do_syscall_64+0x3b/0x90
    [  101.388757]  entry_SYSCALL_64_after_hwframe+0x63/0xcd
    [  101.389289] RIP: 0033:0x7fa0f70e948a
    [  101.390048] Code: 48 8b 0d 11 fa 2a 00 f7 d8 64 89 01 48 83 c8 ff c3 66 2e 0f 1f 84
00 00 00 00 00 0f 1f 44 00 00 49 89 ca b8 a5 00 00 008
    [  101.391297] RSP: 002b:00007ffc24fdecc8 EFLAGS: 00000202 ORIG_RAX: 00000000000000a5
    [  101.391988] RAX: ffffffffffffffda RBX: 000055932c183060 RCX: 00007fa0f70e948a
    [  101.392494] RDX: 000055932c183260 RSI: 000055932c1832e0 RDI: 000055932c18bce0
    [  101.393053] RBP: 0000000000000000 R08: 000055932c183280 R09: 0000000000000020
    [  101.393577] R10: 00000000c0ed0000 R11: 0000000000000202 R12: 000055932c18bce0
    [  101.394044] R13: 000055932c183260 R14: 0000000000000000 R15: 00000000ffffffff
```

```
      [  101.394747]  </TASK>
      [  101.395402]
      [  101.396047] Allocated by task 198:
      [  101.396724]  kasan_save_stack+0x26/0x50
      [  101.397400]  __kasan_slab_alloc+0x6d/0x90
      [  101.397974]  kmem_cache_alloc_lru+0x192/0x5a0
      [  101.398524]  ntfs_alloc_inode+0x23/0x70
      [  101.399137]  alloc_inode+0x3b/0xf0
      [  101.399534]  iget5_locked+0x54/0xa0
      [  101.400026]  ntfs_iget5+0xaf/0x1780
      [  101.400414]  ntfs_loadlog_and_replay+0xe5/0x320
      [  101.400883]  ntfs_fill_super+0xecb/0x1ba0
      [  101.401313]  get_tree_bdev+0x232/0x370
      [  101.401774]  ntfs_fs_get_tree+0x15/0x20
      [  101.402224]  vfs_get_tree+0x4c/0x130
      [  101.402673]  path_mount+0x654/0xfe0
      [  101.403160]  do_mount+0xd6/0xf0
      [  101.403537]  __x64_sys_mount+0xca/0x110
      [  101.404058]  do_syscall_64+0x3b/0x90
      [  101.404333]  entry_SYSCALL_64_after_hwframe+0x63/0xcd
      [  101.404816]
      [  101.405067] The buggy address belongs to the object at ffff888008cc9ea0
      [  101.405067]  which belongs to the cache ntfs_inode_cache of size 992
      [  101.406171] The buggy address is located 232 bytes inside of
      [  101.406171]  992-byte region [ffff888008cc9ea0, ffff888008cca280)
      [  101.406995]
      [  101.408559] The buggy address belongs to the physical page:
      [  101.409320] page:00000000dccf19dd refcount:1 mapcount:0 mapping:0000000000000000
  index:0x0 pfn:0x8cc8
      [  101.410654] head:00000000dccf19dd order:2 compound_mapcount:0 compound_pincount:0
      [  101.411533] flags: 0xfffffc0010200(slab|head|node=0|zone=1|lastcpupid=0x1fffff)
      [  101.412665] raw: 000fffffc0010200 0000000000000000 dead000000000122 ffff888003695140
      [  101.413209] raw: 0000000000000000 00000000800e000e 00000001ffffffff 0000000000000000
      [  101.413799] page dumped because: kasan: bad access detected
      [  101.414213]
      [  101.414427] Memory state around the buggy address:
      [  101.414991]  ffff888008cc9e80: fc fc fc fc 00 00 00 00 00 00 00 00 00 00 00 00
      [  101.415785]  ffff888008cc9f00: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
      [  101.416933] >ffff888008cc9f80: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
      [  101.417857]                    ^
      [  101.418566]  ffff888008cca000: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
      [  101.419704]  ffff888008cca080: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

      Signed-off-by: Edward Lo <edward.lo@ambergroup.io>
      Signed-off-by: Konstantin Komarov <almaz.alexandrovich@paragon-software.com>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit b1135fbaf8ebef93df326761ac70ebcc3c2e3d63
  Author: Abdun Nihaal <abdun.nihaal@gmail.com>
  Date:   Sun Oct 30 12:32:51 2022 +0530

      fs/ntfs3: Fix NULL dereference in ni_write_inode

      [ Upstream commit 8dae4f6341e335a09575be60b4fdf697c732a470 ]

      Syzbot reports a NULL dereference in ni_write_inode.
      When creating a new inode, if allocation fails in mi_init function
      (called in mi_format_new function), mi->mrec is set to NULL.
      In the error path of this inode creation, mi->mrec is later
      dereferenced in ni_write_inode.

      Add a NULL check to prevent NULL dereference.

      Link: https://syzkaller.appspot.com/bug?extid=f45957555ed4a808cc7a
      Reported-and-tested-by: syzbot+f45957555ed4a808cc7a@syzkaller.appspotmail.com
```

```
    Signed-off-by: Abdun Nihaal <abdun.nihaal@gmail.com>
    Signed-off-by: Konstantin Komarov <almaz.alexandrovich@paragon-software.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f28d9e02c2c242e8f9af9e13ba263fcc0211be49
Author: Edward Lo <edward.lo@ambergroup.io>
Date:   Thu Oct 27 23:33:37 2022 +0800

    fs/ntfs3: Enhance the attribute size check

    [ Upstream commit 4f082a7531223a438c757bb20e304f4c941c67a8 ]

    This combines the overflow and boundary check so that all attribute size
    will be properly examined while enumerating them.

    [  169.181521] BUG: KASAN: slab-out-of-bounds in run_unpack+0x2e3/0x570
    [  169.183161] Read of size 1 at addr ffff8880094b6240 by task mount/247
    [  169.184046]
    [  169.184925] CPU: 0 PID: 247 Comm: mount Not tainted 6.0.0-rc7+ #3
    [  169.185908] Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS rel-1.14.0-0-
g155821a1990b-prebuilt.qemu.org 04/01/2014
    [  169.187066] Call Trace:
    [  169.187492]  <TASK>
    [  169.188049]  dump_stack_lvl+0x49/0x63
    [  169.188495]  print_report.cold+0xf5/0x689
    [  169.188964]  ? run_unpack+0x2e3/0x570
    [  169.189331]  kasan_report+0xa7/0x130
    [  169.189714]  ? run_unpack+0x2e3/0x570
    [  169.190079]  __asan_load1+0x51/0x60
    [  169.190634]  run_unpack+0x2e3/0x570
    [  169.191290]  ? run_pack+0x840/0x840
    [  169.191569]  ? run_lookup_entry+0xb3/0x1f0
    [  169.192443]  ? mi_enum_attr+0x20a/0x230
    [  169.192886]  run_unpack_ex+0xad/0x3e0
    [  169.193276]  ? run_unpack+0x570/0x570
    [  169.193557]  ? ni_load_mi+0x80/0x80
    [  169.193889]  ? debug_smp_processor_id+0x17/0x20
    [  169.194236]  ? mi_init+0x4a/0x70
    [  169.194496]  attr_load_runs_vcn+0x166/0x1c0
    [  169.194851]  ? attr_data_write_resident+0x250/0x250
    [  169.195188]  mi_read+0x133/0x2c0
    [  169.195481]  ntfs_iget5+0x277/0x1780
    [  169.196017]  ? call_rcu+0x1c7/0x330
    [  169.196392]  ? ntfs_get_block_bmap+0x70/0x70
    [  169.196708]  ? evict+0x223/0x280
    [  169.197014]  ? __kmalloc+0x33/0x540
    [  169.197305]  ? wnd_init+0x15b/0x1b0
    [  169.197599]  ntfs_fill_super+0x1026/0x1ba0
    [  169.197994]  ? put_ntfs+0x1d0/0x1d0
    [  169.198299]  ? vsprintf+0x20/0x20
    [  169.198583]  ? mutex_unlock+0x81/0xd0
    [  169.198930]  ? set_blocksize+0x95/0x150
    [  169.199269]  get_tree_bdev+0x232/0x370
    [  169.199750]  ? put_ntfs+0x1d0/0x1d0
    [  169.200094]  ntfs_fs_get_tree+0x15/0x20
    [  169.200431]  vfs_get_tree+0x4c/0x130
    [  169.200714]  path_mount+0x654/0xfe0
    [  169.201067]  ? putname+0x80/0xa0
    [  169.201358]  ? finish_automount+0x2e0/0x2e0
    [  169.201965]  ? putname+0x80/0xa0
    [  169.202445]  ? kmem_cache_free+0x1c4/0x440
    [  169.203075]  ? putname+0x80/0xa0
    [  169.203414]  do_mount+0xd6/0xf0
    [  169.203719]  ? path_mount+0xfe0/0xfe0
    [  169.203977]  ? __kasan_check_write+0x14/0x20
```

```
     [  169.204382]  __x64_sys_mount+0xca/0x110
     [  169.204711]  do_syscall_64+0x3b/0x90
     [  169.205059]  entry_SYSCALL_64_after_hwframe+0x63/0xcd
     [  169.205571] RIP: 0033:0x7f67a80e948a
     [  169.206327] Code: 48 8b 0d 11 fa 2a 00 f7 d8 64 89 01 48 83 c8 ff c3 66 2e 0f 1f 84
  00 00 00 00 00 0f 1f 44 00 00 49 89 ca b8 a5 00 00 008
     [  169.208296] RSP: 002b:00007ffddf020f58 EFLAGS: 00000202 ORIG_RAX: 00000000000000a5
     [  169.209253] RAX: ffffffffffffffda RBX: 000055e2547a6060 RCX: 00007f67a80e948a
     [  169.209777] RDX: 000055e2547a6260 RSI: 000055e2547a62e0 RDI: 000055e2547aeaf0
     [  169.210342] RBP: 0000000000000000 R08: 000055e2547a6280 R09: 0000000000000020
     [  169.210843] R10: 00000000c0ed0000 R11: 0000000000000202 R12: 000055e2547aeaf0
     [  169.211307] R13: 000055e2547a6260 R14: 0000000000000000 R15: 00000000ffffffff
     [  169.211913]  </TASK>
     [  169.212304]
     [  169.212680] Allocated by task 0:
     [  169.212963] (stack is not available)
     [  169.213200]
     [  169.213472] The buggy address belongs to the object at ffff8880094b5e00
     [  169.213472]  which belongs to the cache UDP of size 1152
     [  169.214095] The buggy address is located 1088 bytes inside of
     [  169.214095]  1152-byte region [ffff8880094b5e00, ffff8880094b6280)
     [  169.214639]
     [  169.215004] The buggy address belongs to the physical page:
     [  169.215766] page:000000002e324c8c refcount:1 mapcount:0 mapping:0000000000000000
  index:0x0 pfn:0x94b4
     [  169.218412] head:000000002e324c8c order:2 compound_mapcount:0 compound_pincount:0
     [  169.219078] flags: 0xfffffc0010200(slab|head|node=0|zone=1|lastcpupid=0x1fffff)
     [  169.220272] raw: 000fffffc0010200 0000000000000000 dead000000000122 ffff888002409b40
     [  169.221006] raw: 0000000000000000 00000000800c000c 00000001ffffffff 0000000000000000
     [  169.222320] page dumped because: kasan: bad access detected
     [  169.222922]
     [  169.223119] Memory state around the buggy address:
     [  169.224056]  ffff8880094b6100: fc fc fc fc fc fc fc fc fc fc fc fc fc fc fc fc
     [  169.224908]  ffff8880094b6180: fc fc fc fc fc fc fc fc fc fc fc fc fc fc fc fc
     [  169.225677] >ffff8880094b6200: fc fc fc fc fc fc fc fc fc fc fc fc fc fc fc fc
     [  169.226445]                                                 ^
     [  169.227055]  ffff8880094b6280: fc fc fc fc fc fc fc fc fc fc fc fc fc fc fc fc
     [  169.227638]  ffff8880094b6300: fa fb fb fb fb fb fb fb fb fb fb fb fb fb fb fb

     Signed-off-by: Edward Lo <edward.lo@ambergroup.io>
     Signed-off-by: Konstantin Komarov <almaz.alexandrovich@paragon-software.com>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 1c5cffe0d662fb2de7b63176c2582abb69b5f538
  Author: Ye Bin <yebin10@huawei.com>
  Date:   Thu Nov 17 17:19:12 2022 +0800

     fs/ntfs3: Fix NULL pointer dereference in 'ni_write_inode'

     [ Upstream commit db2a3cc6a3481076da6344cc62a80a4e2525f36f ]

     Syzbot found the following issue:
     Unable to handle kernel NULL pointer dereference at virtual address 0000000000000016
     Mem abort info:
       ESR = 0x0000000096000006
       EC = 0x25: DABT (current EL), IL = 32 bits
       SET = 0, FnV = 0
       EA = 0, S1PTW = 0
       FSC = 0x06: level 2 translation fault
     Data abort info:
       ISV = 0, ISS = 0x00000006
       CM = 0, WnR = 0
     user pgtable: 4k pages, 48-bit VAs, pgdp=000000010af56000
     [0000000000000016] pgd=08000001090da003, p4d=08000001090da003, pud=08000001090ce003,
  pmd=0000000000000000
```

```
    Internal error: Oops: 0000000096000006 [#1] PREEMPT SMP
    Modules linked in:
    CPU: 1 PID: 3036 Comm: syz-executor206 Not tainted 6.0.0-rc6-syzkaller-17739-
g16c9f284e746 #0
    Hardware name: Google Google Compute Engine/Google Compute Engine, BIOS Google
08/26/2022
    pstate: 80400005 (Nzcv daif +PAN -UAO -TCO -DIT -SSBS BTYPE=--)
    pc : is_rec_inuse fs/ntfs3/ntfs.h:313 [inline]
    pc : ni_write_inode+0xac/0x798 fs/ntfs3/frecord.c:3232
    lr : ni_write_inode+0xa0/0x798 fs/ntfs3/frecord.c:3226
    sp : ffff8000126c3800
    x29: ffff8000126c3860 x28: 0000000000000000 x27: ffff0000c8b02000
    x26: ffff0000c7502320 x25: ffff0000c7502288 x24: 0000000000000000
    x23: ffff80000cbec91c x22: ffff0000c8b03000 x21: ffff0000c8b02000
    x20: 0000000000000001 x19: ffff0000c75024d8 x18: 00000000000000c0
    x17: ffff80000dd1b198 x16: ffff80000db59158 x15: ffff0000c4b6b500
    x14: 00000000000000b8 x13: 0000000000000000 x12: ffff0000c4b6b500
    x11: ff80800008be1b60 x10: 0000000000000000 x9 : ffff0000c4b6b500
    x8 : 0000000000000000 x7 : ffff800008be1b50 x6 : 0000000000000000
    x5 : 0000000000000000 x4 : 0000000000000001 x3 : 0000000000000000
    x2 : 0000000000000008 x1 : 0000000000000001 x0 : 0000000000000000
    Call trace:
     is_rec_inuse fs/ntfs3/ntfs.h:313 [inline]
     ni_write_inode+0xac/0x798 fs/ntfs3/frecord.c:3232
     ntfs_evict_inode+0x54/0x84 fs/ntfs3/inode.c:1744
     evict+0xec/0x334 fs/inode.c:665
     iput_final fs/inode.c:1748 [inline]
     iput+0x2c4/0x324 fs/inode.c:1774
     ntfs_new_inode+0x7c/0xe0 fs/ntfs3/fsntfs.c:1660
     ntfs_create_inode+0x20c/0xe78 fs/ntfs3/inode.c:1278
     ntfs_create+0x54/0x74 fs/ntfs3/namei.c:100
     lookup_open fs/namei.c:3413 [inline]
     open_last_lookups fs/namei.c:3481 [inline]
     path_openat+0x804/0x11c4 fs/namei.c:3688
     do_filp_open+0xdc/0x1b8 fs/namei.c:3718
     do_sys_openat2+0xb8/0x22c fs/open.c:1311
     do_sys_open fs/open.c:1327 [inline]
     __do_sys_openat fs/open.c:1343 [inline]
     __se_sys_openat fs/open.c:1338 [inline]
     __arm64_sys_openat+0xb0/0xe0 fs/open.c:1338
     __invoke_syscall arch/arm64/kernel/syscall.c:38 [inline]
     invoke_syscall arch/arm64/kernel/syscall.c:52 [inline]
     el0_svc_common+0x138/0x220 arch/arm64/kernel/syscall.c:142
     do_el0_svc+0x48/0x164 arch/arm64/kernel/syscall.c:206
     el0_svc+0x58/0x150 arch/arm64/kernel/entry-common.c:636
     el0t_64_sync_handler+0x84/0xf0 arch/arm64/kernel/entry-common.c:654
     el0t_64_sync+0x18c/0x190
    Code: 97dafee4 340001b4 f9401328 2a1f03e0 (79402d14)
    ---[ end trace 0000000000000000 ]---

    Above issue may happens as follows:
    ntfs_new_inode
      mi_init
        mi->mrec = kmalloc(sbi->record_size, GFP_NOFS); -->failed to allocate memory
          if (!mi->mrec)
            return -ENOMEM;
    iput
      iput_final
        evict
          ntfs_evict_inode
            ni_write_inode
              is_rec_inuse(ni->mi.mrec)-> As 'ni->mi.mrec' is NULL trigger NULL-ptr-deref

    To solve above issue if new inode failed make inode bad before call 'iput()' in
    'ntfs_new_inode()'.
```

```
    Reported-by: syzbot+f45957555ed4a808cc7a@syzkaller.appspotmail.com
    Signed-off-by: Ye Bin <yebin10@huawei.com>
    Signed-off-by: Konstantin Komarov <almaz.alexandrovich@paragon-software.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f13905a7ac609a91bd91a81214e0db2dc6eed9b1
Author: Manivannan Sadhasivam <mani@kernel.org>
Date:   Mon Mar 27 13:30:29 2023 +0530

    iommu/arm-smmu-qcom: Limit the SMR groups to 128

    [ Upstream commit 12261134732689b7e30c59db9978f81230965181 ]

    Some platforms support more than 128 stream matching groups than what is
    defined by the ARM SMMU architecture specification. But due to some unknown
    reasons, those additional groups don't exhibit the same behavior as the
    architecture supported ones.

    For instance, the additional groups will not detect the quirky behavior of
    some firmware versions intercepting writes to S2CR register, thus skipping
    the quirk implemented in the driver and causing boot crash.

    So let's limit the groups to 128 for now until the issue with those groups
    are fixed and issue a notice to users in that case.

    Reviewed-by: Johan Hovold <johan+linaro@kernel.org>
    Tested-by: Johan Hovold <johan+linaro@kernel.org>
    Signed-off-by: Manivannan Sadhasivam <manivannan.sadhasivam@linaro.org>
    Link: https://lore.kernel.org/r/20230327080029.11584-1-manivannan.sadhasivam@linaro.org
    [will: Reworded the comment slightly]
    Signed-off-by: Will Deacon <will@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4b894457e7435eab87b21be87f7b8a43cd79d721
Author: Ivan Orlov <ivan.orlov0322@gmail.com>
Date:   Wed Mar 22 18:45:28 2023 +0400

    KVM: selftests: Add 'malloc' failure check in vcpu_save_state

    [ Upstream commit 735b0e0f2d001b7ed9486db84453fb860e764a4d ]

    There is a 'malloc' call in vcpu_save_state function, which can
    be unsuccessful. This patch will add the malloc failure checking
    to avoid possible null dereference and give more information
    about test fail reasons.

    Signed-off-by: Ivan Orlov <ivan.orlov0322@gmail.com>
    Link: https://lore.kernel.org/r/20230322144528.704077-1-ivan.orlov0322@gmail.com
    Signed-off-by: Sean Christopherson <seanjc@google.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ce9d36c0e86cb1c314293ba0eeca5579f6e93688
Author: Gustavo A. R. Silva <gustavoars@kernel.org>
Date:   Tue Mar 21 17:47:03 2023 -0600

    RDMA/core: Fix multiple -Warray-bounds warnings

    [ Upstream commit aa4d540b4150052ae3b36d286b9c833a961ce291 ]

    GCC-13 (and Clang)[1] does not like to access a partially allocated
    object, since it cannot reason about it for bounds checking.

    In this case 140 bytes are allocated for an object of type struct
    ib_umad_packet:
```

```
            packet = kzalloc(sizeof(*packet) + IB_MGMT_RMPP_HDR, GFP_KERNEL);

    However, notice that sizeof(*packet) is only 104 bytes:

    struct ib_umad_packet {
            struct ib_mad_send_buf *   msg;                 /*     0     8 */
            struct ib_mad_recv_wc *    recv_wc;             /*     8     8 */
            struct list_head           list;                /*    16    16 */
            int                        length;              /*    32     4 */

            /* XXX 4 bytes hole, try to pack */

            struct ib_user_mad         mad __attribute__((__aligned__(8))); /*    40    64
 */

            /* size: 104, cachelines: 2, members: 5 */
            /* sum members: 100, holes: 1, sum holes: 4 */
            /* forced alignments: 1, forced holes: 1, sum forced holes: 4 */
            /* last cacheline: 40 bytes */
    } __attribute__((__aligned__(8)));

    and 36 bytes extra bytes are allocated for a flexible-array member in
    struct ib_user_mad:

    include/rdma/ib_mad.h:
    120 enum {
    ...
    123         IB_MGMT_RMPP_HDR = 36,
    ... }

    struct ib_user_mad {
            struct ib_user_mad_hdr     hdr;                 /*     0    64 */
            /* --- cacheline 1 boundary (64 bytes) --- */
            __u64                      data[] __attribute__((__aligned__(8))); /*    64
 0 */

            /* size: 64, cachelines: 1, members: 2 */
            /* forced alignments: 1 */
    } __attribute__((__aligned__(8)));

    So we have sizeof(*packet) + IB_MGMT_RMPP_HDR == 140 bytes

    Then the address of the flex-array member (for which only 36 bytes were
    allocated) is casted and copied into a pointer to struct ib_rmpp_mad,
    which, in turn, is of size 256 bytes:

            rmpp_mad = (struct ib_rmpp_mad *) packet->mad.data;

    struct ib_rmpp_mad {
            struct ib_mad_hdr          mad_hdr;             /*     0    24 */
            struct ib_rmpp_hdr         rmpp_hdr;            /*    24    12 */
            u8                         data[220];           /*    36   220 */

            /* size: 256, cachelines: 4, members: 3 */
    };

    The thing is that those 36 bytes allocated for flex-array member data
    in struct ib_user_mad onlly account for the size of both struct ib_mad_hdr
    and struct ib_rmpp_hdr, but nothing is left for array u8 data[220].
    So, the compiler is legitimately complaining about accessing an object
    for which not enough memory was allocated.

    Apparently, the only members of struct ib_rmpp_mad that are relevant
    (that are actually being used) in function ib_umad_write() are mad_hdr
```

```
    and rmpp_hdr. So, instead of casting packet->mad.data to
    (struct ib_rmpp_mad *) create a new structure

    struct ib_rmpp_mad_hdr {
            struct ib_mad_hdr       mad_hdr;
            struct ib_rmpp_hdr      rmpp_hdr;
    } __packed;

    and cast packet->mad.data to (struct ib_rmpp_mad_hdr *).

    Notice that

            IB_MGMT_RMPP_HDR == sizeof(struct ib_rmpp_mad_hdr) == 36 bytes

    Refactor the rest of the code, accordingly.

    Fix the following warnings seen under GCC-13 and -Warray-bounds:
    drivers/infiniband/core/user_mad.c:564:50: warning: array subscript ‘struct
ib_rmpp_mad[0]’ is partly outside array bounds of ‘unsigned char[140]’ [-Warray-bounds=]
    drivers/infiniband/core/user_mad.c:566:42: warning: array subscript ‘struct
ib_rmpp_mad[0]’ is partly outside array bounds of ‘unsigned char[140]’ [-Warray-bounds=]
    drivers/infiniband/core/user_mad.c:618:25: warning: array subscript ‘struct
ib_rmpp_mad[0]’ is partly outside array bounds of ‘unsigned char[140]’ [-Warray-bounds=]
    drivers/infiniband/core/user_mad.c:622:44: warning: array subscript ‘struct
ib_rmpp_mad[0]’ is partly outside array bounds of ‘unsigned char[140]’ [-Warray-bounds=]

    Link: https://github.com/KSPP/linux/issues/273
    Link: https://godbolt.org/z/oYWaGM4Yb [1]
    Signed-off-by: Gustavo A. R. Silva <gustavoars@kernel.org>
    Link: https://lore.kernel.org/r/ZBpB91qQcB10m3Fw@work
    Signed-off-by: Leon Romanovsky <leon@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 25c9b185f121812cbc215fdaa1192c6b9025b428
Author: Hao Zeng <zenghao@kylinos.cn>
Date:   Wed Apr 26 09:05:27 2023 +0800

    recordmcount: Fix memory leaks in the uwrite function

    [ Upstream commit fa359d068574d29e7d2f0fdd0ebe4c6a12b5cfb9 ]

    Common realloc mistake: 'file_append' nulled but not freed upon failure

    Link: https://lkml.kernel.org/r/20230426010527.703093-1-zenghao@kylinos.cn

    Signed-off-by: Hao Zeng <zenghao@kylinos.cn>
    Suggested-by: Steven Rostedt <rostedt@goodmis.org>
    Signed-off-by: Steven Rostedt (Google) <rostedt@goodmis.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0ceeae671edea14c340a2d0d76238b912ac0efac
Author: Alexandre Ghiti <alexghiti@rivosinc.com>
Date:   Fri Feb 3 08:52:30 2023 +0100

    riscv: Fix EFI stub usage of KASAN instrumented strcmp function

    [ Upstream commit 617955ca6e275c4dd0dcf5316fca7fc04a8f2fe6 ]

    The EFI stub must not use any KASAN instrumented code as the kernel
    proper did not initialize the thread pointer and the mapping for the
    KASAN shadow region.

    Avoid using the generic strcmp function, instead use the one in
    drivers/firmware/efi/libstub/string.c.
```

```
    Signed-off-by: Alexandre Ghiti <alexghiti@rivosinc.com>
    Acked-by: Ard Biesheuvel <ardb@kernel.org>
    Reviewed-by: Atish Patra <atishp@rivosinc.com>
    Link: https://lore.kernel.org/r/20230203075232.274282-5-alexghiti@rivosinc.com
    Signed-off-by: Palmer Dabbelt <palmer@rivosinc.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 040a01f51ce240de9528685e8bf1056131dbe29a
Author: Josh Poimboeuf <jpoimboe@kernel.org>
Date:   Wed Apr 12 10:24:08 2023 -0700

    lkdtm/stackleak: Fix noinstr violation

    [ Upstream commit f571da059f86fd9d432aea32c9c7e5aaa53245d8 ]

    Fixes the following warning:

      vmlinux.o: warning: objtool: check_stackleak_irqoff+0x2b6: call to _printk() leaves
.noinstr.text section

    Signed-off-by: Josh Poimboeuf <jpoimboe@kernel.org>
    Signed-off-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Link:
https://lore.kernel.org/r/ee5209f53aa0a62aea58be18f2b78b17606779a6.1681320026.git.jpoimboe@k
ernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d32c0686c04d9fb2b4970834e220e4898bf35ba0
Author: Josh Poimboeuf <jpoimboe@kernel.org>
Date:   Wed Apr 12 10:24:07 2023 -0700

    sched: Fix KCSAN noinstr violation

    [ Upstream commit e0b081d17a9f4e5c0cbb0e5fbeb1abe3de0f7e4e ]

    With KCSAN enabled, end_of_stack() can get out-of-lined.  Force it
    inline.

    Fixes the following warnings:

      vmlinux.o: warning: objtool: check_stackleak_irqoff+0x2b: call to end_of_stack()
leaves .noinstr.text section

    Signed-off-by: Josh Poimboeuf <jpoimboe@kernel.org>
    Signed-off-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Link:
https://lore.kernel.org/r/cc1b4d73d3a428a00d206242a68fdf99a934ca7b.1681320026.git.jpoimboe@k
ernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 68f28118efb9cc7c929e727e4e9ec134bd475df1
Author: Rob Herring <robh@kernel.org>
Date:   Fri Mar 10 08:46:56 2023 -0600

    powerpc: Use of_property_present() for testing DT property presence

    [ Upstream commit 857d423c74228cfa064f79ff3a16b163fdb8d542 ]

    It is preferred to use typed property access functions (i.e.
    of_property_read_<type> functions) rather than low-level
    of_get_property/of_find_property functions for reading properties. As
    part of this, convert of_get_property/of_find_property calls to the
    recently added of_property_present() helper when we just want to test
    for presence of a property and nothing more.
```

```
    Signed-off-by: Rob Herring <robh@kernel.org>
    [mpe: Drop change in ppc4xx_probe_pci_bridge(), formatting]
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Link: https://msgid.link/20230310144657.1541039-1-robh@kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c85eece32672cc81b5b0fea4700e5af6ab44a910
Author: Rodríguez Barbarin, José Javier <JoseJavier.Rodriguez@duagon.com>
Date:   Tue Apr 11 10:33:28 2023 +0200

    mcb-pci: Reallocate memory region to avoid memory overlapping

    [ Upstream commit 9be24faadd085c284890c3afcec7a0184642315a ]

    mcb-pci requests a fixed-size memory region to parse the chameleon
    table, however, if the chameleon table is smaller that the allocated
    region, it could overlap with the IP Cores' memory regions.

    After parsing the chameleon table, drop/reallocate the memory region
    with the actual chameleon table size.

    Co-developed-by: Jorge Sanjuan Garcia <jorge.sanjuangarcia@duagon.com>
    Signed-off-by: Jorge Sanjuan Garcia <jorge.sanjuangarcia@duagon.com>
    Signed-off-by: Javier Rodriguez <josejavier.rodriguez@duagon.com>
    Signed-off-by: Johannes Thumshirn <jth@kernel.org>
    Link: https://lore.kernel.org/r/20230411083329.4506-3-jth@kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8e596aed5f2f98cf3e6e98d6fe1d689f4a319308
Author: Tony Lindgren <tony@atomide.com>
Date:   Tue Apr 18 13:14:06 2023 +0300

    serial: 8250: Reinit port->pm on port specific driver unbind

    [ Upstream commit 04e82793f068d2f0ffe62fcea03d007a8cdc16a7 ]

    When we unbind a serial port hardware specific 8250 driver, the generic
    serial8250 driver takes over the port. After that we see an oops about 10
    seconds later. This can produce the following at least on some TI SoCs:

    Unhandled fault: imprecise external abort (0x1406)
    Internal error: : 1406 [#1] SMP ARM

    Turns out that we may still have the serial port hardware specific driver
    port->pm in use, and serial8250_pm() tries to call it after the port
    specific driver is gone:

    serial8250_pm [8250_base] from uart_change_pm+0x54/0x8c [serial_base]
    uart_change_pm [serial_base] from uart_hangup+0x154/0x198 [serial_base]
    uart_hangup [serial_base] from __tty_hangup.part.0+0x328/0x37c
    __tty_hangup.part.0 from disassociate_ctty+0x154/0x20c
    disassociate_ctty from do_exit+0x744/0xaac
    do_exit from do_group_exit+0x40/0x8c
    do_group_exit from __wake_up_parent+0x0/0x1c

    Let's fix the issue by calling serial8250_set_defaults() in
    serial8250_unregister_port(). This will set the port back to using
    the serial8250 default functions, and sets the port->pm to point to
    serial8250_pm.

    Signed-off-by: Tony Lindgren <tony@atomide.com>
    Link: https://lore.kernel.org/r/20230418101407.12403-1-tony@atomide.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 224878b2b47c017f7eda9849b66c323d5b62f83e
Author: Frank Wang <frank.wang@rock-chips.com>
Date:   Thu Mar 16 16:11:49 2023 +0800

    usb: typec: tcpm: fix multiple times discover svids error

    [ Upstream commit dac3b192107b978198e89ec0f77375738352e0c8 ]

    PD3.0 Spec 6.4.4.3.2 say that only Responder supports 12 or more SVIDs,
    the Discover SVIDs Command Shall be executed multiple times until a
    Discover SVIDs VDO is returned ending either with a SVID value of
    0x0000 in the last part of the last VDO or with a VDO containing two
    SVIDs with values of 0x0000.

    In the current implementation, if the last VDO does not find that the
    Discover SVIDs Command would be executed multiple times even if the
    Responder SVIDs are less than 12, and we found some odd dockers just
    meet this case. So fix it.

    Acked-by: Heikki Krogerus <heikki.krogerus@linux.intel.com>
    Signed-off-by: Frank Wang <frank.wang@rock-chips.com>
    Link: https://lore.kernel.org/r/20230316081149.24519-1-frank.wang@rock-chips.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0220e651af376d2ef9f84606f5dfb6bc20847a60
Author: Jason Gerecke <killertofu@gmail.com>
Date:   Thu Apr 13 11:17:43 2023 -0700

    HID: wacom: generic: Set battery quirk only when we see battery data

    [ Upstream commit bea407a427baa019758f29f4d31b26f008bb8cc6 ]

    Some devices will include battery status usages in the HID descriptor
    but we won't see that battery data for one reason or another. For example,
    AES sensors won't send battery data unless an AES pen is in proximity.
    If a user does not have an AES pen but instead only interacts with the
    AES touchscreen with their fingers then there is no need for us to create
    a battery object. Similarly, if a family of peripherals shares the same
    HID descriptor between wired-only and wireless-capable SKUs, users of the
    former may never see a battery event and will not want a power_supply
    object created.

    Link: https://bugzilla.kernel.org/show_bug.cgi?id=217062
    Link: https://gitlab.gnome.org/GNOME/gnome-control-center/-/issues/2354
    Signed-off-by: Jason Gerecke <jason.gerecke@wacom.com>
    Tested-by: Mario Limonciello <mario.limonciello@amd.com>
    Signed-off-by: Jiri Kosina <jkosina@suse.cz>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 438d469359059a850d13d0deaa802fb986d96252
Author: weiliang1503 <weiliang1503@gmail.com>
Date:   Thu Mar 30 19:56:38 2023 +0800

    HID: Ignore battery for ELAN touchscreen on ROG Flow X13 GV301RA

    [ Upstream commit 35903009dbde804a1565dc89e431c0f15179f054 ]

    Ignore the reported battery level of the built-in touchscreen to suppress
    battery warnings when a stylus is used. The device ID was added and the
    battery ignore quirk was enabled.

    Signed-off-by: weiliang1503 <weiliang1503@gmail.com>
    Link: https://lore.kernel.org/r/20230330115638.16146-1-weiliang1503@gmail.com
```

```
    Signed-off-by: Benjamin Tissoires <benjamin.tissoires@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit efc3ab20e65aa521ddad86933bcec0e96eedb8c1
Author: Alex Henrie <alexhenrie24@gmail.com>
Date:   Mon Apr 3 20:48:29 2023 -0600

    HID: apple: Set the tilde quirk flag on the Geyser 3

    [ Upstream commit 29e1ecc197d410ee59c8877098d54cf417075f7d ]

    I was finally able to obtain a MacBook1,1 to test and I've now confirmed
    that it has the tilde key quirk as well:

    Product    Model  Year  System      CPU    Shape  Labels     Country  Quirky
    ================================================================================
    05ac:0218  A1181  2006  MacBook1,1  T2500  ISO    British    13       Yes

    Signed-off-by: Alex Henrie <alexhenrie24@gmail.com>
    Link: https://lore.kernel.org/r/20230404024829.13982-1-alexhenrie24@gmail.com
    Signed-off-by: Benjamin Tissoires <benjamin.tissoires@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5f6f1d329744ef5bf613cd942e3b0799a48cf30a
Author: Syed Saba Kareem <Syed.SabaKareem@amd.com>
Date:   Wed Apr 12 14:46:16 2023 +0530

    ASoC: amd: Add check for acp config flags

    [ Upstream commit bddcfb0802eb69b0f51293eab5db33d344c0262f ]

    We have SOF and generic ACP support enabled for Rembrandt and
    pheonix platforms on some machines. Since we have same PCI id
    used for probing, add check for machine configuration flag to
    avoid conflict with newer pci drivers. Such machine flag has
    been initialized via dmi match on few Chrome machines. If no
    flag is specified probe and register older platform device.

    Signed-off-by: Syed Saba Kareem <Syed.SabaKareem@amd.com>
    Reviewed-by: Vijendar Mukunda <Vijendar.Mukunda@amd.com>
    Link: https://lore.kernel.org/r/20230412091638.1158901-1-Syed.SabaKareem@amd.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit df3fdfd2ea31b3a4e3de24acd0cf365c522bd419
Author: Baishan Jiang <bjiang400@outlook.com>
Date:   Wed Apr 12 16:40:43 2023 +0800

    ASoC: amd: yc: Add ThinkBook 14 G5+ ARP to quirks list for acp6x

    [ Upstream commit a8f5da0bf4d85a6ad03810d902aba61c572102a6 ]

    ThinkBook 14 G5+ ARP uses Ryzen 7735H processor, and has the same
    microphone problem as ThinkBook 14 G4+ ARA.

    Adding 21HY to acp6x quirks table enables microphone for ThinkBook
    14 G5+ ARP.

    Signed-off-by: Baishan Jiang <bjiang400@outlook.com>
    Link:
https://lore.kernel.org/r/OS3P286MB1711DD6556284B69C79C0C4FE19B9@OS3P286MB1711.JPNP286.PROD.
OUTLOOK.COM
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 23326857e939011df60420e9c0090f2a0b197016
Author: Cem Kaya <cemkaya.boun@gmail.com>
Date:   Mon Apr 10 20:38:15 2023 +0200

    ASoC: amd: Add Dell G15 5525 to quirks list

    [ Upstream commit faf15233e59052f4d61cad2da6e56daf33124d96 ]

    Add Dell G15 5525 Ryzen Edition to quirks list for acp6x so that
    internal mic works.

    Link: https://bugzilla.kernel.org/show_bug.cgi?id=217155
    Signed-off-by: Cem Kaya <cemkaya.boun@gmail.com>
    Link: https://lore.kernel.org/r/20230410183814.260518-1-cemkaya.boun@gmail.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 27c64464295c527f0fdb66bbdafa8d3b784fec8a
Author: Fred Oh <fred.oh@linux.intel.com>
Date:   Thu Apr 6 10:25:00 2023 -0500

    ALSA: hda: LNL: add HD Audio PCI ID

    [ Upstream commit 714b2f025d767e7df1fe9da18bd70537d64cc157 ]

    Add HD Audio PCI ID for Intel Lunarlake platform.

    Signed-off-by: Fred Oh <fred.oh@linux.intel.com>
    Signed-off-by: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com>
    Reviewed-by: Péter Ujfalusi <peter.ujfalusi@linux.intel.com>
    Reviewed-by: Bard Liao <yung-chuan.liao@linux.intel.com>
    Link: https://lore.kernel.org/r/20230406152500.15104-1-pierre-
louis.bossart@linux.intel.com
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4a0f5c4db37791002e03a060990f3ec57f9f998f
Author: Samuel Čavoj <samuel@cavoj.net>
Date:   Wed Apr 5 16:44:56 2023 +0300

    usb: typec: ucsi: acpi: add quirk for ASUS Zenbook UM325

    [ Upstream commit 326e1c208f3f24d14b93f910b8ae32c94923d22c ]

    On some ACPI platforms (namely the ASUS Zenbook UM325) the _DSM method must
    not be called after a notification is received but instead the mailbox
    should be read immediately from RAM. This is because the ACPI interrupt
    handler destroys the CCI in ERAM after copying to system memory, and when
    _DSM is later called to perform a second copy, it retrieves a garbage
    value.

    Instead, the _DSM(read) method should only be called when necessary, i.e.
    for polling the state after reset and for retrieving the version. Other
    reads should not call _DSM and only peek into the RAM region.

    This adds a separate read operation for the Zenbook that syncs the
    ACPI mailbox only with polled commands.

    Link: https://lore.kernel.org/linux-
usb/20210823180626.tb6m7h5tp6adhvt2@fastboi.localdomain/
    Signed-off-by: Samuel Čavoj <samuel@cavoj.net>
    [ heikki : handling everything in ucsi_acpi.c with DMI quirk ]
    Signed-off-by: Heikki Krogerus <heikki.krogerus@linux.intel.com>
    Link: https://lore.kernel.org/r/20230405134456.49607-1-heikki.krogerus@linux.intel.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

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

commit 4f73c5544a314d1b399cb3b4caf2a9ff43f7c96e
Author: Kevin Groeneveld <kgroeneveld@lenbrook.com>
Date:   Sat Mar 18 18:21:32 2023 -0400

    spi: spi-imx: fix MX51_ECSPI_* macros when cs > 3

    [ Upstream commit 87c614175bbf28d3fd076dc2d166bac759e41427 ]

    When using gpio based chip select the cs value can go outside the range
    0 – 3. The various MX51_ECSPI_* macros did not take this into consideration
    resulting in possible corruption of the configuration.

    For example for any cs value over 3 the SCLKPHA bits would not be set and
    other values in the register possibly corrupted.

    One way to fix this is to just mask the cs bits to 2 bits. This still
    allows all 4 native chip selects to work as well as gpio chip selects
    (which can use any of the 4 chip select configurations).

    Signed-off-by: Kevin Groeneveld <kgroeneveld@lenbrook.com>
    Link: https://lore.kernel.org/r/20230318222132.3373-1-kgroeneveld@lenbrook.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0fcc564763303af202d0640c2370c64a5399905d
Author: Bastien Nocera <hadess@hadess.net>
Date:   Thu Mar 2 14:01:17 2023 +0100

    HID: logitech-hidpp: Reconcile USB and Unifying serials

    [ Upstream commit 5b3691d15e04b6d5a32c915577b8dbc5cfb56382 ]

    Now that USB HID++ devices can gather a serial number that matches the
    one that would be gathered when connected through a Unifying receiver,
    remove the last difference by dropping the product ID as devices
    usually have different product IDs when connected through USB or
    Unifying.

    For example, on the serials on a G903 wired/wireless mouse:
    - Unifying before patch: 4067-e8-ce-cd-45
    - USB before patch: c086-e8-ce-cd-45
    - Unifying and USB after patch: e8-ce-cd-45

    Signed-off-by: Bastien Nocera <hadess@hadess.net>
    Link: https://lore.kernel.org/r/20230302130117.3975-2-hadess@hadess.net
    Signed-off-by: Benjamin Tissoires <benjamin.tissoires@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 445e590f841db23eb178fe2637d8587f90152f3c
Author: Bastien Nocera <hadess@hadess.net>
Date:   Thu Mar 2 14:01:16 2023 +0100

    HID: logitech-hidpp: Don't use the USB serial for USB devices

    [ Upstream commit 7ad1fe0da0fa91bf920b79ab05ae97bfabecc4f4 ]

    For devices that support the 0x0003 feature (Device Information) version 4,
    set the serial based on the output of that feature, rather than relying
    on the usbhid code setting the USB serial.

    This should allow the serial when connected through USB to (nearly)
    match the one when connected through a unifying receiver.
```

```
    For example, on the serials on a G903 wired/wireless mouse:
    - Unifying: 4067-e8-ce-cd-45
    - USB before patch: 017C385C3837
    - USB after patch: c086-e8-ce-cd-45

    Signed-off-by: Bastien Nocera <hadess@hadess.net>
    Link: https://lore.kernel.org/r/20230302130117.3975-1-hadess@hadess.net
    Signed-off-by: Benjamin Tissoires <benjamin.tissoires@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f17e035b3d47962e22218fb6dbe08d3679912e0e
Author: Prajna Sariputra <putr4.s@gmail.com>
Date:   Sun Apr 2 02:21:30 2023 +1100

    ASoC: amd: yc: Add DMI entries to support HP OMEN 16-n0xxx (8A42)

    [ Upstream commit ee4281de4d60288b9c802bb0906061ec355ecef2 ]

    This model requires an additional detection quirk to enable the internal microphone.

    Signed-off-by: Prajna Sariputra <putr4.s@gmail.com>
    Link: https://lore.kernel.org/r/2283110.ElGaqSPkdT@n0067ax-linux62
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f6c367ddb08a9e70f9d810b0c91081840faa4d89
Author: Mika Westerberg <mika.westerberg@linux.intel.com>
Date:   Fri Mar 31 08:28:12 2023 +0300

    spi: intel-pci: Add support for Meteor Lake-S SPI serial flash

    [ Upstream commit c2912d42e86e494935722669e4d9eade69649072 ]

    Intel Meteor Lake-S has the same SPI serial flash controller as Meteor
    Lake-P. Add Meteor Lake-S PCI ID to the driver list of supported
    devices.

    Signed-off-by: Mika Westerberg <mika.westerberg@linux.intel.com>
    Link: https://lore.kernel.org/r/20230331052812.39983-1-mika.westerberg@linux.intel.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 178ed14a98a9f252cb6659ed7ed2ebb3da891b4a
Author: Andy Shevchenko <andriy.shevchenko@linux.intel.com>
Date:   Fri Mar 17 17:47:02 2023 +0200

    xhci: mem: Carefully calculate size for memory allocations

    [ Upstream commit 347284984f415e52590373253c6943bbdc806ebf ]

    Carefully calculate size for memory allocations, i.e. with help
    of size_mul() macro from overflow.h.

    Signed-off-by: Andy Shevchenko <andriy.shevchenko@linux.intel.com>
    Signed-off-by: Mathias Nyman <mathias.nyman@linux.intel.com>
    Link: https://lore.kernel.org/r/20230317154715.535523-2-mathias.nyman@linux.intel.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 72fcb43814d8f3ae1f826ddb2dcf71bda2eb3667
Author: Khadija Kamran <kamrankhadijadj@gmail.com>
Date:   Fri Mar 17 01:09:00 2023 +0500

    staging: axis-fifo: initialize timeouts in init only
```

```
    [ Upstream commit 752cbd8f191678e86aa754f795546b7f06b7f171 ]

    Initialize the module parameters, read_timeout and write_timeout once in
    init().

    Module parameters can only be set once and cannot be modified later, so we
    don't need to evaluate them again when passing the parameters to
    wait_event_interruptible_timeout().

    Convert datatype of {read,write}_timeout from 'int' to 'long int' because
    implicit conversion of 'long int' to 'int' in statement
    '{read,write}_timeout = MAX_SCHEDULE_TIMEOUT' results in an overflow.

    Change format specifier for {read,write}_timeout from %i to %li.

    Reviewed-by: Fabio M. De Francesco <fmdefrancesco@gmail.com>
    Suggested-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Khadija Kamran <kamrankhadijadj@gmail.com>
    Link: https://lore.kernel.org/r/ZBN3XAsItCiTk7CV@khadija-virtual-machine
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5357c16917ec55901c56da579d1379449ab7a0ba
Author: Lorenzo Bianconi <lorenzo@kernel.org>
Date:   Mon Feb 27 12:14:56 2023 +0100

    iio: imu: st_lsm6dsx: discard samples during filters settling time

    [ Upstream commit db3c490503bee4d0611f9fc17fcd8cfe6fcdbcad ]

    During digital filters settling time the driver is expected to drop
    samples since they can be corrupted. Introduce the capability to drop
    a given number of samples according to the configured ODR.
    Add sample_to_discard for LSM6DSM-like sensors since new generation
    devices (e.g. LSM6DSO) support DRDY mask where corrupted samples are
    masked in hw with values greather than 0x7ffd so the driver can easily
    discard them.
    I have not added sample_to_discard support for LSM6DS3 or LSM6DS3H since
    I do not have any sample for testing at the moment.

    Reported-by: Philippe De Muyter <phdm@macqel.be>
    Tested-by: Philippe De Muyter <phdm@macqel.be>
    Signed-off-by: Lorenzo Bianconi <lorenzo@kernel.org>
    Link:
https://lore.kernel.org/r/21dcd94935c147ef9b1da4984b3da6264ee9609e.1677496295.git.lorenzo@ke
rnel.org
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit af04b4541a1bbaf13ad13e6220d1e68b66332ce1
Author: Alex Henrie <alexhenrie24@gmail.com>
Date:   Sun Feb 26 20:06:13 2023 -0700

    HID: apple: Set the tilde quirk flag on the Geyser 4 and later

    [ Upstream commit c3388ddc74a863466c7c3fa24d3a9cea9c9bca53 ]

    I recently tested several old MacBooks and as far as I can tell, all
    MacBooks that have an ISO keyboard have the tilde key quirk:

    Product    Model  Year  System      CPU    Shape  Labels     Country  Quirky
    ================================================================================
    05ac:021b  A1181  2006  MacBook2,1  T5600  ISO    British    13       Yes
    05ac:021b  A1181  2007  MacBook2,1  T7200  ISO    Québécois  13       Yes
    05ac:0229  A1181  2007  MacBook4,1  T8300  ANSI   Usonian    33       No
```

```
      05ac:022a  A1181  2007  MacBook4,1  T8100  ISO    English    13       Yes
      05ac:022a  A1181  2007  MacBook5,2  P7350  ISO    Québécois  13       Yes
      05ac:0237  A1278  2008  MacBook5,1  P7350  ISO    Dutch      13       Yes
      05ac:0237  A1278  2009  MacBook5,5  P7550  ISO    British    13       Yes

      The model number and year are from the laptop case. Since Apple printed
      the same model and year on many different laptops, the system name (as
      reported in the SMBIOS tables) and CPU form a more precise identifier.

      Signed-off-by: Alex Henrie <alexhenrie24@gmail.com>
      Signed-off-by: Jiri Kosina <jkosina@suse.cz>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 21d58e5ac3062e931d9f5a9eb58a6caacb910856
  Author: Philipp Hortmann <philipp.g.hortmann@gmail.com>
  Date:   Thu Feb 23 07:47:21 2023 +0100

      staging: rtl8192e: Replace macro RTL_PCI_DEVICE with PCI_DEVICE

      [ Upstream commit fda2093860df4812d69052a8cf4997e53853a340 ]

      Replace macro RTL_PCI_DEVICE with PCI_DEVICE to get rid of rtl819xp_ops
      which is empty.

      Signed-off-by: Philipp Hortmann <philipp.g.hortmann@gmail.com>
      Link:
  https://lore.kernel.org/r/8b45ee783fa91196b7c9d6fc840a189496afd2f4.1677133271.git.philipp.g.
  hortmann@gmail.com
      Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit acd11f662c2b3b9f8aa83276788a64ce9c5ea2cd
  Author: Max Chou <max.chou@realtek.com>
  Date:   Tue Apr 18 13:43:54 2023 +0800

      Bluetooth: btrtl: Add the support for RTL8851B

      [ Upstream commit 7948fe1c92d92313eea5453f83deb7f0141355e8 ]

      Add the support for RTL8851B BT controller on USB interface.
      The necessary firmware will be submitted to linux-firmware project.

      Note that the Bluetooth devices WITH the VID=0x0bda would be set the
      feature quirk in btrtl_setup_realtek(). It's able to ignore the
      feature flag set for the specific VID and PID in blacklist_table[] of
      btusb.c. (check [1])

      If Realtek Bluetooth chips WITHOUT the VID=0x0bda, it shall be added
      the feature flag for the specific VID and PID in blacklist_table[] of
      btusb.c. (check [2])

      [1] '9ab9235fe5cf ("Bluetooth: btrtl: Enable WBS for the specific
          Realtek devices")'
      [2] '73280f13c9bb ("Bluetooth: btusb: Add the more support IDs for
          Realtek RTL8822CE")'

      The device info from /sys/kernel/debug/usb/devices as below.

      T:  Bus=03 Lev=01 Prnt=01 Port=00 Cnt=01 Dev#= 33 Spd=480  MxCh= 0
      D:  Ver= 2.00 Cls=ef(misc ) Sub=02 Prot=01 MxPS=64 #Cfgs=  1
      P:  Vendor=0bda ProdID=b851 Rev= 0.00
      S:  Manufacturer=Realtek
      S:  Product=802.11ax WLAN Adapter
      S:  SerialNumber=00E04C885A01
      C:* #Ifs= 3 Cfg#= 1 Atr=80 MxPwr=500mA
```

```
      A:  FirstIf#= 0 IfCount= 2 Cls=e0(wlcon) Sub=01 Prot=01
      I:* If#= 0 Alt= 0 #EPs= 3 Cls=e0(wlcon) Sub=01 Prot=01 Driver=btusb
      E:  Ad=81(I) Atr=03(Int.) MxPS=  16 Ivl=1ms
      E:  Ad=02(O) Atr=02(Bulk) MxPS= 512 Ivl=0ms
      E:  Ad=82(I) Atr=02(Bulk) MxPS= 512 Ivl=0ms
      I:* If#= 1 Alt= 0 #EPs= 2 Cls=e0(wlcon) Sub=01 Prot=01 Driver=btusb
      E:  Ad=03(O) Atr=01(Isoc) MxPS=   0 Ivl=1ms
      E:  Ad=83(I) Atr=01(Isoc) MxPS=   0 Ivl=1ms
      I:  If#= 1 Alt= 1 #EPs= 2 Cls=e0(wlcon) Sub=01 Prot=01 Driver=btusb
      E:  Ad=03(O) Atr=01(Isoc) MxPS=   9 Ivl=1ms
      E:  Ad=83(I) Atr=01(Isoc) MxPS=   9 Ivl=1ms
      I:  If#= 1 Alt= 2 #EPs= 2 Cls=e0(wlcon) Sub=01 Prot=01 Driver=btusb
      E:  Ad=03(O) Atr=01(Isoc) MxPS=  17 Ivl=1ms
      E:  Ad=83(I) Atr=01(Isoc) MxPS=  17 Ivl=1ms
      I:  If#= 1 Alt= 3 #EPs= 2 Cls=e0(wlcon) Sub=01 Prot=01 Driver=btusb
      E:  Ad=03(O) Atr=01(Isoc) MxPS=  25 Ivl=1ms
      E:  Ad=83(I) Atr=01(Isoc) MxPS=  25 Ivl=1ms
      I:  If#= 1 Alt= 4 #EPs= 2 Cls=e0(wlcon) Sub=01 Prot=01 Driver=btusb
      E:  Ad=03(O) Atr=01(Isoc) MxPS=  33 Ivl=1ms
      E:  Ad=83(I) Atr=01(Isoc) MxPS=  33 Ivl=1ms
      I:  If#= 1 Alt= 5 #EPs= 2 Cls=e0(wlcon) Sub=01 Prot=01 Driver=btusb
      E:  Ad=03(O) Atr=01(Isoc) MxPS=  49 Ivl=1ms
      E:  Ad=83(I) Atr=01(Isoc) MxPS=  49 Ivl=1ms
      I:* If#= 2 Alt= 0 #EPs= 8 Cls=ff(vend.) Sub=ff Prot=ff Driver=(none)
      E:  Ad=84(I) Atr=02(Bulk) MxPS= 512 Ivl=0ms
      E:  Ad=05(O) Atr=02(Bulk) MxPS= 512 Ivl=0ms
      E:  Ad=06(O) Atr=02(Bulk) MxPS= 512 Ivl=0ms
      E:  Ad=07(O) Atr=02(Bulk) MxPS= 512 Ivl=0ms
      E:  Ad=09(O) Atr=02(Bulk) MxPS= 512 Ivl=0ms
      E:  Ad=0a(O) Atr=02(Bulk) MxPS= 512 Ivl=0ms
      E:  Ad=0b(O) Atr=02(Bulk) MxPS= 512 Ivl=0ms
      E:  Ad=0c(O) Atr=02(Bulk) MxPS= 512 Ivl=0ms

      Signed-off-by: Max Chou <max.chou@realtek.com>
      Signed-off-by: Luiz Augusto von Dentz <luiz.von.dentz@intel.com>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 5134556c9be582793f30695c09d18a26fe1ff2d7
  Author: Min Li <lm0963hack@gmail.com>
  Date:   Mon Apr 17 10:27:54 2023 +0800

      Bluetooth: L2CAP: fix "bad unlock balance" in l2cap_disconnect_rsp

      [ Upstream commit 25e97f7b1866e6b8503be349eeea44bb52d661ce ]

      conn->chan_lock isn't acquired before l2cap_get_chan_by_scid,
      if l2cap_get_chan_by_scid returns NULL, then 'bad unlock balance'
      is triggered.

      Reported-by: syzbot+9519d6b5b79cf7787cf3@syzkaller.appspotmail.com
      Link: https://lore.kernel.org/all/000000000000894f5f05f95e9f4d@google.com/
      Signed-off-by: Min Li <lm0963hack@gmail.com>
      Signed-off-by: Luiz Augusto von Dentz <luiz.von.dentz@intel.com>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 477a816b31a2150650e3cd34145e9044120868b9
  Author: Raul Cheleguini <raul.cheleguini@gmail.com>
  Date:   Thu Mar 23 10:45:39 2023 -0300

      Bluetooth: Add new quirk for broken set random RPA timeout for ATS2851

      [ Upstream commit 91b6d02ddcd113352bdd895990b252065c596de7 ]

      The ATS2851 based controller advertises support for command "LE Set Random
      Private Address Timeout" but does not actually implement it, impeding the
```

```
    controller initialization.

    Add the quirk HCI_QUIRK_BROKEN_SET_RPA_TIMEOUT to unblock the controller
    initialization.

    < HCI Command: LE Set Resolvable Private... (0x08|0x002e) plen 2
            Timeout: 900 seconds
    > HCI Event: Command Status (0x0f) plen 4
          LE Set Resolvable Private Address Timeout (0x08|0x002e) ncmd 1
            Status: Unknown HCI Command (0x01)

    Co-developed-by: imoc <wzj9912@gmail.com>
    Signed-off-by: imoc <wzj9912@gmail.com>
    Signed-off-by: Raul Cheleguini <raul.cheleguini@gmail.com>
    Signed-off-by: Luiz Augusto von Dentz <luiz.von.dentz@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 37275ce35b439cbfbc606694370f9fa33ffb66e0
Author: Hans de Goede <hdegoede@redhat.com>
Date:   Fri Mar 31 23:11:21 2023 +0200

    Bluetooth: hci_bcm: Fall back to getting bdaddr from EFI if not set

    [ Upstream commit 0d218c3642b9ccf71f44987cd03c19320f3bd918 ]

    On some devices the BCM Bluetooth adapter does not have a valid bdaddr set.

    btbcm.c currently sets HCI_QUIRK_INVALID_BDADDR to indicate when this is
    the case. But this requires users to manual setup a btaddr, by doing e.g.:

    btmgmt -i hci0 public-addr 'B0:F1:EC:82:1D:B3'

    Which means that Bluetooth will not work out of the box on such devices.
    To avoid this (where possible) hci_bcm sets: HCI_QUIRK_USE_BDADDR_PROPERTY
    which tries to get the bdaddr from devicetree.

    But this only works on devicetree platforms. On UEFI based platforms
    there is a special Broadcom UEFI variable which when present contains
    the devices bdaddr, just like how there is another UEFI variable which
    contains wifi nvram contents including the wifi MAC address.

    Add support for getting the bdaddr from this Broadcom UEFI variable,
    so that Bluetooth will work OOTB for users on devices where this
    UEFI variable is present.

    This fixes Bluetooth not working on for example Asus T100HA 2-in-1s.

    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Luiz Augusto von Dentz <luiz.von.dentz@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 476b9564ac6c6c35b42df4014e50eb37fad03bd5
Author: Chethan T N <chethan.tumkur.narayan@intel.com>
Date:   Tue Mar 21 10:03:10 2023 +0530

    Bluetooth: btintel: Add LE States quirk support

    [ Upstream commit 77f542b10c535c9a93bf8afdd2665524935807c2 ]

    Basically all Intel controllers support both Central/Peripheral
    LE states.

    This patch enables the LE States quirk by default on all
    Solar and Magnertor Intel controllers.
```

```
    Signed-off-by: Chethan T N <chethan.tumkur.narayan@intel.com>
    Signed-off-by: Luiz Augusto von Dentz <luiz.von.dentz@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c34722f0bb9f7efb0e7e7a75a9cb57601132b51f
Author: Max Chou <max.chou@realtek.com>
Date:   Tue Mar 21 19:48:26 2023 +0800

    Bluetooth: btrtl: check for NULL in btrtl_set_quirks()

    [ Upstream commit 253cf30e8d3d001850a95c4729d668f916b037ab ]

    The btrtl_set_quirks() has accessed btrtl_dev->ic_info->lmp_subver since
    b8e482d02513. However, if installing a Realtek Bluetooth controller
    without the driver supported, it will hit the NULL point accessed.

    Add a check for NULL to avoid the Kernel Oops.

    Signed-off-by: Max Chou <max.chou@realtek.com>
    Signed-off-by: Luiz Augusto von Dentz <luiz.von.dentz@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a1c967d9602c447592dd514ea430f32b7f52929a
Author: Raul Cheleguini <rcheleguini@google.com>
Date:   Fri Mar 10 15:14:10 2023 +0000

    Bluetooth: Improve support for Actions Semi ATS2851 based devices

    [ Upstream commit 7c2b2d2d0cb658aa543e11e90ae95621d3cb5fe6 ]

    Add two more quirks to resume the device initialization and basic
    operation as the device seems not to support "Read Transmit Power"
    and "Set Extended Scan Parameters".

    < HCI Command: LE Read Transmit Power (0x08|0x004b) plen 0
    > HCI Event: Command Status (0x0f) plen 4
          LE Read Transmit Power (0x08|0x004b) ncmd 1
            Status: Unknown HCI Command (0x01)

    < HCI Command: LE Set Extended Scan Parameters (0x08|0x0041) plen 8
            Own address type: Random (0x01)
            Filter policy: Accept all advertisement (0x00)
            PHYs: 0x01
            Entry 0: LE 1M
              Type: Active (0x01)
              Interval: 11.250 msec (0x0012)
              Window: 11.250 msec (0x0012)
    > HCI Event: Command Status (0x0f) plen 4
          LE Set Extended Scan Parameters (0x08|0x0041) ncmd 1
            Status: Unknown HCI Command (0x01)

    Signed-off-by: Raul Cheleguini <rcheleguini@google.com>
    Signed-off-by: Luiz Augusto von Dentz <luiz.von.dentz@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 140d4a7c4709ae7445fd655a787bd240151a2040
Author: Vasily Khoruzhick <anarsoul@gmail.com>
Date:   Tue Mar 7 23:17:31 2023 +0100

    Bluetooth: btrtl: add support for the RTL8723CS

    [ Upstream commit c0123cb6c4c7fc2a42ead6cd7d3e82b8e1c25c6f ]

    The Realtek RTL8723CS is a SDIO WiFi chip. It also contains a Bluetooth
    module which is connected via UART to the host.
```

```
    It shares lmp subversion with 8703B, so Realtek's userspace
    initialization tool (rtk_hciattach) differentiates varieties of RTL8723CS
    (CG, VF, XX) with RTL8703B using vendor's command to read chip type.

    Also this chip declares support for some features it doesn't support
    so add a quirk to indicate that these features are broken.

    Signed-off-by: Vasily Khoruzhick <anarsoul@gmail.com>
    Signed-off-by: Bastian Germann <bage@debian.org>
    Signed-off-by: Luiz Augusto von Dentz <luiz.von.dentz@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7f9d9c3fae54ffac72188ef3350fd973cd98180f
Author: Vasily Khoruzhick <anarsoul@gmail.com>
Date:   Tue Mar 7 23:17:30 2023 +0100

    Bluetooth: Add new quirk for broken local ext features page 2

    [ Upstream commit 8194f1ef5a815aea815a91daf2c721eab2674f1f ]

    Some adapters (e.g. RTL8723CS) advertise that they have more than
    2 pages for local ext features, but they don't support any features
    declared in these pages. RTL8723CS reports max_page = 2 and declares
    support for sync train and secure connection, but it responds with
    either garbage or with error in status on corresponding commands.

    Signed-off-by: Vasily Khoruzhick <anarsoul@gmail.com>
    Signed-off-by: Bastian Germann <bage@debian.org>
    Signed-off-by: Luiz Augusto von Dentz <luiz.von.dentz@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0af7505f98818e03f99c1d619d68813656eb1556
Author: Meng Tang <tangmeng@uniontech.com>
Date:   Tue Feb 28 13:55:17 2023 +0800

    Bluetooth: btusb: Add new PID/VID 04ca:3801 for MT7663

    [ Upstream commit 13209415d0e88396d99d346b184864834d70d68a ]

    This bluetooth device is found in a combo WLAN/BT card
    for a MediaTek 7663.

    Tested on Acer Aspire A315-24P Notebook

    The device information:

    T:  Bus=01 Lev=01 Prnt=01 Port=01 Cnt=01 Dev#=  2 Spd=480  MxCh= 0
    D:  Ver= 2.10 Cls=ef(misc ) Sub=02 Prot=01 MxPS=64 #Cfgs=  1
    P:  Vendor=04ca ProdID=3801 Rev= 1.00
    S:  Manufacturer=MediaTek Inc.
    S:  Product=Wireless_Device
    S:  SerialNumber=000000000
    C:* #Ifs= 2 Cfg#= 1 Atr=e0 MxPwr=100mA
    A:  FirstIf#= 0 IfCount= 2 Cls=e0(wlcon) Sub=01 Prot=01
    I:* If#= 0 Alt= 0 #EPs= 3 Cls=e0(wlcon) Sub=01 Prot=01 Driver=btusb
    E:  Ad=81(I) Atr=03(Int.) MxPS=  16 Ivl=125us
    E:  Ad=82(I) Atr=02(Bulk) MxPS= 512 Ivl=0ms
    E:  Ad=02(O) Atr=02(Bulk) MxPS= 512 Ivl=0ms
    I:* If#= 1 Alt= 0 #EPs= 2 Cls=e0(wlcon) Sub=01 Prot=01 Driver=btusb
    E:  Ad=83(I) Atr=01(Isoc) MxPS=   0 Ivl=1ms
    E:  Ad=03(O) Atr=01(Isoc) MxPS=   0 Ivl=1ms
    I:  If#= 1 Alt= 1 #EPs= 2 Cls=e0(wlcon) Sub=01 Prot=01 Driver=btusb
    E:  Ad=83(I) Atr=01(Isoc) MxPS=   9 Ivl=1ms
    E:  Ad=03(O) Atr=01(Isoc) MxPS=   9 Ivl=1ms
```

```
     I:  If#= 1 Alt= 2 #EPs= 2 Cls=e0(wlcon) Sub=01 Prot=01 Driver=btusb
     E:  Ad=83(I) Atr=01(Isoc) MxPS=  17 Ivl=1ms
     E:  Ad=03(O) Atr=01(Isoc) MxPS=  17 Ivl=1ms
     I:  If#= 1 Alt= 3 #EPs= 2 Cls=e0(wlcon) Sub=01 Prot=01 Driver=btusb
     E:  Ad=83(I) Atr=01(Isoc) MxPS=  25 Ivl=1ms
     E:  Ad=03(O) Atr=01(Isoc) MxPS=  25 Ivl=1ms
     I:  If#= 1 Alt= 4 #EPs= 2 Cls=e0(wlcon) Sub=01 Prot=01 Driver=btusb
     E:  Ad=83(I) Atr=01(Isoc) MxPS=  33 Ivl=1ms
     E:  Ad=03(O) Atr=01(Isoc) MxPS=  33 Ivl=1ms
     I:  If#= 1 Alt= 5 #EPs= 2 Cls=e0(wlcon) Sub=01 Prot=01 Driver=btusb
     E:  Ad=83(I) Atr=01(Isoc) MxPS=  49 Ivl=1ms
     E:  Ad=03(O) Atr=01(Isoc) MxPS=  49 Ivl=1ms
     I:  If#= 1 Alt= 6 #EPs= 2 Cls=e0(wlcon) Sub=01 Prot=01 Driver=btusb
     E:  Ad=83(I) Atr=01(Isoc) MxPS=  63 Ivl=1ms
     E:  Ad=03(O) Atr=01(Isoc) MxPS=  63 Ivl=1ms

     Signed-off-by: Meng Tang <tangmeng@uniontech.com>
     Signed-off-by: Luiz Augusto von Dentz <luiz.von.dentz@intel.com>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 219b69202a41b010c3166b08590e9a233d3d6563
  Author: Simon Horman <horms@kernel.org>
  Date:   Mon Apr 17 17:10:45 2023 +0200

     ipvs: Update width of source for ip_vs_sync_conn_options

     [ Upstream commit e3478c68f6704638d08f437cbc552ca5970c151a ]

     In ip_vs_sync_conn_v0() copy is made to struct ip_vs_sync_conn_options.
     That structure looks like this:

     struct ip_vs_sync_conn_options {
             struct ip_vs_seq        in_seq;
             struct ip_vs_seq        out_seq;
     };

     The source of the copy is the in_seq field of struct ip_vs_conn.  Whose
     type is struct ip_vs_seq. Thus we can see that the source - is not as
     wide as the amount of data copied, which is the width of struct
     ip_vs_sync_conn_option.

     The copy is safe because the next field in is another struct ip_vs_seq.
     Make use of struct_group() to annotate this.

     Flagged by gcc-13 as:

      In file included from ./include/linux/string.h:254,
                       from ./include/linux/bitmap.h:11,
                       from ./include/linux/cpumask.h:12,
                       from ./arch/x86/include/asm/paravirt.h:17,
                       from ./arch/x86/include/asm/cpuid.h:62,
                       from ./arch/x86/include/asm/processor.h:19,
                       from ./arch/x86/include/asm/timex.h:5,
                       from ./include/linux/timex.h:67,
                       from ./include/linux/time32.h:13,
                       from ./include/linux/time.h:60,
                       from ./include/linux/stat.h:19,
                       from ./include/linux/module.h:13,
                       from net/netfilter/ipvs/ip_vs_sync.c:38:
      In function 'fortify_memcpy_chk',
          inlined from 'ip_vs_sync_conn_v0' at net/netfilter/ipvs/ip_vs_sync.c:606:3:
     ./include/linux/fortify-string.h:529:25: error: call to '__read_overflow2_field'
  declared with attribute warning: detected read beyond size of field (2nd parameter); maybe
  use struct_group()? [-Werror=attribute-warning]
        529 |                         __read_overflow2_field(q_size_field, size);
```

```
             |

    Compile tested only.

    Signed-off-by: Simon Horman <horms@kernel.org>
    Reviewed-by: Horatiu Vultur <horatiu.vultur@microchip.com>
    Signed-off-by: Pablo Neira Ayuso <pablo@netfilter.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ffb75ffaa68723276365d0f9d00b03362b750657
Author: Zhong Jinghua <zhongjinghua@huawei.com>
Date:   Mon Feb 6 22:58:05 2023 +0800

    nbd: fix incomplete validation of ioctl arg

    [ Upstream commit 55793ea54d77719a071b1ccc05a05056e3b5e009 ]

    We tested and found an alarm caused by nbd_ioctl arg without verification.
    The UBSAN warning calltrace like below:

    UBSAN: Undefined behaviour in fs/buffer.c:1709:35
    signed integer overflow:
    -9223372036854775808 - 1 cannot be represented in type 'long long int'
    CPU: 3 PID: 2523 Comm: syz-executor.0 Not tainted 4.19.90 #1
    Hardware name: linux,dummy-virt (DT)
    Call trace:
     dump_backtrace+0x0/0x3f0 arch/arm64/kernel/time.c:78
     show_stack+0x28/0x38 arch/arm64/kernel/traps.c:158
     __dump_stack lib/dump_stack.c:77 [inline]
     dump_stack+0x170/0x1dc lib/dump_stack.c:118
     ubsan_epilogue+0x18/0xb4 lib/ubsan.c:161
     handle_overflow+0x188/0x1dc lib/ubsan.c:192
     __ubsan_handle_sub_overflow+0x34/0x44 lib/ubsan.c:206
     __block_write_full_page+0x94c/0xa20 fs/buffer.c:1709
     block_write_full_page+0x1f0/0x280 fs/buffer.c:2934
     blkdev_writepage+0x34/0x40 fs/block_dev.c:607
     __writepage+0x68/0xe8 mm/page-writeback.c:2305
     write_cache_pages+0x44c/0xc70 mm/page-writeback.c:2240
     generic_writepages+0xdc/0x148 mm/page-writeback.c:2329
     blkdev_writepages+0x2c/0x38 fs/block_dev.c:2114
     do_writepages+0xd4/0x250 mm/page-writeback.c:2344

    The reason for triggering this warning is __block_write_full_page()
    -> i_size_read(inode) - 1 overflow.
    inode->i_size is assigned in __nbd_ioctl() -> nbd_set_size() -> bytesize.
    We think it is necessary to limit the size of arg to prevent errors.

    Moreover, __nbd_ioctl() -> nbd_add_socket(), arg will be cast to int.
    Assuming the value of arg is 0x80000000000000001) (on a 64-bit machine),
    it will become 1 after the coercion, which will return unexpected results.

    Fix it by adding checks to prevent passing in too large numbers.

    Signed-off-by: Zhong Jinghua <zhongjinghua@huawei.com>
    Reviewed-by: Yu Kuai <yukuai3@huawei.com>
    Reviewed-by: Josef Bacik <josef@toxicpanda.com>
    Link: https://lore.kernel.org/r/20230206145805.2645671-1-zhongjinghua@huawei.com
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5bba1ad561a8b5bb14704d8f511cf10466336e3d
Author: Larry Finger <Larry.Finger@lwfinger.net>
Date:   Mon Apr 17 11:03:31 2023 -0500

    wifi: rtw88: Fix memory leak in rtw88_usb
```

```
    [ Upstream commit 59a3a312009723e3e5082899655fdcc420e2b47a ]

    Kmemleak shows the following leak arising from routine in the usb
    probe routine:

    unreferenced object 0xffff895cb29bba00 (size 512):
      comm "(udev-worker)", pid 534, jiffies 4294903932 (age 102751.088s)
      hex dump (first 32 bytes):
        77 30 30 30 00 00 00 00 02 2f 2d 2b 30 00 00 00  w000...../-+0...
        02 00 2a 28 00 00 00 00 ff 55 ff ff ff 00 00 00  ..*(.....U......
      backtrace:
        [<ffffffff9265fa36>] kmalloc_trace+0x26/0x90
        [<ffffffffc17eec41>] rtw_usb_probe+0x2f1/0x680 [rtw_usb]
        [<ffffffffc03e19fd>] usb_probe_interface+0xdd/0x2e0 [usbcore]
        [<ffffffff92b4f2fe>] really_probe+0x18e/0x3d0
        [<ffffffff92b4f5b8>] __driver_probe_device+0x78/0x160
        [<ffffffff92b4f6bf>] driver_probe_device+0x1f/0x90
        [<ffffffff92b4f8df>] __driver_attach+0xbf/0x1b0
        [<ffffffff92b4d350>] bus_for_each_dev+0x70/0xc0
        [<ffffffff92b4e51e>] bus_add_driver+0x10e/0x210
        [<ffffffff92b50935>] driver_register+0x55/0xf0
        [<ffffffffc03e0708>] usb_register_driver+0x88/0x140 [usbcore]
        [<ffffffff92401153>] do_one_initcall+0x43/0x210
        [<ffffffff9254f42a>] do_init_module+0x4a/0x200
        [<ffffffff92551d1c>] __do_sys_finit_module+0xac/0x120
        [<ffffffff92ee6626>] do_syscall_64+0x56/0x80
        [<ffffffff9300006a>] entry_SYSCALL_64_after_hwframe+0x46/0xb0

    The leak was verified to be real by unloading the driver, which resulted
    in a dangling pointer to the allocation.

    The allocated memory is freed in rtw_usb_intf_deinit().

    Signed-off-by: Larry Finger <Larry.Finger@lwfinger.net>
    Cc: Sascha Hauer <s.hauer@pengutronix.de>
    Cc: Ping-Ke Shih <pkshih@realtek.com>
    Reviewed-by: Ping-Ke Shih <pkshih@realtek.com>
    Signed-off-by: Kalle Valo <kvalo@kernel.org>
    Link: https://lore.kernel.org/r/20230417160331.23071-1-Larry.Finger@lwfinger.net
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 67459491f78146bcf7d93596e5b709d063dff5d8
Author: Nagarajan Maran <quic_nmaran@quicinc.com>
Date:   Mon Apr 17 13:35:02 2023 +0300

    wifi: ath11k: Fix SKB corruption in REO destination ring

    [ Upstream commit f9fff67d2d7ca6fa8066132003a3deef654c55b1 ]

    While running traffics for a long time, randomly an RX descriptor
    filled with value "0" from REO destination ring is received.
    This descriptor which is invalid causes the wrong SKB (SKB stored in
    the IDR lookup with buffer id "0") to be fetched which in turn
    causes SKB memory corruption issue and the same leads to crash
    after some time.

    Changed the start id for idr allocation to "1" and the buffer id "0"
    is reserved for error validation. Introduced Sanity check to validate
    the descriptor, before processing the SKB.

    Crash Signature :

    Unable to handle kernel paging request at virtual address 3f004900
    PC points to "b15_dma_inv_range+0x30/0x50"
```

```
    LR points to "dma_cache_maint_page+0x8c/0x128".
    The Backtrace obtained is as follows:
    [<8031716c>] (b15_dma_inv_range) from [<80313a4c>] (dma_cache_maint_page+0x8c/0x128)
    [<80313a4c>] (dma_cache_maint_page) from [<80313b90>] (__dma_page_dev_to_cpu+0x28/0xcc)
    [<80313b90>] (__dma_page_dev_to_cpu) from [<7fb5dd68>] (ath11k_dp_process_rx+0x1e8/0x4a4
 [ath11k])
    [<7fb5dd68>] (ath11k_dp_process_rx [ath11k]) from [<7fb53c20>]
 (ath11k_dp_service_srng+0xb0/0x2ac [ath11k])
    [<7fb53c20>] (ath11k_dp_service_srng [ath11k]) from [<7f67bba4>]
 (ath11k_pci_ext_grp_napi_poll+0x1c/0x78 [ath11k_pci])
    [<7f67bba4>] (ath11k_pci_ext_grp_napi_poll [ath11k_pci]) from [<807d5cf4>]
 (__napi_poll+0x28/0xb8)
    [<807d5cf4>] (__napi_poll) from [<807d5f28>] (net_rx_action+0xf0/0x280)
    [<807d5f28>] (net_rx_action) from [<80302148>] (__do_softirq+0xd0/0x280)
    [<80302148>] (__do_softirq) from [<80320408>] (irq_exit+0x74/0xd4)
    [<80320408>] (irq_exit) from [<803638a4>] (__handle_domain_irq+0x90/0xb4)
    [<803638a4>] (__handle_domain_irq) from [<805bedec>] (gic_handle_irq+0x58/0x90)
    [<805bedec>] (gic_handle_irq) from [<80301a78>] (__irq_svc+0x58/0x8c)

    Tested-on: IPQ8074 hw2.0 AHB WLAN.HK.2.7.0.1-01744-QCAHKSWPL_SILICONZ-1

    Signed-off-by: Nagarajan Maran <quic_nmaran@quicinc.com>
    Signed-off-by: Kalle Valo <quic_kvalo@quicinc.com>
    Link: https://lore.kernel.org/r/20230403191533.28114-1-quic_nmaran@quicinc.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit 6cd644f66b43709816561d63e0173cb0c7aab159
 Author: Hans de Goede <hdegoede@redhat.com>
 Date:   Tue Apr 18 15:25:46 2023 +0200

    wifi: iwlwifi: dvm: Fix memcpy: detected field-spanning write backtrace

    [ Upstream commit ef16799640865f937719f0771c93be5dca18adc6 ]

    A received TKIP key may be up to 32 bytes because it may contain
    MIC rx/tx keys too. These are not used by iwl and copying these
    over overflows the iwl_keyinfo.key field.

    Add a check to not copy more data to iwl_keyinfo.key then will fit.

    This fixes backtraces like this one:

     memcpy: detected field-spanning write (size 32) of single field "sta_cmd.key.key" at
 drivers/net/wireless/intel/iwlwifi/dvm/sta.c:1103 (size 16)
     WARNING: CPU: 1 PID: 946 at drivers/net/wireless/intel/iwlwifi/dvm/sta.c:1103
 iwlagn_send_sta_key+0x375/0x390 [iwldvm]
     <snip>
     Hardware name: Dell Inc. Latitude E6430/0H3MT5, BIOS A21 05/08/2017
     RIP: 0010:iwlagn_send_sta_key+0x375/0x390 [iwldvm]
     <snip>
     Call Trace:
      <TASK>
      iwl_set_dynamic_key+0x1f0/0x220 [iwldvm]
      iwlagn_mac_set_key+0x1e4/0x280 [iwldvm]
      drv_set_key+0xa4/0x1b0 [mac80211]
      ieee80211_key_enable_hw_accel+0xa8/0x2d0 [mac80211]
      ieee80211_key_replace+0x22d/0x8e0 [mac80211]
     <snip>

    Link: https://www.alionet.org/index.php?topic=1469.0
    Link: https://lore.kernel.org/linux-wireless/20230218191056.never.374-kees@kernel.org/
    Link: https://lore.kernel.org/linux-wireless/68760035-7f75-1b23-e355-
 bfb758a87d83@redhat.com/
    Cc: Kees Cook <keescook@chromium.org>
    Suggested-by: Johannes Berg <johannes@sipsolutions.net>
```

```
    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Reviewed-by: Kees Cook <keescook@chromium.org>
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 651260e563d9f50827dac496dc8a0b9b23d5db1a
Author: Chaitanya Kulkarni <kch@nvidia.com>
Date:   Sun Apr 16 15:03:39 2023 -0700

    null_blk: Always check queue mode setting from configfs

    [ Upstream commit 63f8793ee60513a09f110ea460a6ff2c33811cdb ]

    Make sure to check device queue mode in the null_validate_conf() and
    return error for NULL_Q_RQ as we don't allow legacy I/O path, without
    this patch we get OOPs when queue mode is set to 1 from configfs,
    following are repro steps :-

    modprobe null_blk nr_devices=0
    mkdir config/nullb/nullb0
    echo 1 > config/nullb/nullb0/memory_backed
    echo 4096 > config/nullb/nullb0/blocksize
    echo 20480 > config/nullb/nullb0/size
    echo 1 > config/nullb/nullb0/queue_mode
    echo 1 > config/nullb/nullb0/power

    Entering kdb (current=0xffff88810acdd080, pid 2372) on processor 42 Oops: (null)
    due to oops @ 0xffffffffc041c329
    CPU: 42 PID: 2372 Comm: sh Tainted: G           O     N 6.3.0-rc5lblk+ #5
    Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS rel-1.14.0-0-g155821a1990b-
prebuilt.qemu.org 04/01/2014
    RIP: 0010:null_add_dev.part.0+0xd9/0x720 [null_blk]
    Code: 01 00 00 85 d2 0f 85 a1 03 00 00 48 83 bb 08 01 00 00 00 0f 85 f7 03 00 00 80 bb
62 01 00 00 00 48 8b 75 20 0f 85 6d 02 00 00 <48> 89 6e 60 48 8b 75 20 bf 06 00 00 00 e8 f5
37 2c c1 48 8b 75 20
    RSP: 0018:ffffc900052cbde0 EFLAGS: 00010246
    RAX: 0000000000000001 RBX: ffff88811084d800 RCX: 0000000000000001
    RDX: 0000000000000000 RSI: 0000000000000000 RDI: ffff888100042e00
    RBP: ffff8881053d8200 R08: ffffc900052cbd68 R09: ffff888105db2000
    R10: 0000000000000001 R11: 0000000000000000 R12: 0000000000000002
    R13: ffff888104765200 R14: ffff88810eec1748 R15: ffff88810eec1740
    FS:  00007fd445fd1740(0000) GS:ffff8897dfc80000(0000) knlGS:0000000000000000
    CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    CR2: 0000000000000060 CR3: 0000000166a00000 CR4: 0000000000350ee0
    DR0: ffffffff8437a488 DR1: ffffffff8437a489 DR2: ffffffff8437a48a
    DR3: ffffffff8437a48b DR6: 00000000ffff0ff0 DR7: 0000000000000400
    Call Trace:
     <TASK>
     nullb_device_power_store+0xd1/0x120 [null_blk]
     configfs_write_iter+0xb4/0x120
     vfs_write+0x2ba/0x3c0
     ksys_write+0x5f/0xe0
     do_syscall_64+0x3b/0x90
     entry_SYSCALL_64_after_hwframe+0x72/0xdc
    RIP: 0033:0x7fd4460c57a7
    Code: 0d 00 f7 d8 64 89 02 48 c7 c0 ff ff ff ff eb b7 0f 1f 00 f3 0f 1e fa 64 8b 04 25
18 00 00 00 85 c0 75 10 b8 01 00 00 00 0f 05 <48> 3d 00 f0 ff ff 77 51 c3 48 83 ec 28 48 89
54 24 18 48 89 74 24
    RSP: 002b:00007ffd3792a4a8 EFLAGS: 00000246 ORIG_RAX: 0000000000000001
    RAX: ffffffffffffffda RBX: 0000000000000002 RCX: 00007fd4460c57a7
    RDX: 0000000000000002 RSI: 000055b43c02e4c0 RDI: 0000000000000001
    RBP: 000055b43c02e4c0 R08: 000000000000000a R09: 00007fd44615b4e0
    R10: 00007fd44615b3e0 R11: 0000000000000246 R12: 0000000000000002
    R13: 00007fd446198520 R14: 0000000000000002 R15: 00007fd446198700
     </TASK>
```

```
    Signed-off-by: Chaitanya Kulkarni <kch@nvidia.com>
    Reviewed-by: Damien Le Moal <dlemoal@kernel.org>
    Reviewed-by: Ming Lei <ming.lei@redhat.com>
    Reviewed-by: Nitesh Shetty <nj.shetty@samsung.com>
    Link: https://lore.kernel.org/r/20230416220339.43845-1-kch@nvidia.com
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 87cb2bca63388b18e11fa302408cdaca91ac3243
Author: Jaegeuk Kim <jaegeuk@kernel.org>
Date:   Thu Apr 6 16:56:20 2023 -0700

    f2fs: relax sanity check if checkpoint is corrupted

    [ Upstream commit bd90c5cd339a9d7cdc609d2d6310b80dc697070d ]

    1. extent_cache
     - let's drop the largest extent_cache
    2. invalidate_block
     - don't show the warnings

    Reviewed-by: Chao Yu <chao@kernel.org>
    Signed-off-by: Jaegeuk Kim <jaegeuk@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4489aa868bc6343afdaf5ef324af5b1f64962b25
Author: Johannes Berg <johannes.berg@intel.com>
Date:   Mon Apr 17 11:41:30 2023 +0300

    wifi: iwlwifi: fix iwl_mvm_max_amsdu_size() for MLO

    [ Upstream commit b2bc600cced23762d4e97db8989b18772145604f ]

    For MLO, we cannot use vif->bss_conf.chandef.chan->band, since
    that will lead to a NULL-ptr dereference as bss_conf isn't used.
    However, in case of real MLO, we also need to take both LMACs
    into account if they exist, since the station might be active
    on both LMACs at the same time.

    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link:
https://lore.kernel.org/r/20230417113648.3588afc85d79.I11592893bbc191b9548518b8bd782de568a9f
848@changeid
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 68e4b3e711601cf0db6b2bfe6d72cfe2436613de
Author: Reese Russell <git@qrsnap.io>
Date:   Tue Apr 4 00:35:12 2023 -0700

    wifi: mt76: mt7921: add Netgear AXE3000 (A8000) support

    [ Upstream commit 03eb52dd78cab08f13925aeec8315fbdbcba3253 ]

    Issue: Though the Netgear AXE3000 (A8000) is based on the mt7921
    chipset because of the unique USB VID:PID combination this device
    does not initialize/register. Thus making it not plug and play.

    Fix: Adds support for the Netgear AXE3000 (A8000) based on the Mediatek
    mt7921au chipset. The method of action is adding the USD VID/PID
    pair to the mt7921u_device_table[] array.

    Notes: A retail sample of the Netgear AXE3000 (A8000) yeilds the following
```

```
    from lsusb D 0846:9060 NetGear, Inc. Wireless_Device. This pair
    0846:9060 VID:PID has been reported by other users on Github.

    Signed-off-by: Reese Russell <git@qrsnap.io>
    Signed-off-by: Felix Fietkau <nbd@nbd.name>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 41efc47f5bc53e63461579e206adc17c4452ab6e
Author: Harshitha Prem <quic_hprem@quicinc.com>
Date:   Tue Apr 4 00:11:54 2023 +0530

    wifi: ath11k: Ignore frags from uninitialized peer in dp.

    [ Upstream commit a06bfb3c9f69f303692cdae87bc0899d2ae8b2a6 ]

    When max virtual ap interfaces are configured in all the bands with
    ACS and hostapd restart is done every 60s, a crash is observed at
    random times.
    In this certain scenario, a fragmented packet is received for
    self peer, for which rx_tid and rx_frags are not initialized in
    datapath. While handling this fragment, crash is observed as the
    rx_frag list is uninitialised and when we walk in
    ath11k_dp_rx_h_sort_frags, skb null leads to exception.

    To address this, before processing received fragments we check
    dp_setup_done flag is set to ensure that peer has completed its
    dp peer setup for fragment queue, else ignore processing the
    fragments.

    Call trace:
      ath11k_dp_process_rx_err+0x550/0x1084 [ath11k]
      ath11k_dp_service_srng+0x70/0x370 [ath11k]
      0xffffffc009693a04
      __napi_poll+0x30/0xa4
      net_rx_action+0x118/0x270
      __do_softirq+0x10c/0x244
      irq_exit+0x64/0xb4
      __handle_domain_irq+0x88/0xac
      gic_handle_irq+0x74/0xbc
      el1_irq+0xf0/0x1c0
      arch_cpu_idle+0x10/0x18
      do_idle+0x104/0x248
      cpu_startup_entry+0x20/0x64
      rest_init+0xd0/0xdc
      arch_call_rest_init+0xc/0x14
      start_kernel+0x480/0x4b8
      Code: f9400281 f94066a2 91405021 b94a0023 (f9406401)

    Tested-on: IPQ8074 hw2.0 AHB WLAN.HK.2.7.0.1-01744-QCAHKSWPL_SILICONZ-1

    Signed-off-by: Harshitha Prem <quic_hprem@quicinc.com>
    Signed-off-by: Nagarajan Maran <quic_nmaran@quicinc.com>
    Signed-off-by: Kalle Valo <quic_kvalo@quicinc.com>
    Link: https://lore.kernel.org/r/20230403184155.8670-2-quic_nmaran@quicinc.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c0346a59d719461248c6dc6f21c9e55ef836b66f
Author: Colin Ian King <colin.i.king@gmail.com>
Date:   Thu Apr 13 14:30:09 2023 +0100

    block, bfq: Fix division by zero error on zero wsum

    [ Upstream commit e53413f8deedf738a6782cc14cc00bd5852ccf18 ]

    When the weighted sum is zero the calculation of limit causes
```

```
    a division by zero error. Fix this by continuing to the next level.

    This was discovered by running as root:

    stress-ng --ioprio 0

    Fixes divison by error oops:

    [  521.450556] divide error: 0000 [#1] SMP NOPTI
    [  521.450766] CPU: 2 PID: 2684464 Comm: stress-ng-iopri Not tainted 6.2.1-1280.native
#1
    [  521.451117] Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS rel-1.16.1-0-
g3208b098f51a-prebuilt.qemu.org 04/01/2014
    [  521.451627] RIP: 0010:bfqq_request_over_limit+0x207/0x400
    [  521.451875] Code: 01 48 8d 0c c8 74 0b 48 8b 82 98 00 00 00 48 8d 0c c8 8b 85 34 ff
ff ff 48 89 ca 41 0f af 41 50 48 d1 ea 48 98 48 01 d0 31 d2 <48> f7 f1 41 39 41 48 89 85 34
ff ff ff 0f 8c 7b 01 00 00 49 8b 44
    [  521.452699] RSP: 0018:ffffb1af84eb3948 EFLAGS: 00010046
    [  521.452938] RAX: 000000000000003c RBX: 0000000000000000 RCX: 0000000000000000
    [  521.453262] RDX: 0000000000000000 RSI: 0000000000000000 RDI: ffffb1af84eb3978
    [  521.453584] RBP: ffffb1af84eb3a30 R08: 0000000000000001 R09: ffff8f88ab8a4ba0
    [  521.453905] R10: 0000000000000000 R11: 0000000000000001 R12: ffff8f88ab8a4b18
    [  521.454224] R13: ffff8f8699093000 R14: 0000000000000001 R15: ffffb1af84eb3970
    [  521.454549] FS:  00005640b6b0b580(0000) GS:ffff8f88b3880000(0000)
knlGS:0000000000000000
    [  521.454912] CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    [  521.455170] CR2: 00007ffcbcae4e38 CR3: 00000002e46de001 CR4: 0000000000770ee0
    [  521.455491] PKRU: 55555554
    [  521.455619] Call Trace:
    [  521.455736]  <TASK>
    [  521.455837]  ? bfq_request_merge+0x3a/0xc0
    [  521.456027]  ? elv_merge+0x115/0x140
    [  521.456191]  bfq_limit_depth+0xc8/0x240
    [  521.456366]  __blk_mq_alloc_requests+0x21a/0x2c0
    [  521.456577]  blk_mq_submit_bio+0x23c/0x6c0
    [  521.456766]  __submit_bio+0xb8/0x140
    [  521.457236]  submit_bio_noacct_nocheck+0x212/0x300
    [  521.457748]  submit_bio_noacct+0x1a6/0x580
    [  521.458220]  submit_bio+0x43/0x80
    [  521.458660]  ext4_io_submit+0x23/0x80
    [  521.459116]  ext4_do_writepages+0x40a/0xd00
    [  521.459596]  ext4_writepages+0x65/0x100
    [  521.460050]  do_writepages+0xb7/0x1c0
    [  521.460492]  __filemap_fdatawrite_range+0xa6/0x100
    [  521.460979]  file_write_and_wait_range+0xbf/0x140
    [  521.461452]  ext4_sync_file+0x105/0x340
    [  521.461882]  __x64_sys_fsync+0x67/0x100
    [  521.462305]  ? syscall_exit_to_user_mode+0x2c/0x1c0
    [  521.462768]  do_syscall_64+0x3b/0xc0
    [  521.463165]  entry_SYSCALL_64_after_hwframe+0x5a/0xc4
    [  521.463621] RIP: 0033:0x5640b6c56590
    [  521.464006] Code: 00 f7 d8 64 89 01 48 83 c8 ff c3 66 2e 0f 1f 84 00 00 00 00 00 0f
1f 44 00 00 80 3d 71 70 0e 00 00 74 17 b8 4a 00 00 00 0f 05 <48> 3d 00 f0 ff ff 77 48 c3 0f
1f 80 00 00 00 00 48 83 ec 18 89 7c

    Signed-off-by: Colin Ian King <colin.i.king@gmail.com>
    Link: https://lore.kernel.org/r/20230413133009.1605335-1-colin.i.king@gmail.com
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9c0cf1913ea7c56e8e39ce310b9442e2df094fad
Author: Johannes Berg <johannes.berg@intel.com>
Date:   Fri Apr 14 13:12:02 2023 +0300

    wifi: iwlwifi: mvm: fix ptk_pn memory leak
```

```
    [ Upstream commit d066a530af8e1833c7ea2cef7784004700c85f79 ]

    If adding a key to firmware fails we leak the allocated ptk_pn.
    This shouldn't happen in practice, but we should still fix it.

    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link:
https://lore.kernel.org/r/20230414130637.99446ffd02bc.I82a2ad6ec1395f188e0a1677cc619e3fcb1fe
ac9@changeid
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit de78456976026102babe66258c228691ca5677c0
Author: Hyunwoo Kim <imv4bel@gmail.com>
Date:   Fri Apr 14 13:11:59 2023 +0300

    wifi: iwlwifi: pcie: Fix integer overflow in iwl_write_to_user_buf

    [ Upstream commit 58d1b717879bfeabe09b35e41ad667c79933eb2e ]

    An integer overflow occurs in the iwl_write_to_user_buf() function,
    which is called by the iwl_dbgfs_monitor_data_read() function.

    static bool iwl_write_to_user_buf(char __user *user_buf, ssize_t count,
                                      void *buf, ssize_t *size,
                                      ssize_t *bytes_copied)
    {
            int buf_size_left = count - *bytes_copied;

            buf_size_left = buf_size_left - (buf_size_left % sizeof(u32));
            if (*size > buf_size_left)
                    *size = buf_size_left;

    If the user passes a SIZE_MAX value to the "ssize_t count" parameter,
    the ssize_t count parameter is assigned to "int buf_size_left".
    Then compare "*size" with "buf_size_left" . Here, "buf_size_left" is a
    negative number, so "*size" is assigned "buf_size_left" and goes into
    the third argument of the copy_to_user function, causing a heap overflow.

    This is not a security vulnerability because iwl_dbgfs_monitor_data_read()
    is a debugfs operation with 0400 privileges.

    Signed-off-by: Hyunwoo Kim <imv4bel@gmail.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link:
https://lore.kernel.org/r/20230414130637.2d80ace81532.Iecfba549e0e0be21bbb0324675392e42e75bd
5ad@changeid
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6bb40b886a27a005aa2fd15c66b39010d1bb1ee7
Author: Mukesh Sisodiya <mukesh.sisodiya@intel.com>
Date:   Fri Apr 14 13:11:53 2023 +0300

    wifi: iwlwifi: add a new PCI device ID for BZ device

    [ Upstream commit c30a2a64788b3d617a9c5d96adb76c68b0862e5f ]

    Add support for a new PCI device ID 0x272b once registering with PCIe.

    Signed-off-by: Mukesh Sisodiya <mukesh.sisodiya@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link:
```

```
https://lore.kernel.org/r/20230414130637.56342664110d.I5aa6f2858fdcf69fdea4f1a873115a48bd437
64e@changeid
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dcd23aa6cc0ded7950b60ce1badb80b84045c6c0
Author: Daniel Gabay <daniel.gabay@intel.com>
Date:   Thu Apr 13 21:40:32 2023 +0300

    wifi: iwlwifi: pcie: fix possible NULL pointer dereference

    [ Upstream commit b655b9a9f8467684cfa8906713d33b71ea8c8f54 ]

    It is possible that iwl_pci_probe() will fail and free the trans,
    then afterwards iwl_pci_remove() will be called and crash by trying
    to access trans which is already freed, fix it.

    iwlwifi 0000:01:00.0: Detected crf-id 0xa5a5a5a2, cnv-id 0xa5a5a5a2
                          wfpm id 0xa5a5a5a2
    iwlwifi 0000:01:00.0: Can't find a correct rfid for crf id 0x5a2
    ...
    BUG: kernel NULL pointer dereference, address: 0000000000000028
    ...
    RIP: 0010:iwl_pci_remove+0x12/0x30 [iwlwifi]
    pci_device_remove+0x3e/0xb0
    device_release_driver_internal+0x103/0x1f0
    driver_detach+0x4c/0x90
    bus_remove_driver+0x5c/0xd0
    driver_unregister+0x31/0x50
    pci_unregister_driver+0x40/0x90
    iwl_pci_unregister_driver+0x15/0x20 [iwlwifi]
    __exit_compat+0x9/0x98 [iwlwifi]
    __x64_sys_delete_module+0x147/0x260

    Signed-off-by: Daniel Gabay <daniel.gabay@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link:
https://lore.kernel.org/r/20230413213309.082f6e21341b.I0db21d7fa9a828d571ca886713bd0b5d0b6e1
e5c@changeid
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 485ab7fb5eda1ab2653cb29094f2fcf088d4aba4
Author: Vladimir Oltean <vladimir.oltean@nxp.com>
Date:   Tue Apr 11 21:01:53 2023 +0300

    net/sched: pass netlink extack to mqprio and taprio offload

    [ Upstream commit c54876cd5961ce0f8e74807f79a6739cd6b35ddf ]

    With the multiplexed ndo_setup_tc() model which lacks a first-class
    struct netlink_ext_ack * argument, the only way to pass the netlink
    extended ACK message down to the device driver is to embed it within the
    offload structure.

    Do this for struct tc_mqprio_qopt_offload and struct tc_taprio_qopt_offload.

    Since struct tc_taprio_qopt_offload also contains a tc_mqprio_qopt_offload
    structure, and since device drivers might effectively reuse their mqprio
    implementation for the mqprio portion of taprio, we make taprio set the
    extack in both offload structures to point at the same netlink extack
    message.

    In fact, the taprio handling is a bit more tricky, for 2 reasons.
```

```
    First is because the offload structure has a longer lifetime than the
    extack structure. The driver is supposed to populate the extack
    synchronously from ndo_setup_tc() and leave it alone afterwards.
    To not have any use-after-free surprises, we zero out the extack pointer
    when we leave taprio_enable_offload().

    The second reason is because taprio does overwrite the extack message on
    ndo_setup_tc() error. We need to switch to the weak form of setting an
    extack message, which preserves a potential message set by the driver.

    Signed-off-by: Vladimir Oltean <vladimir.oltean@nxp.com>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Acked-by: Jamal Hadi Salim <jhs@mojatatu.com>
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 23309704e90859af2662bedc44101e6d1d2ece7e
Author: Yu Kuai <yukuai3@huawei.com>
Date:   Fri Mar 10 15:38:51 2023 +0800

    md: fix soft lockup in status_resync

    [ Upstream commit 6efddf1e32e2a264694766ca485a4f5e04ee82a7 ]

    status_resync() will calculate 'curr_resync - recovery_active' to show
    user a progress bar like following:

    [============>.......]  resync = 61.4%

    'curr_resync' and 'recovery_active' is updated in md_do_sync(), and
    status_resync() can read them concurrently, hence it's possible that
    'curr_resync - recovery_active' can overflow to a huge number. In this
    case status_resync() will be stuck in the loop to print a large amount
    of '=', which will end up soft lockup.

    Fix the problem by setting 'resync' to MD_RESYNC_ACTIVE in this case,
    this way resync in progress will be reported to user.

    Signed-off-by: Yu Kuai <yukuai3@huawei.com>
    Signed-off-by: Song Liu <song@kernel.org>
    Link: https://lore.kernel.org/r/20230310073855.1337560-3-yukuai1@huaweicloud.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b9168d41b83d182f34ba927ee822edaee18d5fc8
Author: Yafang <laoar.shao@gmail.com>
Date:   Thu Apr 13 02:52:48 2023 +0000

    bpf: Add preempt_count_{sub,add} into btf id deny list

    [ Upstream commit c11bd046485d7bf1ca200db0e7d0bdc4bafdd395 ]

    The recursion check in __bpf_prog_enter* and __bpf_prog_exit*
    leave preempt_count_{sub,add} unprotected. When attaching trampoline to
    them we get panic as follows,

    [  867.843050] BUG: TASK stack guard page was hit at 0000000009d325cf (stack is
0000000046a46a15..00000000537e7b28)
    [  867.843064] stack guard page: 0000 [#1] PREEMPT SMP NOPTI
    [  867.843067] CPU: 8 PID: 11009 Comm: trace Kdump: loaded Not tainted 6.2.0+ #4
    [  867.843100] Call Trace:
    [  867.843101]  <TASK>
    [  867.843104]  asm_exc_int3+0x3a/0x40
    [  867.843108] RIP: 0010:preempt_count_sub+0x1/0xa0
    [  867.843135]  __bpf_prog_enter_recur+0x17/0x90
    [  867.843148]  bpf_trampoline_6442468108_0+0x2e/0x1000
```

```
     [  867.843154]  ? preempt_count_sub+0x1/0xa0
     [  867.843157]  preempt_count_sub+0x5/0xa0
     [  867.843159]  ? migrate_enable+0xac/0xf0
     [  867.843164]  __bpf_prog_exit_recur+0x2d/0x40
     [  867.843168]  bpf_trampoline_6442468108_0+0x55/0x1000
     ...
     [  867.843788]  preempt_count_sub+0x5/0xa0
     [  867.843793]  ? migrate_enable+0xac/0xf0
     [  867.843829]  __bpf_prog_exit_recur+0x2d/0x40
     [  867.843837] BUG: IRQ stack guard page was hit at 0000000099bd8228 (stack is
  00000000b23e2bc4..000000006d95af35)
     [  867.843841] BUG: IRQ stack guard page was hit at 000000005ae07924 (stack is
  00000000ffd69623..0000000014eb594c)
     [  867.843843] BUG: IRQ stack guard page was hit at 00000000028320f0 (stack is
  00000000034b6438..0000000078d1bcec)
     [  867.843842]  bpf_trampoline_6442468108_0+0x55/0x1000
     ...

     That is because in __bpf_prog_exit_recur, the preempt_count_{sub,add} are
     called after prog->active is decreased.

     Fixing this by adding these two functions into btf ids deny list.

     Suggested-by: Steven Rostedt <rostedt@goodmis.org>
     Signed-off-by: Yafang <laoar.shao@gmail.com>
     Cc: Masami Hiramatsu <mhiramat@kernel.org>
     Cc: Steven Rostedt <rostedt@goodmis.org>
     Cc: Jiri Olsa <olsajiri@gmail.com>
     Acked-by: Hao Luo <haoluo@google.com>
     Link: https://lore.kernel.org/r/20230413025248.79764-1-laoar.shao@gmail.com
     Signed-off-by: Alexei Starovoitov <ast@kernel.org>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit edf37bc8b03d3f948e679b2fd2d14464495f5d1b
  Author: Hao Zeng <zenghao@kylinos.cn>
  Date:   Tue Apr 11 16:43:49 2023 +0800

     samples/bpf: Fix fout leak in hbm's run_bpf_prog

     [ Upstream commit 23acb14af1914010dd0aae1bbb7fab28bf518b8e ]

     Fix fout being fopen'ed but then not subsequently fclose'd. In the affected
     branch, fout is otherwise going out of scope.

     Signed-off-by: Hao Zeng <zenghao@kylinos.cn>
     Signed-off-by: Daniel Borkmann <daniel@iogearbox.net>
     Link: https://lore.kernel.org/bpf/20230411084349.1999628-1-zenghao@kylinos.cn
     Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit da8c535b28696017e5d1532d12ea78e836432d9e
  Author: Chao Yu <chao@kernel.org>
  Date:   Tue Apr 4 23:28:07 2023 +0800

     f2fs: fix to check readonly condition correctly

     [ Upstream commit d78dfefcde9d311284434560d69c0478c55a657e ]

     With below case, it can mount multi-device image w/ rw option, however
     one of secondary device is set as ro, later update will cause panic, so
     let's introduce f2fs_dev_is_readonly(), and check multi-devices rw status
     in f2fs_remount() w/ it in order to avoid such inconsistent mount status.

     mkfs.f2fs -c /dev/zram1 /dev/zram0 -f
     blockdev --setro /dev/zram1
     mount -t f2fs dev/zram0 /mnt/f2fs
```

```
    mount: /mnt/f2fs: WARNING: source write-protected, mounted read-only.
    mount -t f2fs -o remount,rw mnt/f2fs
    dd if=/dev/zero  of=/mnt/f2fs/file bs=1M count=8192

    kernel BUG at fs/f2fs/inline.c:258!
    RIP: 0010:f2fs_write_inline_data+0x23e/0x2d0 [f2fs]
    Call Trace:
      f2fs_write_single_data_page+0x26b/0x9f0 [f2fs]
      f2fs_write_cache_pages+0x389/0xa60 [f2fs]
      __f2fs_write_data_pages+0x26b/0x2d0 [f2fs]
      f2fs_write_data_pages+0x2e/0x40 [f2fs]
      do_writepages+0xd3/0x1b0
      __writeback_single_inode+0x5b/0x420
      writeback_sb_inodes+0x236/0x5a0
      __writeback_inodes_wb+0x56/0xf0
      wb_writeback+0x2a3/0x490
      wb_do_writeback+0x2b2/0x330
      wb_workfn+0x6a/0x260
      process_one_work+0x270/0x5e0
      worker_thread+0x52/0x3e0
      kthread+0xf4/0x120
      ret_from_fork+0x29/0x50

    Signed-off-by: Chao Yu <chao@kernel.org>
    Signed-off-by: Jaegeuk Kim <jaegeuk@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 82c3d6e9db41cbd3af1d4f90bdb441740b5fad10
Author: Chao Yu <chao@kernel.org>
Date:   Mon Apr 10 10:12:22 2023 +0800

    f2fs: fix to drop all dirty pages during umount() if cp_error is set

    [ Upstream commit c9b3649a934d131151111354bcbb638076f03a30 ]

    xfstest generic/361 reports a bug as below:

    f2fs_bug_on(sbi, sbi->fsync_node_num);

    kernel BUG at fs/f2fs/super.c:1627!
    RIP: 0010:f2fs_put_super+0x3a8/0x3b0
    Call Trace:
     generic_shutdown_super+0x8c/0x1b0
     kill_block_super+0x2b/0x60
     kill_f2fs_super+0x87/0x110
     deactivate_locked_super+0x39/0x80
     deactivate_super+0x46/0x50
     cleanup_mnt+0x109/0x170
     __cleanup_mnt+0x16/0x20
     task_work_run+0x65/0xa0
     exit_to_user_mode_prepare+0x175/0x190
     syscall_exit_to_user_mode+0x25/0x50
     do_syscall_64+0x4c/0x90
     entry_SYSCALL_64_after_hwframe+0x72/0xdc

    During umount(), if cp_error is set, f2fs_wait_on_all_pages() should
    not stop waiting all F2FS_WB_CP_DATA pages to be writebacked, otherwise,
    fsync_node_num can be non-zero after f2fs_wait_on_all_pages() causing
    this bug.

    In this case, to avoid deadloop in f2fs_wait_on_all_pages(), it needs
    to drop all dirty pages rather than redirtying them.

    Signed-off-by: Chao Yu <chao@kernel.org>
    Signed-off-by: Jaegeuk Kim <jaegeuk@kernel.org>
```

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

commit ce71c61d661cfac3f097af928995abfcebd2b8c5
Author: Yonggil Song <yonggil.song@samsung.com>
Date:   Tue Mar 21 09:12:51 2023 +0900

    f2fs: Fix system crash due to lack of free space in LFS

    [ Upstream commit d11cef14f8146f3babd286c2cc8ca09c166295e2 ]

    When f2fs tries to checkpoint during foreground gc in LFS mode, system
    crash occurs due to lack of free space if the amount of dirty node and
    dentry pages generated by data migration exceeds free space.
    The reproduction sequence is as follows.

     - 20GiB capacity block device (null_blk)
     - format and mount with LFS mode
     - create a file and write 20,000MiB
     - 4k random write on full range of the file

     RIP: 0010:new_curseg+0x48a/0x510 [f2fs]
     Code: 55 e7 f5 89 c0 48 0f af c3 48 8b 5d c0 48 c1 e8 20 83 c0 01 89 43 6c 48 83 c4 28
5b 41 5c 41 5d 41 5e 41 5f 5d c3 cc cc cc cc <0f> 0b f0 41 80 4f 48 04 45 85 f6 0f 84 ba fd
ff ff e9 ef fe ff ff
     RSP: 0018:ffff977bc397b218 EFLAGS: 00010246
     RAX: 00000000000027b9 RBX: 0000000000000000 RCX: 00000000000027c0
     RDX: 0000000000000000 RSI: 00000000000027b9 RDI: ffff8c25ab4e74f8
     RBP: ffff977bc397b268 R08: 00000000000027b9 R09: ffff8c29e4a34b40
     R10: 0000000000000001 R11: ffff977bc397b0d8 R12: 0000000000000000
     R13: ffff8c25b4dd81a0 R14: 0000000000000000 R15: ffff8c2f667f9000
     FS: 0000000000000000(0000) GS:ffff8c344ec80000(0000) knlGS:0000000000000000
     CS: 0010 DS: 0000 ES: 0000 CR0: 0000000080050033
     CR2: 000000c00055d000 CR3: 0000000e30810003 CR4: 00000000003706e0
     DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000
     DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400
     Call Trace:
     <TASK>
     allocate_segment_by_default+0x9c/0x110 [f2fs]
     f2fs_allocate_data_block+0x243/0xa30 [f2fs]
     ? __mod_lruvec_page_state+0xa0/0x150
     do_write_page+0x80/0x160 [f2fs]
     f2fs_do_write_node_page+0x32/0x50 [f2fs]
     __write_node_page+0x339/0x730 [f2fs]
     f2fs_sync_node_pages+0x5a6/0x780 [f2fs]
     block_operations+0x257/0x340 [f2fs]
     f2fs_write_checkpoint+0x102/0x1050 [f2fs]
     f2fs_gc+0x27c/0x630 [f2fs]
     ? folio_mark_dirty+0x36/0x70
     f2fs_balance_fs+0x16f/0x180 [f2fs]

    This patch adds checking whether free sections are enough before checkpoint
    during gc.

    Signed-off-by: Yonggil Song <yonggil.song@samsung.com>
    [Jaegeuk Kim: code clean-up]
    Reviewed-by: Chao Yu <chao@kernel.org>
    Signed-off-by: Jaegeuk Kim <jaegeuk@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 35b0b59743a254fe46c2c3928cb429be199fb3ed
Author: Stephan Müller <smueller@chronox.de>
Date:   Mon Mar 27 09:03:52 2023 +0200

    crypto: jitter - permanent and intermittent health errors
```

```
    [ Upstream commit 3fde2fe99aa6dacd4151c87382b07ce7f30f0a52 ]

    According to SP800-90B, two health failures are allowed: the intermittend
    and the permanent failure. So far, only the intermittent failure was
    implemented. The permanent failure was achieved by resetting the entire
    entropy source including its health test state and waiting for two or
    more back-to-back health errors.

    This approach is appropriate for RCT, but not for APT as APT has a
    non-linear cutoff value. Thus, this patch implements 2 cutoff values
    for both RCT/APT. This implies that the health state is left untouched
    when an intermittent failure occurs. The noise source is reset
    and a new APT powerup-self test is performed. Yet, whith the unchanged
    health test state, the counting of failures continues until a permanent
    failure is reached.

    Any non-failing raw entropy value causes the health tests to reset.

    The intermittent error has an unchanged significance level of 2^-30.
    The permanent error has a significance level of 2^-60. Considering that
    this level also indicates a false-positive rate (see SP800-90B section 4.2)
    a false-positive must only be incurred with a low probability when
    considering a fleet of Linux kernels as a whole. Hitting the permanent
    error may cause a panic(), the following calculation applies: Assuming
    that a fleet of 10^9 Linux kernels run concurrently with this patch in
    FIPS mode and on each kernel 2 health tests are performed every minute
    for one year, the chances of a false positive is about 1:1000
    based on the binomial distribution.

    In addition, any power-up health test errors triggered with
    jent_entropy_init are treated as permanent errors.

    A permanent failure causes the entire entropy source to permanently
    return an error. This implies that a caller can only remedy the situation
    by re-allocating a new instance of the Jitter RNG. In a subsequent
    patch, a transparent re-allocation will be provided which also changes
    the implied heuristic entropy assessment.

    In addition, when the kernel is booted with fips=1, the Jitter RNG
    is defined to be part of a FIPS module. The permanent error of the
    Jitter RNG is translated as a FIPS module error. In this case, the entire
    FIPS module must cease operation. This is implemented in the kernel by
    invoking panic().

    The patch also fixes an off-by-one in the RCT cutoff value which is now
    set to 30 instead of 31. This is because the counting of the values
    starts with 0.

    Reviewed-by: Vladis Dronov <vdronov@redhat.com>
    Signed-off-by: Stephan Mueller <smueller@chronox.de>
    Reviewed-by: Marcelo Henrique Cerri <marcelo.cerri@canonical.com>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cec4ef62b36b04e0bc8905732adab091f4bc1cfd
Author: Ojaswin Mujoo <ojaswin@linux.ibm.com>
Date:   Sat Mar 25 13:43:39 2023 +0530

    ext4: Fix best extent lstart adjustment logic in ext4_mb_new_inode_pa()

    [ Upstream commit 93cdf49f6eca5e23f6546b8f28457b2e6a6961d9 ]

    When the length of best extent found is less than the length of goal extent
    we need to make sure that the best extent atleast covers the start of the
    original request. This is done by adjusting the ac_b_ex.fe_logical (logical
```

```
    start) of the extent.

    While doing so, the current logic sometimes results in the best extent's
    logical range overflowing the goal extent. Since this best extent is later
    added to the inode preallocation list, we have a possibility of introducing
    overlapping preallocations. This is discussed in detail here [1].

    As per Jan's suggestion, to fix this, replace the existing logic with the
    below logic for adjusting best extent as it keeps fragmentation in check
    while ensuring logical range of best extent doesn't overflow out of goal
    extent:

    1. Check if best extent can be kept at end of goal range and still cover
       original start.
    2. Else, check if best extent can be kept at start of goal range and still
       cover original start.
    3. Else, keep the best extent at start of original request.

    Also, add a few extra BUG_ONs that might help catch errors faster.

    [1] https://lore.kernel.org/r/Y+OGkVvzPN0RMv0O@li-bb2b2a4c-3307-11b2-a85c-
8fa5c3a69313.ibm.com

    Suggested-by: Jan Kara <jack@suse.cz>
    Signed-off-by: Ojaswin Mujoo <ojaswin@linux.ibm.com>
    Reviewed-by: Ritesh Harjani (IBM) <ritesh.list@gmail.com>
    Reviewed-by: Jan Kara <jack@suse.cz>
    Link:
https://lore.kernel.org/r/f96aca6d415b36d1f90db86c1a8cd7e2e9d7ab0e.1679731817.git.ojaswin@li
nux.ibm.com
    Signed-off-by: Theodore Ts'o <tytso@mit.edu>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit abb330ffaa3a0ae7ce632e28c9260b461c01f19f
Author: Kemeng Shi <shikemeng@huaweicloud.com>
Date:   Sat Mar 4 01:21:01 2023 +0800

    ext4: set goal start correctly in ext4_mb_normalize_request

    [ Upstream commit b07ffe6927c75d99af534d685282ea188d9f71a6 ]

    We need to set ac_g_ex to notify the goal start used in
    ext4_mb_find_by_goal. Set ac_g_ex instead of ac_f_ex in
    ext4_mb_normalize_request.
    Besides we should assure goal start is in range [first_data_block,
    blocks_count) as ext4_mb_initialize_context does.

    [ Added a check to make sure size is less than ar->pright; otherwise
      we could end up passing an underflowed value of ar->pright - size to
      ext4_get_group_no_and_offset(), which will trigger a BUG_ON later on.
      - TYT ]

    Signed-off-by: Kemeng Shi <shikemeng@huaweicloud.com>
    Reviewed-by: Ritesh Harjani (IBM) <ritesh.list@gmail.com>
    Link: https://lore.kernel.org/r/20230303172120.3800725-2-shikemeng@huaweicloud.com
    Signed-off-by: Theodore Ts'o <tytso@mit.edu>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6e2a40b3a332ea84079983be21c944de8ddbc4f3
Author: Xingui Yang <yangxingui@huawei.com>
Date:   Mon Mar 20 11:34:22 2023 +0800

    scsi: hisi_sas: Grab sas_dev lock when traversing the members of sas_dev.list

    [ Upstream commit 71fb36b5ff113a7674710b9d6063241eada84ff7 ]
```

```
    When freeing slots in function slot_complete_v3_hw(), it is possible that
    sas_dev.list is being traversed elsewhere, and it may trigger a NULL
    pointer exception, such as follows:

    ==>cq thread                    ==>scsi_eh_6

                                    ==>scsi_error_handler()
                                      ==>sas_eh_handle_sas_errors()
                                        ==>sas_scsi_find_task()
                                          ==>lldd_abort_task()
    ==>slot_complete_v3_hw()              ==>hisi_sas_abort_task()
      ==>hisi_sas_slot_task_free()          ==>dereg_device_v3_hw()
        ==>list_del_init()                    ==>list_for_each_entry_safe()

    [ 7165.434918] sas: Enter sas_scsi_recover_host busy: 32 failed: 32
    [ 7165.434926] sas: trying to find task 0x00000000769b5ba5
    [ 7165.434927] sas: sas_scsi_find_task: aborting task 0x00000000769b5ba5
    [ 7165.434940] hisi_sas_v3_hw 0000:b4:02.0: slot complete: task(00000000769b5ba5)
 aborted
    [ 7165.434964] hisi_sas_v3_hw 0000:b4:02.0: slot complete: task(00000000c9f7aa07)
 ignored
    [ 7165.434965] hisi_sas_v3_hw 0000:b4:02.0: slot complete: task(00000000e2a1cf01)
 ignored
    [ 7165.434968] Unable to handle kernel NULL pointer dereference at virtual address
 0000000000000000
    [ 7165.434972] hisi_sas_v3_hw 0000:b4:02.0: slot complete: task(0000000022d52d93)
 ignored
    [ 7165.434975] hisi_sas_v3_hw 0000:b4:02.0: slot complete: task(0000000066a7516c)
 ignored
    [ 7165.434976] Mem abort info:
    [ 7165.434982]   ESR = 0x96000004
    [ 7165.434991]   Exception class = DABT (current EL), IL = 32 bits
    [ 7165.434992]   SET = 0, FnV = 0
    [ 7165.434993]   EA = 0, S1PTW = 0
    [ 7165.434994] Data abort info:
    [ 7165.434994]   ISV = 0, ISS = 0x00000004
    [ 7165.434995]   CM = 0, WnR = 0
    [ 7165.434997] user pgtable: 4k pages, 48-bit VAs, pgdp = 00000000f29543f2
    [ 7165.434998] [0000000000000000] pgd=0000000000000000
    [ 7165.435003] Internal error: Oops: 96000004 [#1] SMP
    [ 7165.439863] Process scsi_eh_6 (pid: 4109, stack limit = 0x00000000c43818d5)
    [ 7165.468862] pstate: 00c00009 (nzcv daif +PAN +UAO)
    [ 7165.473637] pc : dereg_device_v3_hw+0x68/0xa8 [hisi_sas_v3_hw]
    [ 7165.479443] lr : dereg_device_v3_hw+0x2c/0xa8 [hisi_sas_v3_hw]
    [ 7165.485247] sp : ffff00001d623bc0
    [ 7165.488546] x29: ffff00001d623bc0 x28: ffffa027d03b9508
    [ 7165.493835] x27: ffff80278ed50af0 x26: ffffa027dd31e0a8
    [ 7165.499123] x25: ffffa027d9b27f88 x24: ffffa027d9b209f8
    [ 7165.504411] x23: ffffa027c45b0d60 x22: ffff80278ec07c00
    [ 7165.509700] x21: 0000000000000008 x20: ffffa027d9b209f8
    [ 7165.514988] x19: ffffa027d9b27f88 x18: ffffffffffffffff
    [ 7165.520276] x17: 0000000000000000 x16: 0000000000000000
    [ 7165.525564] x15: ffff0000091d9708 x14: ffff0000093b7dc8
    [ 7165.530852] x13: ffff0000093b7a23 x12: 6e7265746e692067
    [ 7165.536140] x11: 0000000000000000 x10: 0000000000000bb0
    [ 7165.541429] x9 : ffff00001d6238f0 x8 : ffffa027d877af00
    [ 7165.546718] x7 : ffffa027d6329600 x6 : ffff7e809f58ca00
    [ 7165.552006] x5 : 0000000000001f8a x4 : 000000000000088e
    [ 7165.557295] x3 : ffffa027d9b27fa8 x2 : 0000000000000000
    [ 7165.562583] x1 : 0000000000000000 x0 : 000000003000188e
    [ 7165.567872] Call trace:
    [ 7165.570309]  dereg_device_v3_hw+0x68/0xa8 [hisi_sas_v3_hw]
    [ 7165.575775]  hisi_sas_abort_task+0x248/0x358 [hisi_sas_main]
    [ 7165.581415]  sas_eh_handle_sas_errors+0x258/0x8e0 [libsas]
```

```
     [ 7165.586876]  sas_scsi_recover_host+0x134/0x458 [libsas]
     [ 7165.592082]  scsi_error_handler+0xb4/0x488
     [ 7165.596163]  kthread+0x134/0x138
     [ 7165.599380]  ret_from_fork+0x10/0x18
     [ 7165.602940] Code: d5033e9f b9000040 aa0103e2 eb03003f (f9400021)
     [ 7165.609004] kernel fault(0x1) notification starting on CPU 75
     [ 7165.700728] ---[ end trace fc042cbbea224efc ]---
     [ 7165.705326] Kernel panic - not syncing: Fatal exception

     To fix the issue, grab sas_dev lock when traversing the members of
     sas_dev.list in dereg_device_v3_hw() and hisi_sas_release_tasks() to avoid
     concurrency of adding and deleting member. When function
     hisi_sas_release_tasks() calls hisi_sas_do_release_task() to free slot, the
     lock cannot be grabbed again in hisi_sas_slot_task_free(), then a bool
     parameter need_lock is added.

     Signed-off-by: Xingui Yang <yangxingui@huawei.com>
     Signed-off-by: Xiang Chen <chenxiang66@hisilicon.com>
     Link: https://lore.kernel.org/r/1679283265-115066-2-git-send-email-
  chenxiang66@hisilicon.com
     Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 269d858757778fab335a7ed027f858f9f2f340a9
  Author: Adrian Hunter <adrian.hunter@intel.com>
  Date:   Tue Mar 28 13:58:32 2023 +0300

     scsi: ufs: ufs-pci: Add support for Intel Lunar Lake

     [ Upstream commit 0a07d3c7a1d205b47d9f3608ff4e9d1065d63b6d ]

     Add PCI ID to support Intel Lunar Lake, same as MTL.

     Signed-off-by: Adrian Hunter <adrian.hunter@intel.com>
     Link: https://lore.kernel.org/r/20230328105832.3495-1-adrian.hunter@intel.com
     Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit e2daa5de649cc31dc300790d2703f0fef8508a06
  Author: Andreas Gruenbacher <agruenba@redhat.com>
  Date:   Tue Mar 28 00:43:16 2023 +0200

     gfs2: Fix inode height consistency check

     [ Upstream commit cfcdb5bad34f600aed7613c3c1a5e618111f77b7 ]

     The maximum allowed height of an inode's metadata tree depends on the
     filesystem block size; it is lower for bigger-block filesystems.  When
     reading in an inode, make sure that the height doesn't exceed the
     maximum allowed height.

     Arrays like sd_heightsize are sized to be big enough for any filesystem
     block size; they will often be slightly bigger than what's needed for a
     specific filesystem.

     Reported-by: syzbot+45d4691b1ed3c48eba05@syzkaller.appspotmail.com
     Signed-off-by: Andreas Gruenbacher <agruenba@redhat.com>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 48daa4a3015d859ee424948844ce3c12f2fe44e6
  Author: Zheng Wang <zyytlz.wz@163.com>
  Date:   Sat Mar 18 16:16:35 2023 +0800

     scsi: message: mptlan: Fix use after free bug in mptlan_remove() due to race condition
```

```
    [ Upstream commit f486893288f3e9b171b836f43853a6426515d800 ]

    mptlan_probe() calls mpt_register_lan_device() which initializes the
    &priv->post_buckets_task workqueue. A call to
    mpt_lan_wake_post_buckets_task() will subsequently start the work.

    During driver unload in mptlan_remove() the following race may occur:

    CPU0                  CPU1

                         |mpt_lan_post_receive_buckets_work()
    mptlan_remove()      |
      free_netdev()      |
        kfree(dev);      |
                         |
                         | dev->mtu
                         |   //use

    Fix this by finishing the work prior to cleaning up in mptlan_remove().

    [mkp: we really should remove mptlan instead of attempting to fix it]

    Signed-off-by: Zheng Wang <zyytlz.wz@163.com>
    Link: https://lore.kernel.org/r/20230318081635.796479-1-zyytlz.wz@163.com
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 67bca5f1d644f4e79b694abd8052a177de81c37f
Author: Eli Cohen <elic@nvidia.com>
Date:   Wed Feb 8 07:51:02 2023 +0200

    lib: cpu_rmap: Avoid use after free on rmap->obj array entries

    [ Upstream commit 4e0473f1060aa49621d40a113afde24818101d37 ]

    When calling irq_set_affinity_notifier() with NULL at the notify
    argument, it will cause freeing of the glue pointer in the
    corresponding array entry but will leave the pointer in the array. A
    subsequent call to free_irq_cpu_rmap() will try to free this entry again
    leading to possible use after free.

    Fix that by setting NULL to the array entry and checking that we have
    non-zero at the array entry when iterating over the array in
    free_irq_cpu_rmap().

    The current code does not suffer from this since there are no cases
    where irq_set_affinity_notifier(irq, NULL) (note the NULL passed for the
    notify arg) is called, followed by a call to free_irq_cpu_rmap() so we
    don't hit and issue. Subsequent patches in this series excersize this
    flow, hence the required fix.

    Cc: Thomas Gleixner <tglx@linutronix.de>
    Signed-off-by: Eli Cohen <elic@nvidia.com>
    Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
    Reviewed-by: Jacob Keller <jacob.e.keller@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4ce221d295f53e6c6b835ab33181e735482c9aac
Author: Dmitry Bogdanov <d.bogdanov@yadro.com>
Date:   Sat Mar 18 20:56:17 2023 -0500

    scsi: target: iscsit: Free cmds before session free

    [ Upstream commit d8990b5a4d065f38f35d69bcd627ec5a7f8330ca ]
```

```
    Commands from recovery entries are freed after session has been closed.
    That leads to use-after-free at command free or NPE with such call trace:

    Time2Retain timer expired for SID: 1, cleaning up iSCSI session.
    BUG: kernel NULL pointer dereference, address: 0000000000000140
    RIP: 0010:sbitmap_queue_clear+0x3a/0xa0
    Call Trace:
     target_release_cmd_kref+0xd1/0x1f0 [target_core_mod]
     transport_generic_free_cmd+0xd1/0x180 [target_core_mod]
     iscsit_free_cmd+0x53/0xd0 [iscsi_target_mod]
     iscsit_free_connection_recovery_entries+0x29d/0x320 [iscsi_target_mod]
     iscsit_close_session+0x13a/0x140 [iscsi_target_mod]
     iscsit_check_post_dataout+0x440/0x440 [iscsi_target_mod]
     call_timer_fn+0x24/0x140

    Move cleanup of recovery enrties to before session freeing.

    Reported-by: Forza <forza@tnonline.net>
    Signed-off-by: Dmitry Bogdanov <d.bogdanov@yadro.com>
    Signed-off-by: Mike Christie <michael.christie@oracle.com>
    Link: https://lore.kernel.org/r/20230319015620.96006-7-michael.christie@oracle.com
    Reviewed-by: Maurizio Lombardi <mlombard@redhat.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit aa1d4c2866eabb7fc9ef2322124816725ce5e860
Author: Nick Child <nnac123@linux.ibm.com>
Date:   Tue Mar 21 10:07:25 2023 -0500

    netdev: Enforce index cap in netdev_get_tx_queue

    [ Upstream commit 1cc6571f562774f1d928dc8b3cff50829b86e970 ]

    When requesting a TX queue at a given index, warn on out-of-bounds
    referencing if the index is greater than the allocated number of
    queues.

    Specifically, since this function is used heavily in the networking
    stack use DEBUG_NET_WARN_ON_ONCE to avoid executing a new branch on
    every packet.

    Signed-off-by: Nick Child <nnac123@linux.ibm.com>
    Link: https://lore.kernel.org/r/20230321150725.127229-2-nnac123@linux.ibm.com
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5ceb794794e25046b27e26820f505f31e253b0e4
Author: Nick Child <nnac123@linux.ibm.com>
Date:   Tue Mar 21 10:07:24 2023 -0500

    net: Catch invalid index in XPS mapping

    [ Upstream commit 5dd0dfd55baec0742ba8f5625a0dd064aca7db16 ]

    When setting the XPS value of a TX queue, warn the user once if the
    index of the queue is greater than the number of allocated TX queues.

    Previously, this scenario went uncaught. In the best case, it resulted
    in unnecessary allocations. In the worst case, it resulted in
    out-of-bounds memory references through calls to `netdev_get_tx_queue(
    dev, index)`. Therefore, it is important to inform the user but not
    worth returning an error and risk downing the netdevice.

    Signed-off-by: Nick Child <nnac123@linux.ibm.com>
    Reviewed-by: Piotr Raczynski <piotr.raczynski@intel.com>
```

```
    Link: https://lore.kernel.org/r/20230321150725.127229-1-nnac123@linux.ibm.com
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a87f59041a7f77b4bdab05cea60ac6adc69dc5d2
Author: Rajat Soni <quic_rajson@quicinc.com>
Date:   Wed Mar 15 14:36:32 2023 +0530

    wifi: ath12k: fix memory leak in ath12k_qmi_driver_event_work()

    [ Upstream commit 960412bee0ea75f6b3c2dca4a3535795ee84c47a ]

    Currently the buffer pointed by event is not freed in case
    ATH12K_FLAG_UNREGISTERING bit is set, this causes memory leak.

    Add a goto skip instead of return, to ensure event and all the
    list entries are freed properly.

    Tested-on: QCN9274 hw2.0 PCI WLAN.WBE.1.0.1-00029-QCAHKSWPL_SILICONZ-1

    Signed-off-by: Rajat Soni <quic_rajson@quicinc.com>
    Signed-off-by: Kalle Valo <quic_kvalo@quicinc.com>
    Link: https://lore.kernel.org/r/20230315090632.15065-1-quic_rajson@quicinc.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit bfa732380888b21b40ef902a54638038f5697385
Author: Nathan Chancellor <nathan@kernel.org>
Date:   Sun Mar 19 16:41:08 2023 -0700

    net: pasemi: Fix return type of pasemi_mac_start_tx()

    [ Upstream commit c8384d4a51e7cb0e6587f3143f29099f202c5de1 ]

    With clang's kernel control flow integrity (kCFI, CONFIG_CFI_CLANG),
    indirect call targets are validated against the expected function
    pointer prototype to make sure the call target is valid to help mitigate
    ROP attacks. If they are not identical, there is a failure at run time,
    which manifests as either a kernel panic or thread getting killed. A
    warning in clang aims to catch these at compile time, which reveals:

      drivers/net/ethernet/pasemi/pasemi_mac.c:1665:21: error: incompatible function pointer
types initializing 'netdev_tx_t (*)(struct sk_buff *, struct net_device *)' (aka 'enum
netdev_tx (*)(struct sk_buff *, struct net_device *)') with an expression of type 'int
(struct sk_buff *, struct net_device *)' [-Werror,-Wincompatible-function-pointer-types-
strict]
              .ndo_start_xmit         = pasemi_mac_start_tx,
                                        ^~~~~~~~~~~~~~~~~~
      1 error generated.

    ->ndo_start_xmit() in 'struct net_device_ops' expects a return type of
    'netdev_tx_t', not 'int'. Adjust the return type of
    pasemi_mac_start_tx() to match the prototype's to resolve the warning.
    While PowerPC does not currently implement support for kCFI, it could in
    the future, which means this warning becomes a fatal CFI failure at run
    time.

    Link: https://github.com/ClangBuiltLinux/linux/issues/1750
    Signed-off-by: Nathan Chancellor <nathan@kernel.org>
    Reviewed-by: Horatiu Vultur <horatiu.vultur@microchip.com>
    Link: https://lore.kernel.org/r/20230319-pasemi-incompatible-pointer-types-strict-v1-1-
1b9459d8aef0@kernel.org
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 17e0453a7523ad7a25bb47af941b150a6c66d7b6
```

```
Author: Maxim Korotkov <korotkov.maxim.s@gmail.com>
Date:   Thu Mar 9 20:43:47 2023 +0300

    bnxt: avoid overflow in bnxt_get_nvram_directory()

    [ Upstream commit 7c6dddc239abe660598c49ec95ea0ed6399a4b2a ]

    The value of an arithmetic expression is subject
    of possible overflow due to a failure to cast operands to a larger data
    type before performing arithmetic. Used macro for multiplication instead
    operator for avoiding overflow.

    Found by Security Code and Linux Verification
    Center (linuxtesting.org) with SVACE.

    Signed-off-by: Maxim Korotkov <korotkov.maxim.s@gmail.com>
    Reviewed-by: Pavan Chebbi <pavan.chebbi@broadcom.com>
    Link: https://lore.kernel.org/r/20230309174347.3515-1-korotkov.maxim.s@gmail.com
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6cc92379b80af005e1f49ef6ef790cddc58cf0da
Author: Dongliang Mu <dzm91@hust.edu.cn>
Date:   Thu Mar 9 10:16:36 2023 +0800

    wifi: rtw88: fix memory leak in rtw_usb_probe()

    [ Upstream commit 48181d285623198c33bb9698992502687b258efa ]

    drivers/net/wireless/realtek/rtw88/usb.c:876 rtw_usb_probe()
    warn: 'hw' from ieee80211_alloc_hw() not released on lines: 811

    Fix this by modifying return to a goto statement.

    Signed-off-by: Dongliang Mu <dzm91@hust.edu.cn>
    Reviewed-by: Ping-Ke Shih <pkshih@realtek.com>
    Signed-off-by: Kalle Valo <kvalo@kernel.org>
    Link: https://lore.kernel.org/r/20230309021636.528601-1-dzm91@hust.edu.cn
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 872902490afc4f29a3a307410c5f66ab31e3c4b6
Author: Justin Tee <justin.tee@broadcom.com>
Date:   Wed Mar 1 15:16:22 2023 -0800

    scsi: lpfc: Correct used_rpi count when devloss tmo fires with no recovery

    [ Upstream commit db651ec22524eb8f9c854fbb4d9acd5d7e5be9e4 ]

    A fabric controller can sometimes send an RDP request right before a link
    down event.  Because of this outstanding RDP request, the driver does not
    remove the last reference count on its ndlp causing a potential leak of RPI
    resources when devloss tmo fires.

    In lpfc_cmpl_els_rsp(), modify the NPIV clause to always allow the
    lpfc_drop_node() routine to execute when not registered with SCSI
    transport.

    This relaxes the contraint that an NPIV ndlp must be in a specific state in
    order to call lpfc_drop node.  Logic is revised such that the
    lpfc_drop_node() routine is always called to ensure the last ndlp decrement
    occurs.

    Signed-off-by: Justin Tee <justin.tee@broadcom.com>
    Link: https://lore.kernel.org/r/20230301231626.9621-7-justintee8345@gmail.com
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
```

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

commit ad050f6cf681ebb850a9d4bc19474d3896476301
Author: Justin Tee <justin.tee@broadcom.com>
Date:   Wed Mar 1 15:16:17 2023 -0800

    scsi: lpfc: Prevent lpfc_debugfs_lockstat_write() buffer overflow

    [ Upstream commit c6087b82a9146826564a55c5ca0164cac40348f5 ]

    A static code analysis tool flagged the possibility of buffer overflow when
    using copy_from_user() for a debugfs entry.

    Currently, it is possible that copy_from_user() copies more bytes than what
    would fit in the mybuf char array.  Add a min() restriction check between
    sizeof(mybuf) - 1 and nbytes passed from the userspace buffer to protect
    against buffer overflow.

    Link: https://lore.kernel.org/r/20230301231626.9621-2-justintee8345@gmail.com
    Signed-off-by: Justin Tee <justin.tee@broadcom.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c2e7776843a953fd7e48895c3880c277f996193e
Author: Jan Kara <jack@suse.cz>
Date:   Wed Mar 1 11:59:39 2023 +0100

    ext2: Check block size validity during mount

    [ Upstream commit 62aeb94433fcec80241754b70d0d1836d5926b0a ]

    Check that log of block size stored in the superblock has sensible
    value. Otherwise the shift computing the block size can overflow leading
    to undefined behavior.

    Reported-by: syzbot+4fec412f59eba8c01b77@syzkaller.appspotmail.com
    Signed-off-by: Jan Kara <jack@suse.cz>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit bc368a4426d18072b056925832ed3cd066e8f45a
Author: Hector Martin <marcan@marcan.st>
Date:   Tue Feb 14 18:24:20 2023 +0900

    wifi: brcmfmac: pcie: Add IDs/properties for BCM4387

    [ Upstream commit 117ace4014cce3fb78b40eb8028bb0f4fc37dd6f ]

    This chip is present on Apple M1 Pro/Max (t600x) platforms:

    * maldives   (apple,j314s): MacBook Pro (14-inch, M1 Pro, 2021)
    * maldives   (apple,j314c): MacBook Pro (14-inch, M1 Max, 2021)
    * madagascar (apple,j316s): MacBook Pro (16-inch, M1 Pro, 2021)
    * madagascar (apple,j316c): MacBook Pro (16-inch, M1 Max, 2021)

    Reviewed-by: Linus Walleij <linus.walleij@linaro.org>
    Reviewed-by: Arend van Spriel <arend.vanspriel@broadcom.com>
    Signed-off-by: Hector Martin <marcan@marcan.st>
    Signed-off-by: Kalle Valo <kvalo@kernel.org>
    Link: https://lore.kernel.org/r/20230214092423.15175-7-marcan@marcan.st
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2fa3a5226b05e0a797c68b9609dcebe0cd236b27
Author: Hector Martin <marcan@marcan.st>
Date:   Tue Feb 14 18:24:19 2023 +0900
```

```
    wifi: brcmfmac: cfg80211: Pass the PMK in binary instead of hex

    [ Upstream commit 89b89e52153fda2733562776c7c9d9d3ebf8dd6d ]

    Apparently the hex passphrase mechanism does not work on newer
    chips/firmware (e.g. BCM4387). It seems there was a simple way of
    passing it in binary all along, so use that and avoid the hexification.

    OpenBSD has been doing it like this from the beginning, so this should
    work on all chips.

    Also clear the structure before setting the PMK. This was leaking
    uninitialized stack contents to the device.

    Reviewed-by: Linus Walleij <linus.walleij@linaro.org>
    Reviewed-by: Arend van Spriel <arend.vanspriel@broadcom.com>
    Signed-off-by: Hector Martin <marcan@marcan.st>
    Signed-off-by: Kalle Valo <kvalo@kernel.org>
    Link: https://lore.kernel.org/r/20230214092423.15175-6-marcan@marcan.st
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ba72baed066f3bfa8b489e4b58f1fcaf51c04f83
Author: Hector Martin <marcan@marcan.st>
Date:   Tue Feb 14 17:00:34 2023 +0900

    wifi: brcmfmac: pcie: Provide a buffer of random bytes to the device

    [ Upstream commit 91918ce88d9fef408bb12c46a27c73d79b604c20 ]

    Newer Apple firmwares on chipsets without a hardware RNG require the
    host to provide a buffer of 256 random bytes to the device on
    initialization. This buffer is present immediately before NVRAM,
    suffixed by a footer containing a magic number and the buffer length.

    This won't affect chips/firmwares that do not use this feature, so do it
    unconditionally for all Apple platforms (those with an Apple OTP).

    Reviewed-by: Linus Walleij <linus.walleij@linaro.org>
    Signed-off-by: Hector Martin <marcan@marcan.st>
    Reviewed-by: Julian Calaby <julian.calaby@gmail.com>
    Signed-off-by: Kalle Valo <kvalo@kernel.org>
    Link: https://lore.kernel.org/r/20230214080034.3828-3-marcan@marcan.st
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 65e4974163ef8d98e9f83e72417f2904f8842cae
Author: Kumar Kartikeya Dwivedi <memxor@gmail.com>
Date:   Tue Feb 21 21:06:42 2023 +0100

    bpf: Annotate data races in bpf_local_storage

    [ Upstream commit 0a09a2f933c73dc76ab0b72da6855f44342a8903 ]

    There are a few cases where hlist_node is checked to be unhashed without
    holding the lock protecting its modification. In this case, one must use
    hlist_unhashed_lockless to avoid load tearing and KCSAN reports. Fix
    this by using lockless variant in places not protected by the lock.

    Since this is not prompted by any actual KCSAN reports but only from
    code review, I have not included a fixes tag.

    Cc: Martin KaFai Lau <martin.lau@kernel.org>
    Cc: KP Singh <kpsingh@kernel.org>
    Signed-off-by: Kumar Kartikeya Dwivedi <memxor@gmail.com>
    Link: https://lore.kernel.org/r/20230221200646.2500777-4-memxor@gmail.com
    Signed-off-by: Alexei Starovoitov <ast@kernel.org>
```

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

commit d3b9dc63d78d6946e90658dc74f8b34b55b84095
Author: Ramya Gnanasekar <quic_rgnanase@quicinc.com>
Date:   Mon Jan 23 15:21:41 2023 +0530

    wifi: ath12k: PCI ops for wakeup/release MHI

    [ Upstream commit 80e396586d0a94c42015dd9472176d89a3b0e4ca ]

    Wakeup/release MHI is not needed before pci_read/write for QCN9274.
    Since wakeup & release MHI is enabled for all QCN9274 and
    WCN7850, below MHI assert is seen in QCN9274

    [  784.906613] BUG: sleeping function called from invalid context at
drivers/bus/mhi/host/pm.c:989
    [  784.906633] in_atomic(): 1, irqs_disabled(): 0, non_block: 0, pid: 0, name: swapper/3
    [  784.906637] preempt_count: 503, expected: 0
    [  784.906641] RCU nest depth: 0, expected: 0
    [  784.906644] 2 locks held by swapper/3/0:
    [  784.906646]  #0: ffff8ed348e429e0 (&ab->ce.ce_lock){+.-.}-{2:2}, at:
ath12k_ce_recv_process_cb+0xb3/0x2f0 [ath12k]
    [  784.906664]  #1: ffff8ed348e491f0 (&srng->lock_key#3){+.-.}-{2:2}, at:
ath12k_ce_recv_process_cb+0xfb/0x2f0 [ath12k]
    [  784.906678] Preemption disabled at:
    [  784.906680] [<0000000000000000>] 0x0
    [  784.906686] CPU: 3 PID: 0 Comm: swapper/3 Tainted: G        W  O       6.1.0-rc2+ #3
    [  784.906688] Hardware name: Intel(R) Client Systems NUC8i7HVK/NUC8i7HVB, BIOS
HNKBLi70.86A.0056.2019.0506.1527 05/06/2019
    [  784.906690] Call Trace:
    [  784.906691]  <IRQ>
    [  784.906693]  dump_stack_lvl+0x56/0x7b
    [  784.906698]  __might_resched+0x21c/0x270
    [  784.906704]  __mhi_device_get_sync+0x7d/0x1c0 [mhi]
    [  784.906714]  mhi_device_get_sync+0xd/0x20 [mhi]
    [  784.906719]  ath12k_pci_write32+0x75/0x170 [ath12k]
    [  784.906729]  ath12k_hal_srng_access_end+0x55/0xc0 [ath12k]
    [  784.906737]  ath12k_ce_recv_process_cb+0x1f3/0x2f0 [ath12k]
    [  784.906776]  ? ath12k_pci_ce_tasklet+0x11/0x30 [ath12k]
    [  784.906788]  ath12k_pci_ce_tasklet+0x11/0x30 [ath12k]
    [  784.906813]  tasklet_action_common.isra.18+0xb7/0xe0
    [  784.906820]  __do_softirq+0xd0/0x4c9
    [  784.906826]  irq_exit_rcu+0x88/0xe0
    [  784.906828]  common_interrupt+0xa5/0xc0
    [  784.906831]  </IRQ>
    [  784.906832]  <TASK>

    Adding function callbacks for MHI wakeup and release operations.
    QCN9274 does not need wakeup/release, function callbacks are initialized
    to NULL. In case of WCN7850, shadow registers are used to access rings.
    Since, shadow register's offset is less than ACCESS_ALWAYS_OFF,
    mhi_device_get_sync() or mhi_device_put() to wakeup
    and release mhi will not be called during service ring accesses.

    Tested-on: QCN9274 hw2.0 PCI WLAN.WBE.1.0-03171-QCAHKSWPL_SILICONZ-1
    Tested-on: WCN7850 hw2.0 PCI WLAN.HMT.1.0-03427-QCAHMTSWPL_V1.0_V2.0_SILICONZ-1.15378.4

    Signed-off-by: Ramya Gnanasekar <quic_rgnanase@quicinc.com>
    Signed-off-by: Kalle Valo <quic_kvalo@quicinc.com>
    Link: https://lore.kernel.org/r/20230123095141.5310-1-quic_rgnanase@quicinc.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9faf7c696610a348ca94a224d55c946b19b3279d
Author: Ramya Gnanasekar <quic_rgnanase@quicinc.com>
Date:   Sun Jan 22 07:19:36 2023 +0530
```

```
    wifi: ath12k: Handle lock during peer_id find

    [ Upstream commit 95a389e2ff3212d866cc51c77d682d2934074eb8 ]

    ath12k_peer_find_by_id() requires that the caller hold the
    ab->base_lock. Currently the WBM error path does not hold
    the lock and calling that function, leads to the
    following lockdep_assert()in QCN9274:

    [105162.160893] ------------[ cut here ]------------
    [105162.160916] WARNING: CPU: 3 PID: 0 at drivers/net/wireless/ath/ath12k/peer.c:71
ath12k_peer_find_by_id+0x52/0x60 [ath12k]
    [105162.160933] Modules linked in: ath12k(O) qrtr_mhi qrtr mac80211 cfg80211 mhi
qmi_helpers libarc4 nvme nvme_core [last unloaded: ath12k(O)]
    [105162.160967] CPU: 3 PID: 0 Comm: swapper/3 Tainted: G        W  O       6.1.0-rc2+ #3
    [105162.160972] Hardware name: Intel(R) Client Systems NUC8i7HVK/NUC8i7HVB, BIOS
HNKBLi70.86A.0056.2019.0506.1527 05/06/2019
    [105162.160977] RIP: 0010:ath12k_peer_find_by_id+0x52/0x60 [ath12k]
    [105162.160990] Code: 07 eb 0f 39 68 24 74 0a 48 8b 00 48 39 f8 75 f3 31 c0 5b 5d c3 48
8d bf b0 f2 00 00 be ff ff ff ff e8 22 20 c4 e2 85 c0 75 bf <0f> 0b eb bb 66 2e 0f 1f 84 00
00 00 00 00 41 54 4c 8d a7 98 f2 00
    [105162.160996] RSP: 0018:ffffa223001acc60 EFLAGS: 00010246
    [105162.161003] RAX: 0000000000000000 RBX: ffff9f0573940000 RCX: 0000000000000000
    [105162.161008] RDX: 0000000000000001 RSI: ffffffffa3951c8e RDI: ffffffffa39a96d7
    [105162.161013] RBP: 000000000000000a R08: 0000000000000000 R09: 0000000000000000
    [105162.161017] R10: ffffa223001acb40 R11: ffffffffa3d57c60 R12: ffff9f057394f2e0
    [105162.161022] R13: ffff9f0573940000 R14: ffff9f04ecd659c0 R15: ffff9f04d5a9b040
    [105162.161026] FS:  0000000000000000(0000) GS:ffff9f0575600000(0000)
knlGS:0000000000000000
    [105162.161031] CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    [105162.161036] CR2: 00001d5c8277a008 CR3: 00000001e6224006 CR4: 00000000003706e0
    [105162.161041] Call Trace:
    [105162.161046]  <IRQ>
    [105162.161051]  ath12k_dp_rx_process_wbm_err+0x6da/0xaf0 [ath12k]
    [105162.161072]  ? ath12k_dp_rx_process_err+0x80e/0x15a0 [ath12k]
    [105162.161084]  ? __lock_acquire+0x4ca/0x1a60
    [105162.161104]  ath12k_dp_service_srng+0x263/0x310 [ath12k]
    [105162.161120]  ath12k_pci_ext_grp_napi_poll+0x1c/0x70 [ath12k]
    [105162.161133]  __napi_poll+0x22/0x260
    [105162.161141]  net_rx_action+0x2f8/0x380
    [105162.161153]  __do_softirq+0xd0/0x4c9
    [105162.161162]  irq_exit_rcu+0x88/0xe0
    [105162.161169]  common_interrupt+0xa5/0xc0
    [105162.161174]  </IRQ>
    [105162.161179]  <TASK>
    [105162.161184]  asm_common_interrupt+0x22/0x40

    Handle spin lock/unlock in WBM error path to hold the necessary lock
    expected by ath12k_peer_find_by_id().

    Tested-on: QCN9274 hw2.0 PCI WLAN.WBE.1.0-03171-QCAHKSWPL_SILICONZ-1

    Signed-off-by: Ramya Gnanasekar <quic_rgnanase@quicinc.com>
    Signed-off-by: Kalle Valo <quic_kvalo@quicinc.com>
    Link: https://lore.kernel.org/r/20230122014936.3594-1-quic_rgnanase@quicinc.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2215d261f0d977b8686d8abd1db66b3789af4c5d
Author: Kees Cook <keescook@chromium.org>
Date:   Wed Feb 15 20:31:38 2023 +0200

    wifi: ath: Silence memcpy run-time false positive warning

    [ Upstream commit bfcc8ba45eb87bfaaff900bbad2b87b204899d41 ]
```

```
    The memcpy() in ath_key_config() was attempting to write across
    neighboring struct members in struct ath_keyval. Introduce a wrapping
    struct_group, kv_values, to be the addressable target of the memcpy
    without overflowing an individual member. Silences the false positive
    run-time warning:

      memcpy: detected field-spanning write (size 32) of single field "hk.kv_val" at
drivers/net/wireless/ath/key.c:506 (size 16)

    Link: https://bbs.archlinux.org/viewtopic.php?id=282254
    Cc: Kalle Valo <kvalo@kernel.org>
    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-wireless@vger.kernel.org
    Cc: netdev@vger.kernel.org
    Signed-off-by: Kees Cook <keescook@chromium.org>
    Signed-off-by: Kalle Valo <quic_kvalo@quicinc.com>
    Link: https://lore.kernel.org/r/20230210054310.never.554-kees@kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3d0e4fcb7130d0a6199c12e910363b4b81248033
Author: Aleksandr Mezin <mezin.alexander@gmail.com>
Date:   Sun Feb 19 12:59:19 2023 +0200

    hwmon: (nzxt-smart2) add another USB ID

    [ Upstream commit 4a148e9b1ee04e608263fa9536a96214d5561220 ]

    This seems to be a new revision of the device. RGB controls have changed,
    but this driver doesn't touch them anyway.

    Fan speed control reported to be working with existing userspace (hidraw)
    software, so I assume it's compatible. Fan channel count is the same.

    Recently added (0x1e71, 0x2019) seems to be the same device.

    Discovered in liquidctl project:

    https://github.com/liquidctl/liquidctl/issues/541

    Signed-off-by: Aleksandr Mezin <mezin.alexander@gmail.com>
    Link: https://lore.kernel.org/r/20230219105924.333007-1-mezin.alexander@gmail.com
    Signed-off-by: Guenter Roeck <linux@roeck-us.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d804adef7b23b22bb82e1b3dd113e9073cea9bc1
Author: Feng Jiang <jiangfeng@kylinos.cn>
Date:   Wed Apr 12 17:37:34 2023 +0800

    platform/x86/amd: pmc: Fix memory leak in amd_pmc_stb_debugfs_open_v2()

    [ Upstream commit f6e7ac4c35a28aef0be93b32c533ae678ad0b9e7 ]

    Function amd_pmc_stb_debugfs_open_v2() may be called when the STB
    debug mechanism enabled.

    When amd_pmc_send_cmd() fails, the 'buf' needs to be released.

    Signed-off-by: Feng Jiang <jiangfeng@kylinos.cn>
    Link: https://lore.kernel.org/r/20230412093734.1126410-1-jiangfeng@kylinos.cn
    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 b8e19bf3b4aebd855be01b64674187dcf6d1db51
Author: Wei Chen <harperchen1110@gmail.com>
Date:   Wed Mar 29 09:05:13 2023 +0100

    media: mediatek: vcodec: Fix potential array out-of-bounds in decoder queue_setup

    [ Upstream commit 8fbcf730cb89c3647f3365226fe7014118fa93c7 ]

    variable *nplanes is provided by user via system call argument. The
    possible value of q_data->fmt->num_planes is 1-3, while the value
    of *nplanes can be 1-8. The array access by index i can cause array
    out-of-bounds.

    Fix this bug by checking *nplanes against the array size.

    Signed-off-by: Wei Chen <harperchen1110@gmail.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 9dd546235d7cf0c6ab392123f64c690e7652cf75
Author: Mukul Joshi <mukul.joshi@amd.com>
Date:   Tue Apr 11 16:32:29 2023 -0400

    drm/amdgpu: Enable IH retry CAM on GFX9

    [ Upstream commit 318e431b306e966d2ee99e900a11bdc9a701ee83 ]

    This patch enables the IH retry CAM on GFX9 series cards. This
    retry filter is used to prevent sending lots of retry interrupts
    in a short span of time and overflowing the IH ring buffer. This
    will also help reduce CPU interrupt workload.

    Signed-off-by: Mukul Joshi <mukul.joshi@amd.com>
    Reviewed-by: Felix Kuehling <Felix.Kuehling@amd.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5855b62d2423fb9769ac41625badb3e4ad9e3359
Author: Laurent Pinchart <laurent.pinchart@ideasonboard.com>
Date:   Wed Feb 15 17:18:39 2023 +0200

    media: Prefer designated initializers over memset for subdev pad ops

    [ Upstream commit e3a69496a1cde364c74a600d7a370179b58aed29 ]

    Structures passed to subdev pad operations are all zero-initialized, but
    not always with the same kind of code constructs. While most drivers
    used designated initializers, which zero all the fields that are not
    specified, when declaring variables, some use memset(). Those two
    methods lead to the same end result, and, depending on compiler
    optimizations, may even be completely equivalent, but they're not
    consistent.

    Improve coding style consistency by using designated initializers
    instead of calling memset(). Where applicable, also move the variables
    to inner scopes of for loops to ensure correct initialization in all
    iterations.

    Signed-off-by: Laurent Pinchart <laurent.pinchart@ideasonboard.com>
    Reviewed-by: Lad Prabhakar <prabhakar.csengg@gmail.com> # For am437x
    Acked-by: Sakari Ailus <sakari.ailus@linux.intel.com>
    Reviewed-by: Tomi Valkeinen <tomi.valkeinen@ideasonboard.com>
```

```
    Reviewed-by: Kieran Bingham <kieran.bingham+renesas@ideasonboard.com>
    Reviewed-by: Philipp Zabel <p.zabel@pengutronix.de>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0ebc02d9ff85626a526353584526da6aa9c96792
Author: lyndonli <Lyndon.Li@amd.com>
Date:   Thu Apr 6 15:30:34 2023 +0800

    drm/amdgpu: Fix sdma v4 sw fini error

    [ Upstream commit 5e08e9c742a00384e5abe74bd40cf4dc15cb3a2e ]

    Fix sdma v4 sw fini error for sdma 4.2.2 to
    solve the following general protection fault

    [  +0.108196] general protection fault, probably for non-canonical
    address 0xd5e5a4ae79d24a32: 0000 [#1] PREEMPT SMP PTI
    [  +0.000018] RIP: 0010:free_fw_priv+0xd/0x70
    [  +0.000022] Call Trace:
    [  +0.000012]  <TASK>
    [  +0.000011]  release_firmware+0x55/0x80
    [  +0.000021]  amdgpu_ucode_release+0x11/0x20 [amdgpu]
    [  +0.000415]  amdgpu_sdma_destroy_inst_ctx+0x4f/0x90 [amdgpu]
    [  +0.000360]  sdma_v4_0_sw_fini+0xce/0x110 [amdgpu]

    Signed-off-by: lyndonli <Lyndon.Li@amd.com>
    Reviewed-by: Likun Gao <Likun.Gao@amd.com>
    Reviewed-by: Feifei Xu <Feifei.Xu@amd.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dea2dbec716c38a0b73b6ad01d91e2b120cc5f1e
Author: Mario Limonciello <mario.limonciello@amd.com>
Date:   Thu Mar 23 14:07:06 2023 -0500

    drm/amd: Fix an out of bounds error in BIOS parser

    [ Upstream commit d116db180decec1b21bba31d2ff495ac4d8e1b83 ]

    The array is hardcoded to 8 in atomfirmware.h, but firmware provides
    a bigger one sometimes. Deferencing the larger array causes an out
    of bounds error.

    commit 4fc1ba4aa589 ("drm/amd/display: fix array index out of bound error
    in bios parser") fixed some of this, but there are two other cases
    not covered by it.  Fix those as well.

    Reported-by: erhard_f@mailbox.org
    Link: https://bugzilla.kernel.org/show_bug.cgi?id=214853
    Link: https://gitlab.freedesktop.org/drm/amd/-/issues/2473
    Signed-off-by: Mario Limonciello <mario.limonciello@amd.com>
    Reviewed-by: Harry Wentland <harry.wentland@amd.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 22b9b3b7b02468e3987af262842bd7f881dd0212
Author: Paul Hsieh <Paul.Hsieh@amd.com>
Date:   Wed Mar 22 17:46:31 2023 +0800

    drm/amd/display: Correct DML calculation to follow HW SPEC

    [ Upstream commit 385c3e4c29e1d4ce8f68687a8c84621e4c0e0416 ]

    [Why]
```

```
    In 2560x1600@240p eDP panel, driver use lowest voltage level
    to play 1080p video cause underflow. According to HW SPEC,
    the senario should use high voltage level.

    [How]
    ChromaPre value is zero when bandwidth validation.
    Correct ChromaPre calculation.

    Reviewed-by: Nicholas Kazlauskas <Nicholas.Kazlauskas@amd.com>
    Reviewed-by: Jun Lei <Jun.Lei@amd.com>
    Acked-by: Qingqing Zhuo <qingqing.zhuo@amd.com>
    Signed-off-by: Paul Hsieh <Paul.Hsieh@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 77dd6c529b9ccb18292b9c71ab6947b7742b74d1
Author: Hans de Goede <hdegoede@redhat.com>
Date:   Tue Apr 4 13:02:51 2023 +0200

    ACPI: video: Remove desktops without backlight DMI quirks

    [ Upstream commit abe4f5ae5efa6a63c7d5abfa07eb02bb56b4654e ]

    After the recent backlight changes acpi_video# backlight devices are only
    registered when explicitly requested from the cmdline, by DMI quirk or by
    the GPU driver.

    This means that we no longer get false-positive backlight control support
    advertised on desktop boards.

    Remove the 3 DMI quirks for desktop boards where the false-positive issue
    was fixed through quirks before. Note many more desktop boards were
    affected but we never build a full quirk list for this.

    Reviewed-by: Mario Limonciello <mario.limonciello@amd.com>
    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b4d9b3990ee8c6fe4edc80cb4786f115ea4c3b4b
Author: Bingbu Cao <bingbu.cao@intel.com>
Date:   Fri Mar 10 23:19:10 2023 +0800

    media: ipu3-cio2: support multiple sensors and VCMs with same HID

    [ Upstream commit 567f97bd381fd79fa8563808118fc757cb6fa4ff ]

    In current cio2-bridge, it is using the hid name to register software
    node and software node will create kobject and sysfs entry according to
    the node name, if there are multiple sensors and VCMs which are sharing
    same HID name, it will cause the software nodes registration failure:

    sysfs: cannot create duplicate filename '/kernel/software_nodes/dw9714'
    ...
    Call Trace:
    software_node_register_nodes
    cio2_bridge_init
    ...
    kobject_add_internal failed for dw9714 with -EEXIST,
    don't try to register things with the same name in the same directory.

    One solution is appending the sensor link(Mipi Port) in SSDB as suffix
    of the node name to fix this problem.
```

```
    Signed-off-by: Bingbu Cao <bingbu.cao@intel.com>
    Signed-off-by: Sakari Ailus <sakari.ailus@linux.intel.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 867a4f6cf1a8f511c06e131477988b3b3e7a0633
Author: Shanker Donthineni <sdonthineni@nvidia.com>
Date:   Sat Mar 18 21:43:14 2023 -0500

    irqchip/gicv3: Workaround for NVIDIA erratum T241-FABRIC-4

    [ Upstream commit 35727af2b15d98a2dd2811d631d3a3886111312e ]

    The T241 platform suffers from the T241-FABRIC-4 erratum which causes
    unexpected behavior in the GIC when multiple transactions are received
    simultaneously from different sources. This hardware issue impacts
    NVIDIA server platforms that use more than two T241 chips
    interconnected. Each chip has support for 320 {E}SPIs.

    This issue occurs when multiple packets from different GICs are
    incorrectly interleaved at the target chip. The erratum text below
    specifies exactly what can cause multiple transfer packets susceptible
    to interleaving and GIC state corruption. GIC state corruption can
    lead to a range of problems, including kernel panics, and unexpected
    behavior.

    >From the erratum text:
      "In some cases, inter-socket AXI4 Stream packets with multiple
      transfers, may be interleaved by the fabric when presented to ARM
      Generic Interrupt Controller. GIC expects all transfers of a packet
      to be delivered without any interleaving.

      The following GICv3 commands may result in multiple transfer packets
      over inter-socket AXI4 Stream interface:
       - Register reads from GICD_I* and GICD_N*
       - Register writes to 64-bit GICD registers other than GICD_IROUTERn*
       - ITS command MOVALL

      Multiple commands in GICv4+ utilize multiple transfer packets,
      including VMOVP, VMOVI, VMAPP, and 64-bit register accesses."

      This issue impacts system configurations with more than 2 sockets,
      that require multi-transfer packets to be sent over inter-socket
      AXI4 Stream interface between GIC instances on different sockets.
      GICv4 cannot be supported. GICv3 SW model can only be supported
      with the workaround. Single and Dual socket configurations are not
      impacted by this issue and support GICv3 and GICv4."

    Link: https://developer.nvidia.com/docs/t241-fabric-4/nvidia-t241-fabric-4-errata.pdf

    Writing to the chip alias region of the GICD_In{E} registers except
    GICD_ICENABLERn has an equivalent effect as writing to the global
    distributor. The SPI interrupt deactivate path is not impacted by
    the erratum.

    To fix this problem, implement a workaround that ensures read accesses
    to the GICD_In{E} registers are directed to the chip that owns the
    SPI, and disable GICv4.x features. To simplify code changes, the
    gic_configure_irq() function uses the same alias region for both read
    and write operations to GICD_ICFGR.

    Co-developed-by: Vikram Sethi <vsethi@nvidia.com>
    Signed-off-by: Vikram Sethi <vsethi@nvidia.com>
    Signed-off-by: Shanker Donthineni <sdonthineni@nvidia.com>
    Acked-by: Sudeep Holla <sudeep.holla@arm.com> (for SMCCC/SOC ID bits)
```

```
    Signed-off-by: Marc Zyngier <maz@kernel.org>
    Link: https://lore.kernel.org/r/20230319024314.3540573-2-sdonthineni@nvidia.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4e8738b4c5e39ab4bec206e4023baa522486eb83
Author: Konrad Dybcio <konrad.dybcio@linaro.org>
Date:   Fri Apr 7 21:54:41 2023 +0200

    arm64: dts: qcom: sm6115-j606f: Add ramoops node

    [ Upstream commit 8b0ac59c2da69aaf8e65c6bd648a06b755975302 ]

    Add a ramoops node to enable retrieving crash logs.

    Signed-off-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230406-topic-lenovo_features-v2-1-
625d7cb4a944@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit da6bb810148b12dfaac7b87b3e7ce1b8a7d00518
Author: Konrad Dybcio <konrad.dybcio@linaro.org>
Date:   Thu Apr 6 14:55:45 2023 +0200

    arm64: dts: qcom: sdm845-polaris: Drop inexistent properties

    [ Upstream commit fbc3a1df2866608ca43e7e6d602f66208a5afd88 ]

    Drop the qcom,snoc-host-cap-skip-quirk that was never introduced to
    solve schema warnings.

    Reviewed-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Signed-off-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230406-topic-ath10k_bindings-v3-2-
00895afc7764@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ed2e1e85644ca3d351324e9927a538c8af4df654
Author: void0red <30990023+void0red@users.noreply.github.com>
Date:   Wed Apr 5 15:57:57 2023 +0200

    ACPICA: ACPICA: check null return of ACPI_ALLOCATE_ZEROED in acpi_db_display_objects

    [ Upstream commit ae5a0eccc85fc960834dd66e3befc2728284b86c ]

    ACPICA commit 0d5f467d6a0ba852ea3aad68663cbcbd43300fd4

    ACPI_ALLOCATE_ZEROED may fails, object_info might be null and will cause
    null pointer dereference later.

    Link: https://github.com/acpica/acpica/commit/0d5f467d
    Signed-off-by: Bob Moore <robert.moore@intel.com>
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3048c6b84a51e4ba4a89385ed218d19a670edd47
Author: Tamir Duberstein <tamird@google.com>
Date:   Wed Apr 5 15:42:43 2023 +0200

    ACPICA: Avoid undefined behavior: applying zero offset to null pointer

    [ Upstream commit 05bb0167c80b8f93c6a4e0451b7da9b96db990c2 ]

    ACPICA commit 770653e3ba67c30a629ca7d12e352d83c2541b1e
```

```
    Before this change we see the following UBSAN stack trace in Fuchsia:

      #0    0x000021e4213b3302 in acpi_ds_init_aml_walk(struct acpi_walk_state*, union
 acpi_parse_object*, struct acpi_namespace_node*, u8*, u32, struct acpi_evaluate_info*, u8)
 ../../third_party/acpica/source/components/dispatcher/dswstate.c:682 <platform-bus-
 x86.so>+0x233302
      #1.2  0x000020d0f660777f in ubsan_get_stack_trace() compiler-
 rt/lib/ubsan/ubsan_diag.cpp:41 <libclang_rt.asan.so>+0x3d77f
      #1.1  0x000020d0f660777f in maybe_print_stack_trace() compiler-
 rt/lib/ubsan/ubsan_diag.cpp:51 <libclang_rt.asan.so>+0x3d77f
      #1    0x000020d0f660777f in ~scoped_report() compiler-rt/lib/ubsan/ubsan_diag.cpp:387
 <libclang_rt.asan.so>+0x3d77f
      #2    0x000020d0f660b96d in handlepointer_overflow_impl() compiler-
 rt/lib/ubsan/ubsan_handlers.cpp:809 <libclang_rt.asan.so>+0x4196d
      #3    0x000020d0f660b50d in compiler-rt/lib/ubsan/ubsan_handlers.cpp:815
 <libclang_rt.asan.so>+0x4150d
      #4    0x000021e4213b3302 in acpi_ds_init_aml_walk(struct acpi_walk_state*, union
 acpi_parse_object*, struct acpi_namespace_node*, u8*, u32, struct acpi_evaluate_info*, u8)
 ../../third_party/acpica/source/components/dispatcher/dswstate.c:682 <platform-bus-
 x86.so>+0x233302
      #5    0x000021e4213e2369 in acpi_ds_call_control_method(struct acpi_thread_state*,
 struct acpi_walk_state*, union acpi_parse_object*)
 ../../third_party/acpica/source/components/dispatcher/dsmethod.c:605 <platform-bus-
 x86.so>+0x262369
      #6    0x000021e421437fac in acpi_ps_parse_aml(struct acpi_walk_state*)
 ../../third_party/acpica/source/components/parser/psparse.c:550 <platform-bus-
 x86.so>+0x2b7fac
      #7    0x000021e4214464d2 in acpi_ps_execute_method(struct acpi_evaluate_info*)
 ../../third_party/acpica/source/components/parser/psxface.c:244 <platform-bus-
 x86.so>+0x2c64d2
      #8    0x000021e4213aa052 in acpi_ns_evaluate(struct acpi_evaluate_info*)
 ../../third_party/acpica/source/components/namespace/nseval.c:250 <platform-bus-
 x86.so>+0x22a052
      #9    0x000021e421413dd8 in acpi_ns_init_one_device(acpi_handle, u32, void*, void**)
 ../../third_party/acpica/source/components/namespace/nsinit.c:735 <platform-bus-
 x86.so>+0x293dd8
      #10   0x000021e421429e98 in acpi_ns_walk_namespace(acpi_object_type, acpi_handle, u32,
 u32, acpi_walk_callback, acpi_walk_callback, void*, void**)
 ../../third_party/acpica/source/components/namespace/nswalk.c:298 <platform-bus-
 x86.so>+0x2a9e98
      #11   0x000021e4214131ac in acpi_ns_initialize_devices(u32)
 ../../third_party/acpica/source/components/namespace/nsinit.c:268 <platform-bus-
 x86.so>+0x2931ac
      #12   0x000021e42147c40d in acpi_initialize_objects(u32)
 ../../third_party/acpica/source/components/utilities/utxfinit.c:304 <platform-bus-
 x86.so>+0x2fc40d
      #13   0x000021e42126d603 in acpi::acpi_impl::initialize_acpi(acpi::acpi_impl*)
 ../../src/devices/board/lib/acpi/acpi-impl.cc:224 <platform-bus-x86.so>+0xed603

    Add a simple check that avoids incrementing a pointer by zero, but
    otherwise behaves as before. Note that our findings are against ACPICA
    20221020, but the same code exists on master.

    Link: https://github.com/acpica/acpica/commit/770653e3
    Signed-off-by: Bob Moore <robert.moore@intel.com>
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit 32d6a046941abad45d8a716840e936eef7acb062
 Author: Douglas Anderson <dianders@chromium.org>
 Date:   Thu Jan 26 17:09:12 2023 -0800

    drm/msm/dp: Clean up handling of DP AUX interrupts
```

[ Upstream commit b20566cdef05cd40d95f10869d2a7646f48b1bbe ]

The DP AUX interrupt handling was a bit of a mess.
* There were two functions (one for "native" transfers and one for "i2c" transfers) that were quite similar. It was hard to say how many of the differences between the two functions were on purpose and how many of them were just an accident of how they were coded.
* Each function sometimes used "else if" to test for error bits and sometimes didn't and again it was hard to say if this was on purpose or just an accident.
* The two functions wouldn't notice whether "unknown" bits were set. For instance, there seems to be a bit "DP_INTR_PLL_UNLOCKED" and if it was set there would be no indication.
* The two functions wouldn't notice if more than one error was set.

Let's fix this by being more consistent / explicit about what we're doing.

By design this could cause different handling for AUX transfers, though I'm not actually aware of any bug fixed as a result of this patch (this patch was created because we simply noticed how odd the old code was by code inspection). Specific notes here:
1. In the old native transfer case if we got "done + wrong address" we'd ignore the "wrong address" (because of the "else if"). Now we won't.
2. In the old native transfer case if we got "done + timeout" we'd ignore the "timeout" (because of the "else if"). Now we won't.
3. In the old native transfer case we'd see "nack_defer" and translate it to the error number for "nack". This differed from the i2c transfer case where "nack_defer" was given the error number for "nack_defer". This 100% can't matter because the only user of this error number treats "nack defer" the same as "nack", so it's clear that the difference between the "native" and "i2c" was pointless here.
4. In the old i2c transfer case if we got "done" plus any error besides "nack" or "defer" then we'd ignore the error. Now we don't.
5. If there is more than one error signaled by the hardware it's possible that we'll report a different one than we used to. I don't know if this matters. If someone is aware of a case this matters we should document it and change the code to make it explicit.
6. One quirk we keep (I don't know if this is important) is that in the i2c transfer case if we see "done + defer" we report that as a "nack". That seemed too intentional in the old code to just drop.

After this change we will add extra logging, including:
* A warning if we see more than one error bit set.
* A warning if we see an unexpected interrupt.
* A warning if we get an AUX transfer interrupt when shouldn't.

It actually turns out that as a result of this change then at boot we sometimes see an error:
  [drm:dp_aux_isr] *ERROR* Unexpected DP AUX IRQ 0x01000000 when not busy
That means that, during init, we are seeing DP_INTR_PLL_UNLOCKED. For now I'm going to say that leaving this error reported in the logs is OK-ish and hopefully it will encourage someone to track down what's going on at init time.

One last note here is that this change renames one of the interrupt bits. The bit named "i2c done" clearly was used for native transfers being done too, so I renamed it to indicate this.

Signed-off-by: Douglas Anderson <dianders@chromium.org>
Tested-by: Kuogee Hsieh <quic_khsieh@quicinc.com>
Reviewed-by: Kuogee Hsieh <quic_khsieh@quicinc.com>
Patchwork: https://patchwork.freedesktop.org/patch/520658/

```
    Link:
https://lore.kernel.org/r/20230126170745.v2.1.I90ffed3ddd21e818ae534f820cb4d6d8638859ab@chan
geid
    Signed-off-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 28f227cf9661d725e8910ae8f4bb34acac19ca7e
Author: Nur Hussein <hussein@unixcat.org>
Date:   Thu Apr 6 04:25:59 2023 +0800

    drm/tegra: Avoid potential 32-bit integer overflow

    [ Upstream commit 2429b3c529da29d4277d519bd66d034842dcd70c ]

    In tegra_sor_compute_config(), the 32-bit value mode->clock is
    multiplied by 1000, and assigned to the u64 variable pclk. We can avoid
    a potential 32-bit integer overflow by casting mode->clock to u64 before
    we do the arithmetic and assignment.

    Signed-off-by: Nur Hussein <hussein@unixcat.org>
    Signed-off-by: Thierry Reding <treding@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5afb01cba7d78180d454a5d6b66d2ad3ab0920b3
Author: Karol Wachowski <karol.wachowski@linux.intel.com>
Date:   Mon Apr 3 14:15:45 2023 +0200

    accel/ivpu: Remove D3hot delay for Meteorlake

    [ Upstream commit cb949ce504e829193234e26cb3042bb448465d52 ]

    VPU on MTL has hardware optimizations and does not require 10ms
    D0 - D3hot transition delay imposed by PCI specification (PCIe
    r6.0, sec 5.9.) .

    The delay removal is traditionally done by adding PCI ID to
    quirk_remove_d3hot_delay() in drivers/pci/quirks.c . But since
    we do not need that optimization before driver probe and we
    can better specify in the ivpu driver on what (future) hardware
    use the optimization, we do not use quirk_remove_d3hot_delay()
    for that.

    Signed-off-by: Karol Wachowski <karol.wachowski@linux.intel.com>
    Signed-off-by: Stanislaw Gruszka <stanislaw.gruszka@linux.intel.com>
    Reviewed-by: Jeffrey Hugo <quic_jhugo@quicinc.com>
    Signed-off-by: Jacek Lawrynowicz <jacek.lawrynowicz@linux.intel.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20230403121545.2995279-1-
stanislaw.gruszka@linux.intel.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c1ff9ca8901adcdd8b4b49db9243fd43862931c8
Author: Arnaud Pouliquen <arnaud.pouliquen@foss.st.com>
Date:   Fri Mar 31 18:06:34 2023 +0200

    remoteproc: stm32_rproc: Add mutex protection for workqueue

    [ Upstream commit 35bdafda40cc343ad2ba2cce105eba03a70241cc ]

    The workqueue may execute late even after remoteproc is stopped or
    stopping, some resources (rpmsg device and endpoint) have been
    released in rproc_stop_subdevices(), then rproc_vq_interrupt()
    accessing these resources will cause kernel dump.

    Call trace:
    virtqueue_add_inbuf
```

```
      virtqueue_add_inbuf
      rpmsg_recv_single
      rpmsg_recv_done
      vring_interrupt
      stm32_rproc_mb_vq_work
      process_one_work
      worker_thread
      kthread

      Suggested-by: Mathieu Poirier <mathieu.poirier@linaro.org>
      Signed-off-by: Arnaud Pouliquen <arnaud.pouliquen@foss.st.com>
      Link: https://lore.kernel.org/r/20230331160634.3113031-1-arnaud.pouliquen@foss.st.com
      Signed-off-by: Mathieu Poirier <mathieu.poirier@linaro.org>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit b75729bd5e85e42f2430adb439c76a5e5b1efd13
  Author: Ayush Gupta <ayugupta@amd.com>
  Date:   Thu Mar 16 15:57:30 2023 -0400

      drm/amd/display: fixed dcn30+ underflow issue

      [ Upstream commit 37403ced9f2873fab7f39ab4ac963bbb33fb0bc0 ]

      [Why]
      Observing underflow on dcn30+ system config at 4k144hz

      [How]
      We set the UCLK hardmax on AC/DC switch if softmax is enabled
      and also on boot. While booting up the UCLK Hardmax is set
      to softmax before the init sequence and the init sequence
      resets the hardmax to UCLK max which enables P-state switching.
      Just added a conditional check to avoid setting hardmax on init.

      Reviewed-by: Alvin Lee <Alvin.Lee2@amd.com>
      Reviewed-by: Martin Leung <Martin.Leung@amd.com>
      Acked-by: Qingqing Zhuo <qingqing.zhuo@amd.com>
      Signed-off-by: Ayush Gupta <ayugupta@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 fd2c99e81ae0dbdd62a154ef9c77fc01715cc020
  Author: Armin Wolf <W_Armin@gmx.de>
  Date:   Fri Mar 24 21:26:27 2023 +0100

      ACPI: EC: Fix oops when removing custom query handlers

      [ Upstream commit e5b492c6bb900fcf9722e05f4a10924410e170c1 ]

      When removing custom query handlers, the handler might still
      be used inside the EC query workqueue, causing a kernel oops
      if the module holding the callback function was already unloaded.

      Fix this by flushing the EC query workqueue when removing
      custom query handlers.

      Tested on a Acer Travelmate 4002WLMi

      Signed-off-by: Armin Wolf <W_Armin@gmx.de>
      Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 18d5ea5b746120a3972e6c347ad9428228445327
  Author: Pierre Gondois <pierre.gondois@arm.com>
  Date:   Thu Feb 16 09:49:19 2023 +0100
```

```
    firmware: arm_sdei: Fix sleep from invalid context BUG

    [ Upstream commit d2c48b2387eb89e0bf2a2e06e30987cf410acad4 ]

    Running a preempt-rt (v6.2-rc3-rt1) based kernel on an Ampere Altra
    triggers:

      BUG: sleeping function called from invalid context at kernel/locking/spinlock_rt.c:46
      in_atomic(): 0, irqs_disabled(): 128, non_block: 0, pid: 24, name: cpuhp/0
      preempt_count: 0, expected: 0
      RCU nest depth: 0, expected: 0
      3 locks held by cpuhp/0/24:
        #0: ffffda30217c70d0 (cpu_hotplug_lock){++++}-{0:0}, at: cpuhp_thread_fun+0x5c/0x248
        #1: ffffda30217c7120 (cpuhp_state-up){+.+.}-{0:0}, at: cpuhp_thread_fun+0x5c/0x248
        #2: ffffda3021c711f0 (sdei_list_lock){....}-{3:3}, at: sdei_cpuhp_up+0x3c/0x130
      irq event stamp: 36
      hardirqs last  enabled at (35): [<ffffda301e85b7bc>] finish_task_switch+0xb4/0x2b0
      hardirqs last disabled at (36): [<ffffda301e812fec>] cpuhp_thread_fun+0x21c/0x248
      softirqs last  enabled at (0): [<ffffda301e80b184>] copy_process+0x63c/0x1ac0
      softirqs last disabled at (0): [<0000000000000000>] 0x0
      CPU: 0 PID: 24 Comm: cpuhp/0 Not tainted 5.19.0-rc3-rt5-[...]
      Hardware name: WIWYNN Mt.Jade Server [...]
      Call trace:
        dump_backtrace+0x114/0x120
        show_stack+0x20/0x70
        dump_stack_lvl+0x9c/0xd8
        dump_stack+0x18/0x34
        __might_resched+0x188/0x228
        rt_spin_lock+0x70/0x120
        sdei_cpuhp_up+0x3c/0x130
        cpuhp_invoke_callback+0x250/0xf08
        cpuhp_thread_fun+0x120/0x248
        smpboot_thread_fn+0x280/0x320
        kthread+0x130/0x140
        ret_from_fork+0x10/0x20

    sdei_cpuhp_up() is called in the STARTING hotplug section,
    which runs with interrupts disabled. Use a CPUHP_AP_ONLINE_DYN entry
    instead to execute the cpuhp cb later, with preemption enabled.

    SDEI originally got its own cpuhp slot to allow interacting
    with perf. It got superseded by pNMI and this early slot is not
    relevant anymore. [1]

    Some SDEI calls (e.g. SDEI_1_0_FN_SDEI_PE_MASK) take actions on the
    calling CPU. It is checked that preemption is disabled for them.
    _ONLINE cpuhp cb are executed in the 'per CPU hotplug thread'.
    Preemption is enabled in those threads, but their cpumask is limited
    to 1 CPU.
    Move 'WARN_ON_ONCE(preemptible())' statements so that SDEI cpuhp cb
    don't trigger them.

    Also add a check for the SDEI_1_0_FN_SDEI_PRIVATE_RESET SDEI call
    which acts on the calling CPU.

    [1]:
    https://lore.kernel.org/all/5813b8c5-ae3e-87fd-fccc-94c9cd08816d@arm.com/

    Suggested-by: James Morse <james.morse@arm.com>
    Signed-off-by: Pierre Gondois <pierre.gondois@arm.com>
    Reviewed-by: James Morse <james.morse@arm.com>
    Link: https://lore.kernel.org/r/20230216084920.144064-1-pierre.gondois@arm.com
    Signed-off-by: Will Deacon <will@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit e51a534aba2bfd15935e26975218b92bca655802
Author: Sebastian Krzyszkowiak <sebastian.krzyszkowiak@puri.sm>
Date:   Thu Mar 9 21:46:05 2023 +0100

    arm64: dts: imx8mq-librem5: Remove dis_u3_susphy_quirk from usb_dwc3_0

    [ Upstream commit cfe9de291bd2bbce18c5cd79e1dd582cbbacdb4f ]

    This reduces power consumption in system suspend by about 10%.

    Signed-off-by: Sebastian Krzyszkowiak <sebastian.krzyszkowiak@puri.sm>
    Signed-off-by: Martin Kepplinger <martin.kepplinger@puri.sm>
    Signed-off-by: Shawn Guo <shawnguo@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 76fec5f01c9c70e11b85fdeb3f2707589c9238ca
Author: Zheng Wang <zyytlz.wz@163.com>
Date:   Wed Mar 8 00:43:38 2023 +0800

    memstick: r592: Fix UAF bug in r592_remove due to race condition

    [ Upstream commit 63264422785021704c39b38f65a78ab9e4a186d7 ]

    In r592_probe, dev->detect_timer was bound with r592_detect_timer.
    In r592_irq function, the timer function will be invoked by mod_timer.

    If we remove the module which will call hantro_release to make cleanup,
    there may be a unfinished work. The possible sequence is as follows,
    which will cause a typical UAF bug.

    Fix it by canceling the work before cleanup in r592_remove.

    CPU0                  CPU1

                         |r592_detect_timer
    r592_remove          |
      memstick_free_host|
      put_device;        |
      kfree(host);       |
                         |
                         | queue_work
                         |   &host->media_checker //use

    Signed-off-by: Zheng Wang <zyytlz.wz@163.com>
    Link: https://lore.kernel.org/r/20230307164338.1246287-1-zyytlz.wz@163.com
    Signed-off-by: Ulf Hansson <ulf.hansson@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 218fe9b624545f4bcfb16cdb35ac3d60c8b0d8c7
Author: Toby Chen <tobyc@nvidia.com>
Date:   Thu Mar 16 17:51:26 2023 -0700

    drm/rockchip: dw_hdmi: cleanup drm encoder during unbind

    [ Upstream commit b5af48eedcb53491c02ded55d5991e03d6da6dbf ]

    This fixes a use-after-free crash during rmmod.

    The DRM encoder is embedded inside the larger rockchip_hdmi,
    which is allocated with the component. The component memory
    gets freed before the main drm device is destroyed. Fix it
    by running encoder cleanup before tearing down its container.

    Signed-off-by: Toby Chen <tobyc@nvidia.com>
```

```
      [moved encoder cleanup above clk_disable, similar to bind-error-path]
      Signed-off-by: Heiko Stuebner <heiko@sntech.de>
      Link: https://patchwork.freedesktop.org/patch/msgid/20230317005126.496-1-
  tobyc@nvidia.com
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 8a632ff6a2bea49993002b4c46092a2aea625840
  Author: Kang Chen <void0red@gmail.com>
  Date:   Sun Feb 26 13:54:27 2023 +0800

      ACPI: processor: Check for null return of devm_kzalloc() in fch_misc_setup()

      [ Upstream commit 4dea41775d951ff1f7b472a346a8ca3ae7e74455 ]

      devm_kzalloc() may fail, clk_data->name might be NULL and will
      cause a NULL pointer dereference later.

      Signed-off-by: Kang Chen <void0red@gmail.com>
      [ rjw: Subject and changelog edits ]
      Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit fd909f904499bf81c0c4b9de6705019c51471e75
  Author: David E. Box <david.e.box@linux.intel.com>
  Date:   Thu Mar 16 15:46:28 2023 -0700

      platform/x86/intel: vsec: Explicitly enable capabilities

      [ Upstream commit 3f95ecf2a3e4db09e58d307932037e8f1210d6e7 ]

      Discovered Intel VSEC/DVSEC capabilities are enabled by default and only
      get disabled by quirk. Instead, remove such quirks and only enable support
      for capabilities that have been explicitly added to a new capabilities
      field. While here, also reorder the device info structures alphabetically.

      Signed-off-by: David E. Box <david.e.box@linux.intel.com>
      Reviewed-by: Andy Shevchenko <andy.shevchenko@gmail.com>
      Link: https://lore.kernel.org/r/20230316224628.2855884-1-david.e.box@linux.intel.com
      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 3d0f96d28e4ff9a61bf9ed83b0b6e608f7fc586a
  Author: Tom Rix <trix@redhat.com>
  Date:   Sat Mar 11 14:48:47 2023 +0100

      media: pvrusb2: VIDEO_PVRUSB2 depends on DVB_CORE to use dvb_* symbols

      [ Upstream commit 1107283b3351bef138cd12dbda1f999891cab7db ]

      A rand config causes this link error
      vmlinux.o: In function `pvr2_dvb_create':
      (.text+0x8af1d2): undefined reference to `dvb_register_adapter'

      The rand config has
      CONFIG_VIDEO_PVRUSB2=y
      CONFIG_VIDEO_DEV=y
      CONFIG_DVB_CORE=m

      VIDEO_PVRUSB2 should also depend on DVB_CORE.

      Signed-off-by: Tom Rix <trix@redhat.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 3715c5e9a8f96b6ed0dcbea06da443efccac1ecc
Author: harperchen <harperchen1110@gmail.com>
Date:   Fri Mar 3 16:30:11 2023 +0100

    media: pci: tw68: Fix null-ptr-deref bug in buf prepare and finish

    [ Upstream commit 1634b7adcc5bef645b3666fdd564e5952a9e24e0 ]

    When the driver calls tw68_risc_buffer() to prepare the buffer, the
    function call dma_alloc_coherent may fail, resulting in a empty buffer
    buf->cpu. Later when we free the buffer or access the buffer, null ptr
    deref is triggered.

    This bug is similar to the following one:
    https://git.linuxtv.org/media_stage.git/commit/?
id=2b064d91440b33fba5b452f2d1b31f13ae911d71.

    We believe the bug can be also dynamically triggered from user side.
    Similarly, we fix this by checking the return value of tw68_risc_buffer()
    and the value of buf->cpu before buffer free.

    Signed-off-by: harperchen <harperchen1110@gmail.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 5b8e5e28e85a546dfccc3895befe0e823fdd7c89
Author: harperchen <harperchen1110@gmail.com>
Date:   Thu Mar 2 13:39:05 2023 +0100

    media: cx23885: Fix a null-ptr-deref bug in buffer_prepare() and buffer_finish()

    [ Upstream commit 47e8b73bc35d7c54642f78e498697692f6358996 ]

    When the driver calls cx23885_risc_buffer() to prepare the buffer, the
    function call dma_alloc_coherent may fail, resulting in a empty buffer
    risc->cpu. Later when we free the buffer or access the buffer, null ptr
    deref is triggered.

    This bug is similar to the following one:
    https://git.linuxtv.org/media_stage.git/commit/?
id=2b064d91440b33fba5b452f2d1b31f13ae911d71.

    We believe the bug can be also dynamically triggered from user side.
    Similarly, we fix this by checking the return value of cx23885_risc_buffer()
    and the value of risc->cpu before buffer free.

    Signed-off-by: harperchen <harperchen1110@gmail.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 b707ff24821c96e89efce8edca37fdfa40f01991
Author: Kees Cook <keescook@chromium.org>
Date:   Sat Feb 4 19:38:05 2023 +0100

    media: imx-jpeg: Bounds check sizeimage access

    [ Upstream commit 474acc639fc8671fa4c1919d9e03253c82b6d321 ]

    The call of mxc_jpeg_get_plane_size() from mxc_jpeg_dec_irq() sets
    plane_no argument to 1. The compiler sees that it's possible to end up
    with an access beyond the bounds of sizeimage, if mem_planes was too
    large:
```

```
            if (plane_no >= fmt->mem_planes)        // mem_planes = 2+
                    return 0;

            if (fmt->mem_planes == fmt->comp_planes) // comp_planes != mem_planes
                    return q_data->sizeimage[plane_no];

            if (plane_no < fmt->mem_planes - 1)     // mem_planes = 2
                    return q_data->sizeimage[plane_no];
```

    comp_planes == 0 or 1 is safe. comp_planes > 2 would be out of bounds.

    (This isn't currently possible given the contents of mxc_formats, though.)

    Silence the warning by bounds checking comp_planes for future
    robustness. Seen with GCC 13:

```
    In function 'mxc_jpeg_get_plane_size',
        inlined from 'mxc_jpeg_dec_irq' at ../drivers/media/platform/nxp/imx-jpeg/mxc-
jpeg.c:729:14:
    ../drivers/media/platform/nxp/imx-jpeg/mxc-jpeg.c:641:42: warning: array subscript 2 is
above array bounds of 'u32[2]' {aka 'unsigned int[2]'} [-Warray-bounds=]
      641 |                 size += q_data->sizeimage[i];
          |                         ~~~~~~~~~~~~~~~~~^~~
    In file included from ../drivers/media/platform/nxp/imx-jpeg/mxc-jpeg-hw.h:112,
                     from ../drivers/media/platform/nxp/imx-jpeg/mxc-jpeg.c:63:
    ../drivers/media/platform/nxp/imx-jpeg/mxc-jpeg.h: In function 'mxc_jpeg_dec_irq':
    ../drivers/media/platform/nxp/imx-jpeg/mxc-jpeg.h:84:41: note: while referencing
'sizeimage'
       84 |         u32                             sizeimage[MXC_JPEG_MAX_PLANES];
          |                                         ^~~~~~~~~
```

    Cc: Mirela Rabulea <mirela.rabulea@nxp.com>
    Cc: NXP Linux Team <linux-imx@nxp.com>
    Cc: Shawn Guo <shawnguo@kernel.org>
    Cc: Sascha Hauer <s.hauer@pengutronix.de>
    Cc: Pengutronix Kernel Team <kernel@pengutronix.de>
    Cc: Fabio Estevam <festevam@gmail.com>
    Cc: linux-arm-kernel@lists.infradead.org
    Signed-off-by: Kees Cook <keescook@chromium.org>
    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 a5bcef9908f22bb8f529bf8c9c6888761cf93ab5
Author: Samson Tam <Samson.Tam@amd.com>
Date:   Tue Feb 28 14:33:00 2023 -0500

    drm/amd/display: reallocate DET for dual displays with high pixel rate ratio

    [ Upstream commit 5f3401eeb064fab5ce50728cce46532cce7a85c5 ]

    [Why]
    For dual displays where pixel rate is much higher on one display,
    we may get underflow when DET is evenly allocated.

    [How]
    Allocate less DET segments for the lower pixel rate display and
    more DET segments for the higher pixel rate display

    Reviewed-by: Alvin Lee <Alvin.Lee2@amd.com>
    Acked-by: Qingqing Zhuo <qingqing.zhuo@amd.com>
    Signed-off-by: Samson Tam <Samson.Tam@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 840de329ca99cafd0cdde9c6ac160b1330942aba
Author: Tomer Tayar <ttayar@habana.ai>
Date:   Wed Mar 1 17:45:58 2023 +0200

    accel/habanalabs: postpone mem_mgr IDR destruction to hpriv_release()

    [ Upstream commit 2e8e9a895c4589f124a37fc84d123b5114406e94 ]

    The memory manager IDR is currently destroyed when user releases the
    file descriptor.
    However, at this point the user context might be still held, and memory
    buffers might be still in use.
    Later on, calls to release those buffers will fail due to not finding
    their handles in the IDR, leading to a memory leak.
    To avoid this leak, split the IDR destruction from the memory manager
    fini, and postpone it to hpriv_release() when there is no user context
    and no buffers are used.

    Signed-off-by: Tomer Tayar <ttayar@habana.ai>
    Reviewed-by: Oded Gabbay <ogabbay@kernel.org>
    Signed-off-by: Oded Gabbay <ogabbay@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5524635e6618a8c4a19bedda66e2bab93edab7b8
Author: Konrad Dybcio <konrad.dybcio@linaro.org>
Date:   Thu Mar 2 02:18:49 2023 +0100

    arm64: dts: qcom: msm8996: Add missing DWC3 quirks

    [ Upstream commit d0af0537e28f6eace02deed63b585396de939213 ]

    Add missing dwc3 quirks from msm-3.18. Unfortunately, none of them
    make `dwc3-qcom 6af8800.usb: HS-PHY not in L2` go away.

    Signed-off-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230302011849.1873056-1-konrad.dybcio@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 331cd77f3d02c35f98b48d1aa934c54c4e7102c8
Author: Iuliana Prodan <iuliana.prodan@nxp.com>
Date:   Tue Feb 21 19:03:56 2023 +0200

    remoteproc: imx_dsp_rproc: Add custom memory copy implementation for i.MX DSP Cores

    [ Upstream commit 408ec1ff0caa340c57eecf4cbd14ef0132036a50 ]

    The IRAM is part of the HiFi DSP.
    According to hardware specification only 32-bits write are allowed
    otherwise we get a Kernel panic.

    Therefore add a custom memory copy and memset functions to deal with
    the above restriction.

    Signed-off-by: Iuliana Prodan <iuliana.prodan@nxp.com>
    Link: https://lore.kernel.org/r/20230221170356.27923-1-iuliana.prodan@oss.nxp.com
    Signed-off-by: Mathieu Poirier <mathieu.poirier@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 20467c6133d800b07542fb06999f6349830533b6
Author: Alexander Stein <alexander.stein@ew.tq-group.com>
Date:   Mon Mar 13 08:18:11 2023 +0100
```

```
    regmap: cache: Return error in cache sync operations for REGCACHE_NONE

    [ Upstream commit fd883d79e4dcd2417c2b80756f22a2ff03b0f6e0 ]

    There is no sense in doing a cache sync on REGCACHE_NONE regmaps.
    Instead of panicking the kernel due to missing cache_ops, return an error
    to client driver.

    Signed-off-by: Alexander Stein <alexander.stein@ew.tq-group.com>
    Link: https://lore.kernel.org/r/20230313071812.13577-1-alexander.stein@ew.tq-group.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4eae57359e55f61df8206c21771a147ea6adc9f6
Author: Rodrigo Siqueira <Rodrigo.Siqueira@amd.com>
Date:   Tue Nov 1 10:20:09 2022 -0400

    drm/amd/display: Use DC_LOG_DC in the trasform pixel function

    [ Upstream commit 7222f5841ff49709ca666b05ff336776e0664a20 ]

    [Why & How]
    DC now uses a new commit sequence which is more robust since it
    addresses cases where we need to reorganize pipes based on planes and
    other parameters. As a result, this new commit sequence reset the DC
    state by cleaning plane states and re-creating them accordingly with the
    need. For this reason, the dce_transform_set_pixel_storage_depth can be
    invoked after a plane state is destroyed and before its re-creation. In
    this situation and on DCE devices, DC will hit a condition that will
    trigger a dmesg log that looks like this:

    Console: switching to colour frame buffer device 240x67
    ------------[ cut here ]------------
    [..]
    Hardware name: System manufacturer System Product Name/PRIME X370-PRO, BIOS 5603
07/28/2020
    RIP: 0010:dce_transform_set_pixel_storage_depth+0x3f8/0x480 [amdgpu]
    [..]
    RSP: 0018:ffffc9000202b850 EFLAGS: 00010293
    RAX: ffffffffa081d100 RBX: ffff888110790000 RCX: 000000000000000c
    RDX: ffff888100bedbf8 RSI: 0000000000001a50 RDI: ffff88810463c900
    RBP: 0000000000000000 R08: 0000000000000000 R09: 0000000000000007
    R10: 0000000000000001 R11: 0000000000000f00 R12: ffff88810f500010
    R13: ffff888100bedbf8 R14: ffff88810f515688 R15: 0000000000000000
    FS:  00007ff0159249c0(0000) GS:ffff88840e940000(0000) knlGS:0000000000000000
    CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    CR2: 00007ff01528e550 CR3: 0000000002a10000 CR4: 00000000003506e0
    Call Trace:
     <TASK>
     ? dm_write_reg_func+0x21/0x80 [amdgpu 340dadd3f7c8cf4be11cf0bdc850245e99abe0e8]
     dc_stream_set_dither_option+0xfb/0x130 [amdgpu
340dadd3f7c8cf4be11cf0bdc850245e99abe0e8]
     amdgpu_dm_crtc_configure_crc_source+0x10b/0x190 [amdgpu
340dadd3f7c8cf4be11cf0bdc850245e99abe0e8]
     amdgpu_dm_atomic_commit_tail+0x20a8/0x2a90 [amdgpu
340dadd3f7c8cf4be11cf0bdc850245e99abe0e8]
     ? free_unref_page_commit+0x98/0x170
     ? free_unref_page+0xcc/0x150
     commit_tail+0x94/0x120
     drm_atomic_helper_commit+0x10f/0x140
     drm_atomic_commit+0x94/0xc0
     ? drm_plane_get_damage_clips.cold+0x1c/0x1c
     drm_client_modeset_commit_atomic+0x203/0x250
     drm_client_modeset_commit_locked+0x56/0x150
     drm_client_modeset_commit+0x21/0x40
```

```
       drm_fb_helper_lastclose+0x42/0x70
       amdgpu_driver_lastclose_kms+0xa/0x10 [amdgpu 340dadd3f7c8cf4be11cf0bdc850245e99abe0e8]
       drm_release+0xda/0x110
       __fput+0x89/0x240
       task_work_run+0x5c/0x90
       do_exit+0x333/0xae0
       do_group_exit+0x2d/0x90
       __x64_sys_exit_group+0x14/0x20
       do_syscall_64+0x5b/0x80
       ? exit_to_user_mode_prepare+0x1e/0x140
       entry_SYSCALL_64_after_hwframe+0x44/0xae
     RIP: 0033:0x7ff016ceaca1
     Code: Unable to access opcode bytes at RIP 0x7ff016ceac77.
     RSP: 002b:00007ffe7a2357e8 EFLAGS: 00000246 ORIG_RAX: 00000000000000e7
     RAX: ffffffffffffffda RBX: 00007ff016e15a00 RCX: 00007ff016ceaca1
     RDX: 000000000000003c RSI: 00000000000000e7 RDI: 0000000000000000
     RBP: 0000000000000000 R08: ffffffffffffff78 R09: 0000000000000000
     R10: 0000000000000000 R11: 0000000000000246 R12: 00007ff016e15a00
     R13: 0000000000000000 R14: 00007ff016e1aee8 R15: 00007ff016e1af00
       </TASK>

     Since this issue only happens in a transition state on DC, this commit
     replace BREAK_TO_DEBUGGER with DC_LOG_DC.

     Reviewed-by: Harry Wentland <Harry.Wentland@amd.com>
     Acked-by: Qingqing Zhuo <qingqing.zhuo@amd.com>
     Signed-off-by: Rodrigo Siqueira <Rodrigo.Siqueira@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 8f7b49c7a91a34c6ec7756595e4d62c57b310b2a
  Author: Gabe Teeger <gabe.teeger@amd.com>
  Date:   Thu Feb 23 11:30:31 2023 -0500

     drm/amd/display: Enable HostVM based on rIOMMU active

     [ Upstream commit 97fa4dfa66fdd52ad3d0c9fadeaaa1e87605bac7 ]

     [Why]
     There is underflow and flickering occuring. The
     underflow stops when hostvm is forced to active.
     According to policy, hostvm should be enabled if riommu
     is active, but this is not taken into account when
     deciding whether to enable hostvm.

     [What]
     For DCN314, set hostvm to true if riommu is active.

     Reviewed-by: Nicholas Kazlauskas <Nicholas.Kazlauskas@amd.com>
     Acked-by: Qingqing Zhuo <qingqing.zhuo@amd.com>
     Signed-off-by: Gabe Teeger <gabe.teeger@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 1d62eb772282419ca2bb818ac3dd839d04507cce
  Author: Hans de Goede <hdegoede@redhat.com>
  Date:   Mon Feb 13 21:35:53 2023 +0100

     platform/x86: x86-android-tablets: Add Acer Iconia One 7 B1-750 data

     [ Upstream commit 2f0cf1e85ddb5ae17284050dc1adafb89e4f1d8f ]

     The Acer Iconia One 7 B1-750 is a x86 ACPI tablet which ships with Android
```

```
    x86 as factory OS. Its DSDT contains a bunch of I2C devices which are not
    actually there, causing various resource conflicts. Enumeration of these
    is skipped through the acpi_quirk_skip_i2c_client_enumeration().

    Add support for manually instantiating the I2C + other devices which are
    actually present on this tablet by adding the necessary device info to
    the x86-android-tablets module.

    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Reviewed-by: Andy Shevchenko <andy.shevchenko@gmail.com>
    Link: https://lore.kernel.org/r/20230301092331.7038-2-hdegoede@redhat.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ef6c86d067b43a51905910c8ab839bef658d25e4
Author: Samson Tam <samson.tam@amd.com>
Date:   Fri Jan 27 18:30:08 2023 -0500

    drm/amd/display: enable DPG when disabling plane for phantom pipe

    [ Upstream commit f3f8f16b10f8258f1836e1110099097490a1d6c1 ]

    [Why]
    In disable_dangling_plane, for phantom pipes, we enable OTG so
    disable programming gets the double buffer update.  But this
    causes an underflow to occur.

    [How]
    Enable DPG prior to enabling OTG.

    Reviewed-by: Alvin Lee <Alvin.Lee2@amd.com>
    Acked-by: Qingqing Zhuo <qingqing.zhuo@amd.com>
    Signed-off-by: Samson Tam <samson.tam@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 5ccbd6ead295bcd7eaaf30b773cf039b573300fa
Author: Paul Hsieh <Paul.Hsieh@amd.com>
Date:   Fri Feb 10 12:00:16 2023 +0800

    drm/amd/display: Correct DML calculation to align HW formula

    [ Upstream commit 26a9f53198c955b15161da48cdb51041a38d5325 ]

    [Why]
    In 2560x1440@240p eDP panel, some use cases will enable MPC
    combine with RGB MPO then underflow happened. This case is
    not allowed from HW formula.

    [How]
    Correct eDP, DP and DP2 output bpp calculation to align HW
    formula.

    Reviewed-by: Nicholas Kazlauskas <Nicholas.Kazlauskas@amd.com>
    Acked-by: Qingqing Zhuo <qingqing.zhuo@amd.com>
    Signed-off-by: Paul Hsieh <Paul.Hsieh@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 375d192eb1f1d9229a6d994da7ba31f3582b106b
Author: Ayush Gupta <ayush.gupta@amd.com>
Date:   Fri Feb 10 13:02:09 2023 -0500

    drm/amd/display: populate subvp cmd info only for the top pipe
```

```
    [ Upstream commit 9bb10b7aaec3b6278f9cc410c17dcaa129bbbbf0 ]

    [Why]
    System restart observed while changing the display resolution
    to 8k with extended mode. Sytem restart was caused by a page fault.

    [How]
    When the driver populates subvp info it did it for both the pipes using
    vblank which caused an outof bounds array access causing the page fault.
    added checks to allow the top pipe only to fix this issue.

    Co-authored-by: Ayush Gupta <ayush.gupta@amd.com>
    Reviewed-by: Alvin Lee <Alvin.Lee2@amd.com>
    Acked-by: Qingqing Zhuo <qingqing.zhuo@amd.com>
    Signed-off-by: Ayush Gupta <ayush.gupta@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 5c73d85c34ab10dc4e0d7c6a5072b3a4e01ad3a0
Author: Jani Nikula <jani.nikula@intel.com>
Date:   Thu Feb 16 22:44:58 2023 +0200

    drm/displayid: add displayid_get_header() and check bounds better

    [ Upstream commit 5bacecc3c56131c31f18b23d366f2184328fd9cf ]

    Add a helper to get a pointer to struct displayid_header. To be
    pedantic, add buffer overflow checks to not touch the base if that
    itself would overflow.

    Cc: Iaroslav Boliukin <iam@lach.pw>
    Cc: Dmitry Osipenko <dmitry.osipenko@collabora.com>
    Signed-off-by: Jani Nikula <jani.nikula@intel.com>
    Tested-by: Dmitry Osipenko <dmitry.osipenko@collabora.com>
    Reviewed-by: Dmitry Osipenko <dmitry.osipenko@collabora.com>
    Signed-off-by: Dmitry Osipenko <dmitry.osipenko@collabora.com>
    Link:
https://patchwork.freedesktop.org/patch/msgid/4a03b3a5132642d3cdb6d4c2641422955a917292.16765
80180.git.jani.nikula@intel.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3a9d68d84b2e41ba3f2a727b36f035fad6800492
Author: Tetsuo Handa <penguin-kernel@I-love.SAKURA.ne.jp>
Date:   Tue Apr 11 19:57:33 2023 +0900

    fs: hfsplus: remove WARN_ON() from hfsplus_cat_{read,write}_inode()

    [ Upstream commit 81b21c0f0138ff5a499eafc3eb0578ad2a99622c ]

    syzbot is hitting WARN_ON() in hfsplus_cat_{read,write}_inode(), for
    crafted filesystem image can contain bogus length. There conditions are
    not kernel bugs that can justify kernel to panic.

    Reported-by: syzbot <syzbot+e2787430e752a92b8750@syzkaller.appspotmail.com>
    Link: https://syzkaller.appspot.com/bug?extid=e2787430e752a92b8750
    Reported-by: syzbot <syzbot+4913dca2ea6e4d43f3f1@syzkaller.appspotmail.com>
    Link: https://syzkaller.appspot.com/bug?extid=4913dca2ea6e4d43f3f1
    Signed-off-by: Tetsuo Handa <penguin-kernel@I-love.SAKURA.ne.jp>
    Reviewed-by: Viacheslav Dubeyko <slava@dubeyko.com>
    Message-Id: <15308173-5252-d6a3-ae3b-e96d46cb6f41@I-love.SAKURA.ne.jp>
    Signed-off-by: Christian Brauner <brauner@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 055e09295c8cb7f28d1e6ef9bbade75cc3e328b2
Author: Christian Brauner <brauner@kernel.org>
Date:   Tue Mar 21 09:18:07 2023 +0100

    open: return EINVAL for O_DIRECTORY | O_CREAT

    [ Upstream commit 43b450632676fb60e9faeddff285d9fac94a4f58 ]

    After a couple of years and multiple LTS releases we received a report
    that the behavior of O_DIRECTORY | O_CREAT changed starting with v5.7.

    On kernels prior to v5.7 combinations of O_DIRECTORY, O_CREAT, O_EXCL
    had the following semantics:

    (1) open("/tmp/d", O_DIRECTORY | O_CREAT)
        * d doesn't exist:                create regular file
        * d exists and is a regular file: ENOTDIR
        * d exists and is a directory:    EISDIR

    (2) open("/tmp/d", O_DIRECTORY | O_CREAT | O_EXCL)
        * d doesn't exist:                create regular file
        * d exists and is a regular file: EEXIST
        * d exists and is a directory:    EEXIST

    (3) open("/tmp/d", O_DIRECTORY | O_EXCL)
        * d doesn't exist:                ENOENT
        * d exists and is a regular file: ENOTDIR
        * d exists and is a directory:    open directory

    On kernels since to v5.7 combinations of O_DIRECTORY, O_CREAT, O_EXCL
    have the following semantics:

    (1) open("/tmp/d", O_DIRECTORY | O_CREAT)
        * d doesn't exist:                ENOTDIR (create regular file)
        * d exists and is a regular file: ENOTDIR
        * d exists and is a directory:    EISDIR

    (2) open("/tmp/d", O_DIRECTORY | O_CREAT | O_EXCL)
        * d doesn't exist:                ENOTDIR (create regular file)
        * d exists and is a regular file: EEXIST
        * d exists and is a directory:    EEXIST

    (3) open("/tmp/d", O_DIRECTORY | O_EXCL)
        * d doesn't exist:                ENOENT
        * d exists and is a regular file: ENOTDIR
        * d exists and is a directory:    open directory

    This is a fairly substantial semantic change that userspace didn't
    notice until Pedro took the time to deliberately figure out corner
    cases. Since no one noticed this breakage we can somewhat safely assume
    that O_DIRECTORY | O_CREAT combinations are likely unused.

    The v5.7 breakage is especially weird because while ENOTDIR is returned
    indicating failure a regular file is actually created. This doesn't make
    a lot of sense.

    Time was spent finding potential users of this combination. Searching on
    codesearch.debian.net showed that codebases often express semantical
    expectations about O_DIRECTORY | O_CREAT which are completely contrary
    to what our code has done and currently does.

    The expectation often is that this particular combination would create
    and open a directory. This suggests users who tried to use that
    combination would stumble upon the counterintuitive behavior no matter
    if pre-v5.7 or post v5.7 and quickly realize neither semantics give them
```

```
    what they want. For some examples see the code examples in [1] to [3]
    and the discussion in [4].

    There are various ways to address this issue. The lazy/simple option
    would be to restore the pre-v5.7 behavior and to just live with that bug
    forever. But since there's a real chance that the O_DIRECTORY | O_CREAT
    quirk isn't relied upon we should try to get away with murder(ing bad
    semantics) first. If we need to Frankenstein pre-v5.7 behavior later so
    be it.

    So let's simply return EINVAL categorically for O_DIRECTORY | O_CREAT
    combinations. In addition to cleaning up the old bug this also opens up
    the possiblity to make that flag combination do something more intuitive
    in the future.

    Starting with this commit the following semantics apply:

    (1) open("/tmp/d", O_DIRECTORY | O_CREAT)
        * d doesn't exist:                EINVAL
        * d exists and is a regular file: EINVAL
        * d exists and is a directory:    EINVAL

    (2) open("/tmp/d", O_DIRECTORY | O_CREAT | O_EXCL)
        * d doesn't exist:                EINVAL
        * d exists and is a regular file: EINVAL
        * d exists and is a directory:    EINVAL

    (3) open("/tmp/d", O_DIRECTORY | O_EXCL)
        * d doesn't exist:                ENOENT
        * d exists and is a regular file: ENOTDIR
        * d exists and is a directory:    open directory

    One additional note, O_TMPFILE is implemented as:

        #define __O_TMPFILE    020000000
        #define O_TMPFILE      (__O_TMPFILE | O_DIRECTORY)
        #define O_TMPFILE_MASK (__O_TMPFILE | O_DIRECTORY | O_CREAT)

    For older kernels it was important to return an explicit error when
    O_TMPFILE wasn't supported. So O_TMPFILE requires that O_DIRECTORY is
    raised alongside __O_TMPFILE. It also enforced that O_CREAT wasn't
    specified. Since O_DIRECTORY | O_CREAT could be used to create a regular
    allowing that combination together with __O_TMPFILE would've meant that
    false positives were possible, i.e., that a regular file was created
    instead of a O_TMPFILE. This could've been used to trick userspace into
    thinking it operated on a O_TMPFILE when it wasn't.

    Now that we block O_DIRECTORY | O_CREAT completely the check for O_CREAT
    in the __O_TMPFILE branch via if ((flags & O_TMPFILE_MASK) != O_TMPFILE)
    can be dropped. Instead we can simply check verify that O_DIRECTORY is
    raised via if (!(flags & O_DIRECTORY)) and explain this in two comments.

    As Aleksa pointed out O_PATH is unaffected by this change since it
    always returned EINVAL if O_CREAT was specified - with or without
    O_DIRECTORY.

    Link: https://lore.kernel.org/lkml/20230320071442.172228-1-pedro.falcato@gmail.com
    Link: https://sources.debian.org/src/flatpak/1.14.4-1/subprojects/libglnx/glnx-dirfd.c/?
hl=324#L324 [1]
    Link: https://sources.debian.org/src/flatpak-builder/1.2.3-1/subprojects/libglnx/glnx-
shutil.c/?hl=251#L251 [2]
    Link: https://sources.debian.org/src/ostree/2022.7-2/libglnx/glnx-dirfd.c/?hl=324#L324
[3]
    Link: https://www.openwall.com/lists/oss-security/2014/11/26/14 [4]
    Reported-by: Pedro Falcato <pedro.falcato@gmail.com>
```

```
    Cc: Aleksa Sarai <cyphar@cyphar.com>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Christian Brauner <brauner@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2bc0ae94ef1f9ed322d8ee439de3239ea3632ab2
Author: Zqiang <qiang1.zhang@intel.com>
Date:   Sat Dec 24 13:25:53 2022 +0800

    rcu: Protect rcu_print_task_exp_stall() ->exp_tasks access

    [ Upstream commit 3c1566bca3f8349f12b75d0a2d5e4a20ad6262ec ]

    For kernels built with CONFIG_PREEMPT_RCU=y, the following scenario can
    result in a NULL-pointer dereference:

               CPU1                                            CPU2
    rcu_preempt_deferred_qs_irqrestore                rcu_print_task_exp_stall
      if (special.b.blocked)                            READ_ONCE(rnp->exp_tasks) != NULL
        raw_spin_lock_rcu_node
        np = rcu_next_node_entry(t, rnp)
        if (&t->rcu_node_entry == rnp->exp_tasks)
          WRITE_ONCE(rnp->exp_tasks, np)
          ....
          raw_spin_unlock_irqrestore_rcu_node
                                                          raw_spin_lock_irqsave_rcu_node
                                                          t = list_entry(rnp->exp_tasks->prev,
                                                              struct task_struct,
rcu_node_entry)
                                                          (if rnp->exp_tasks is NULL, this
                                                             will dereference a NULL pointer)

    The problem is that CPU2 accesses the rcu_node structure's->exp_tasks
    field without holding the rcu_node structure's ->lock and CPU2 did
    not observe CPU1's change to rcu_node structure's ->exp_tasks in time.
    Therefore, if CPU1 sets rcu_node structure's->exp_tasks pointer to NULL,
    then CPU2 might dereference that NULL pointer.

    This commit therefore holds the rcu_node structure's ->lock while
    accessing that structure's->exp_tasks field.

    [ paulmck: Apply Frederic Weisbecker feedback. ]

    Acked-by: Joel Fernandes (Google) <joel@joelfernandes.org>
    Signed-off-by: Zqiang <qiang1.zhang@intel.com>
    Signed-off-by: Paul E. McKenney <paulmck@kernel.org>
    Signed-off-by: Joel Fernandes (Google) <joel@joelfernandes.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3cbc52be67b443ef65c58893d5c93d858eea79b8
Author: Ivan Orlov <ivan.orlov0322@gmail.com>
Date:   Sun Feb 26 16:16:33 2023 +0300

    selftests: cgroup: Add 'malloc' failures checks in test_memcontrol

    [ Upstream commit c83f320e55a49abd90629f42a72897afd579e0de ]

    There are several 'malloc' calls in test_memcontrol, which can be
    unsuccessful. This patch will add 'malloc' failures checking to
    give more details about test's fail reasons and avoid possible
    undefined behavior during the future null dereference (like the
    one in alloc_anon_50M_check_swap function).

    Signed-off-by: Ivan Orlov <ivan.orlov0322@gmail.com>
    Reviewed-by: Muchun Song <songmuchun@bytedance.com>
```

```
    Acked-by: Shakeel Butt <shakeelb@google.com>
    Acked-by: Roman Gushchin <roman.gushchin@linux.dev>
    Signed-off-by: Shuah Khan <skhan@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e8f7a53bd65c45deb113a89f6613d0e76cd4fa75
Author: Paul E. McKenney <paulmck@kernel.org>
Date:   Tue Jan 31 16:12:18 2023 -0800

    refscale: Move shutdown from wait_event() to wait_event_idle()

    [ Upstream commit 6bc6e6b27524304aadb9c04611ddb1c84dd7617a ]

    The ref_scale_shutdown() kthread/function uses wait_event() to wait for
    the refscale test to complete.  However, although the read-side tests
    are normally extremely fast, there is no law against specifying a very
    large value for the refscale.loops module parameter or against having
    a slow read-side primitive.  Either way, this might well trigger the
    hung-task timeout.

    This commit therefore replaces those wait_event() calls with calls to
    wait_event_idle(), which do not trigger the hung-task timeout.

    Signed-off-by: Paul E. McKenney <paulmck@kernel.org>
    Signed-off-by: Boqun Feng <boqun.feng@gmail.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 31668cebf45adfb6283e465e641c4f5a21b07afa
Author: Theodore Ts'o <tytso@mit.edu>
Date:   Sat Apr 29 00:06:28 2023 -0400

    ext4: allow ext4_get_group_info() to fail

    [ Upstream commit 5354b2af34064a4579be8bc0e2f15a7b70f14b5f ]

    Previously, ext4_get_group_info() would treat an invalid group number
    as BUG(), since in theory it should never happen.  However, if a
    malicious attaker (or fuzzer) modifies the superblock via the block
    device while it is the file system is mounted, it is possible for
    s_first_data_block to get set to a very large number.  In that case,
    when calculating the block group of some block number (such as the
    starting block of a preallocation region), could result in an
    underflow and very large block group number.  Then the BUG_ON check in
    ext4_get_group_info() would fire, resutling in a denial of service
    attack that can be triggered by root or someone with write access to
    the block device.

    For a quality of implementation perspective, it's best that even if
    the system administrator does something that they shouldn't, that it
    will not trigger a BUG.  So instead of BUG'ing, ext4_get_group_info()
    will call ext4_error and return NULL.  We also add fallback code in
    all of the callers of ext4_get_group_info() that it might NULL.

    Also, since ext4_get_group_info() was already borderline to be an
    inline function, un-inline it.  The results in a next reduction of the
    compiled text size of ext4 by roughly 2k.

    Cc: stable@kernel.org
    Link: https://lore.kernel.org/r/20230430154311.579720-2-tytso@mit.edu
    Reported-by: syzbot+e2efa3efc15a1c9e95c3@syzkaller.appspotmail.com
    Link: https://syzkaller.appspot.com/bug?id=69b28112e098b070f639efb356393af3ffec4220
    Signed-off-by: Theodore Ts'o <tytso@mit.edu>
    Reviewed-by: Jan Kara <jack@suse.cz>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 5f1f6426d1a9239a730e3360f22200c79c1900a6
Author: Kemeng Shi <shikemeng@huaweicloud.com>
Date:   Sat Mar 4 01:21:02 2023 +0800

    ext4: allow to find by goal if EXT4_MB_HINT_GOAL_ONLY is set

    [ Upstream commit 01e4ca29451760b9ac10b4cdc231c52150842643 ]

    If EXT4_MB_HINT_GOAL_ONLY is set, ext4_mb_regular_allocator will only
    allocate blocks from ext4_mb_find_by_goal. Allow to find by goal in
    ext4_mb_find_by_goal if EXT4_MB_HINT_GOAL_ONLY is set or allocation
    with EXT4_MB_HINT_GOAL_ONLY set will always fail.

    EXT4_MB_HINT_GOAL_ONLY is not used at all, so the problem is not
    found for now.

    Signed-off-by: Kemeng Shi <shikemeng@huaweicloud.com>
    Reviewed-by: Ojaswin Mujoo <ojaswin@linux.ibm.com>
    Link: https://lore.kernel.org/r/20230303172120.3800725-3-shikemeng@huaweicloud.com
    Signed-off-by: Theodore Ts'o <tytso@mit.edu>
    Stable-dep-of: 5354b2af3406 ("ext4: allow ext4_get_group_info() to fail")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5570ac70264021c3a9b0c55e0e6d7a90f25cebbc
Author: Theodore Ts'o <tytso@mit.edu>
Date:   Fri May 5 21:02:30 2023 -0400

    ext4: don't clear SB_RDONLY when remounting r/w until quota is re-enabled

    [ Upstream commit a44be64bbecb15a452496f60db6eacfee2b59c79 ]

    When a file system currently mounted read/only is remounted
    read/write, if we clear the SB_RDONLY flag too early, before the quota
    is initialized, and there is another process/thread constantly
    attempting to create a directory, it's possible to trigger the

            WARN_ON_ONCE(dquot_initialize_needed(inode));

    in ext4_xattr_block_set(), with the following stack trace:

       WARNING: CPU: 0 PID: 5338 at fs/ext4/xattr.c:2141 ext4_xattr_block_set+0x2ef2/0x3680
       RIP: 0010:ext4_xattr_block_set+0x2ef2/0x3680 fs/ext4/xattr.c:2141
       Call Trace:
        ext4_xattr_set_handle+0xcd4/0x15c0 fs/ext4/xattr.c:2458
        ext4_initxattrs+0xa3/0x110 fs/ext4/xattr_security.c:44
        security_inode_init_security+0x2df/0x3f0 security/security.c:1147
        __ext4_new_inode+0x347e/0x43d0 fs/ext4/ialloc.c:1324
        ext4_mkdir+0x425/0xce0 fs/ext4/namei.c:2992
        vfs_mkdir+0x29d/0x450 fs/namei.c:4038
        do_mkdirat+0x264/0x520 fs/namei.c:4061
        __do_sys_mkdirat fs/namei.c:4076 [inline]
        __se_sys_mkdirat fs/namei.c:4074 [inline]
        __x64_sys_mkdirat+0x89/0xa0 fs/namei.c:4074

    Cc: stable@kernel.org
    Link: https://lore.kernel.org/r/20230506142419.984260-1-tytso@mit.edu
    Reported-by: syzbot+6385d7d3065524c5ca6d@syzkaller.appspotmail.com
    Link: https://syzkaller.appspot.com/bug?id=6513f6cb5cd6b5fc9f37e3bb70d273b94be9c34c
    Signed-off-by: Theodore Ts'o <tytso@mit.edu>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1bcc9855b9327ac368393e5810752462417eab6b
Author: Theodore Ts'o <tytso@mit.edu>
Date:   Thu Apr 27 22:49:34 2023 -0400
```

```
    ext4: reflect error codes from ext4_multi_mount_protect() to its callers

    [ Upstream commit 3b50d5018ed06a647bb26c44bb5ae74e59c903c7 ]

    This will allow more fine-grained errno codes to be returned by the
    mount system call.

    Cc: Andreas Dilger <adilger.kernel@dilger.ca>
    Signed-off-by: Theodore Ts'o <tytso@mit.edu>
    Stable-dep-of: a44be64bbecb ("ext4: don't clear SB_RDONLY when remounting r/w until
quota is re-enabled")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e1490b43ab3febea5e5e3e8822197c1da513f1f9
Author: Zongjie Li <u202112089@hust.edu.cn>
Date:   Tue May 9 19:27:26 2023 +0800

    fbdev: arcfb: Fix error handling in arcfb_probe()

    [ Upstream commit 5a6bef734247c7a8c19511664ff77634ab86f45b ]

    Smatch complains that:
    arcfb_probe() warn: 'irq' from request_irq() not released on lines: 587.

    Fix error handling in the arcfb_probe() function. If IO addresses are
    not provided or framebuffer registration fails, the code will jump to
    the err_addr or err_register_fb label to release resources.
    If IRQ request fails, previously allocated resources will be freed.

    Fixes: 1154ea7dcd8e ("[PATCH] Framebuffer driver for Arc LCD board")
    Signed-off-by: Zongjie Li <u202112089@hust.edu.cn>
    Reviewed-by: Dongliang Mu <dzm91@hust.edu.cn>
    Signed-off-by: Helge Deller <deller@gmx.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 307763460ba91206e616401a924ccc1dc50ed90f
Author: Jani Nikula <jani.nikula@intel.com>
Date:   Thu May 4 13:35:08 2023 +0300

    drm/i915: taint kernel when force probing unsupported devices

    [ Upstream commit 79c901c93562bdf1c84ce6c1b744fbbe4389a6eb ]

    For development and testing purposes, the i915.force_probe module
    parameter and DRM_I915_FORCE_PROBE kconfig option allow probing of
    devices that aren't supported by the driver.

    The i915.force_probe module parameter is "unsafe" and setting it taints
    the kernel. However, using the kconfig option does not.

    Always taint the kernel when force probing a device that is not
    supported.

    v2: Drop "depends on EXPERT" to avoid build breakage (kernel test robot)

    Fixes: 7ef5ef5cdead ("drm/i915: add force_probe module parameter to replace
alpha_support")
    Cc: Joonas Lahtinen <joonas.lahtinen@linux.intel.com>
    Cc: Rodrigo Vivi <rodrigo.vivi@intel.com>
    Cc: Tvrtko Ursulin <tvrtko.ursulin@linux.intel.com>
    Cc: Daniel Vetter <daniel@ffwll.ch>
    Cc: Dave Airlie <airlied@gmail.com>
    Acked-by: Daniel Vetter <daniel.vetter@ffwll.ch>
    Reviewed-by: Rodrigo Vivi <rodrigo.vivi@intel.com>
    Signed-off-by: Jani Nikula <jani.nikula@intel.com>
```

```
    Link: https://patchwork.freedesktop.org/patch/msgid/20230504103508.1818540-1-
jani.nikula@intel.com
    (cherry picked from commit 3312bb4ad09ca6423bd4a5b15a94588a8962fb8e)
    Signed-off-by: Joonas Lahtinen <joonas.lahtinen@linux.intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 221a8b052bad0ad6a7d8d42494478268d06f1a58
Author: Nikita Zhandarovich <n.zhandarovich@fintech.ru>
Date:   Tue Apr 18 07:04:30 2023 -0700

    drm/i915/dp: prevent potential div-by-zero

    [ Upstream commit 0ff80028e2702c7c3d78b69705dc47c1ccba8c39 ]

    drm_dp_dsc_sink_max_slice_count() may return 0 if something goes
    wrong on the part of the DSC sink and its DPCD register. This null
    value may be later used as a divisor in intel_dsc_compute_params(),
    which will lead to an error.
    In the unlikely event that this issue occurs, fix it by testing the
    return value of drm_dp_dsc_sink_max_slice_count() against zero.

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

    Fixes: a4a157777c80 ("drm/i915/dp: Compute DSC pipe config in atomic check")
    Signed-off-by: Nikita Zhandarovich <n.zhandarovich@fintech.ru>
    Reviewed-by: Rodrigo Vivi <rodrigo.vivi@intel.com>
    Signed-off-by: Rodrigo Vivi <rodrigo.vivi@intel.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20230418140430.69902-1-
n.zhandarovich@fintech.ru
    (cherry picked from commit 51f7008239de011370c5067bbba07f0207f06b72)
    Signed-off-by: Joonas Lahtinen <joonas.lahtinen@linux.intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8b3c0d2d1685ba40b0af4ee1f8d8824a73870f88
Author: Stanislav Lisovskiy <stanislav.lisovskiy@intel.com>
Date:   Fri May 5 11:22:12 2023 +0300

    drm/i915: Fix NULL ptr deref by checking new_crtc_state

    [ Upstream commit a41d985902c153c31c616fe183cf2ee331e95ecb ]

    intel_atomic_get_new_crtc_state can return NULL, unless crtc state wasn't
    obtained previously with intel_atomic_get_crtc_state, so we must check it
    for NULLness here, just as in many other places, where we can't guarantee
    that intel_atomic_get_crtc_state was called.
    We are currently getting NULL ptr deref because of that, so this fix was
    confirmed to help.

    Fixes: 74a75dc90869 ("drm/i915/display: move plane prepare/cleanup to
intel_atomic_plane.c")
    Signed-off-by: Stanislav Lisovskiy <stanislav.lisovskiy@intel.com>
    Reviewed-by: Andrzej Hajda <andrzej.hajda@intel.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20230505082212.27089-1-
stanislav.lisovskiy@intel.com
    (cherry picked from commit 1d5b09f8daf859247a1ea65b0d732a24d88980d8)
    Signed-off-by: Joonas Lahtinen <joonas.lahtinen@linux.intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c4aeba2226916aa4cebe707854cc258c94bd7474
Author: John Harrison <John.C.Harrison@Intel.com>
Date:   Fri Apr 28 11:56:33 2023 -0700

    drm/i915/guc: Don't capture Gen8 regs on Xe devices
```

```
    [ Upstream commit 275dac1f7f5e9c2a2e806b34d3b10804eec0ac3c ]

    A pair of pre-Xe registers were being included in the Xe capture list.
    GuC was rejecting those as being invalid and logging errors about
    them. So, stop doing it.

    Signed-off-by: John Harrison <John.C.Harrison@Intel.com>
    Reviewed-by: Alan Previn <alan.previn.teres.alexis@intel.com>
    Fixes: dce2bd542337 ("drm/i915/guc: Add Gen9 registers for GuC error state capture.")
    Cc: Alan Previn <alan.previn.teres.alexis@intel.com>
    Cc: Umesh Nerlige Ramappa <umesh.nerlige.ramappa@intel.com>
    Cc: Lucas De Marchi <lucas.demarchi@intel.com>
    Cc: John Harrison <John.C.Harrison@Intel.com>
    Cc: Jani Nikula <jani.nikula@intel.com>
    Cc: Matt Roper <matthew.d.roper@intel.com>
    Cc: Balasubramani Vivekanandan <balasubramani.vivekanandan@intel.com>
    Cc: Daniele Ceraolo Spurio <daniele.ceraolospurio@intel.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20230428185636.457407-2-
John.C.Harrison@Intel.com
    (cherry picked from commit b049132d61336f643d8faf2f6574b063667088cf)
    Signed-off-by: Joonas Lahtinen <joonas.lahtinen@linux.intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f237f79b63c9242450e6869adcd2c10445859f28
Author: Kuniyuki Iwashima <kuniyu@amazon.com>
Date:   Tue May 9 17:34:56 2023 -0700

    af_unix: Fix data races around sk->sk_shutdown.

    [ Upstream commit e1d09c2c2f5793474556b60f83900e088d0d366d ]

    KCSAN found a data race around sk->sk_shutdown where unix_release_sock()
    and unix_shutdown() update it under unix_state_lock(), OTOH unix_poll()
    and unix_dgram_poll() read it locklessly.

    We need to annotate the writes and reads with WRITE_ONCE() and READ_ONCE().

    BUG: KCSAN: data-race in unix_poll / unix_release_sock

    write to 0xffff88800d0f8aec of 1 bytes by task 264 on cpu 0:
     unix_release_sock+0x75c/0x910 net/unix/af_unix.c:631
     unix_release+0x59/0x80 net/unix/af_unix.c:1042
     __sock_release+0x7d/0x170 net/socket.c:653
     sock_close+0x19/0x30 net/socket.c:1397
     __fput+0x179/0x5e0 fs/file_table.c:321
     ____fput+0x15/0x20 fs/file_table.c:349
     task_work_run+0x116/0x1a0 kernel/task_work.c:179
     resume_user_mode_work include/linux/resume_user_mode.h:49 [inline]
     exit_to_user_mode_loop kernel/entry/common.c:171 [inline]
     exit_to_user_mode_prepare+0x174/0x180 kernel/entry/common.c:204
     __syscall_exit_to_user_mode_work kernel/entry/common.c:286 [inline]
     syscall_exit_to_user_mode+0x1a/0x30 kernel/entry/common.c:297
     do_syscall_64+0x4b/0x90 arch/x86/entry/common.c:86
     entry_SYSCALL_64_after_hwframe+0x72/0xdc

    read to 0xffff88800d0f8aec of 1 bytes by task 222 on cpu 1:
     unix_poll+0xa3/0x2a0 net/unix/af_unix.c:3170
     sock_poll+0xcf/0x2b0 net/socket.c:1385
     vfs_poll include/linux/poll.h:88 [inline]
     ep_item_poll.isra.0+0x78/0xc0 fs/eventpoll.c:855
     ep_send_events fs/eventpoll.c:1694 [inline]
     ep_poll fs/eventpoll.c:1823 [inline]
     do_epoll_wait+0x6c4/0xea0 fs/eventpoll.c:2258
     __do_sys_epoll_wait fs/eventpoll.c:2270 [inline]
     __se_sys_epoll_wait fs/eventpoll.c:2265 [inline]
```

```
      __x64_sys_epoll_wait+0xcc/0x190 fs/eventpoll.c:2265
      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: 0x00 -> 0x03

    Reported by Kernel Concurrency Sanitizer on:
    CPU: 1 PID: 222 Comm: dbus-broker Not tainted 6.3.0-rc7-02330-gca6270c12e20 #2
    Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS rel-1.16.0-0-gd239552ce722-
prebuilt.qemu.org 04/01/2014

    Fixes: 3c73419c09a5 ("af_unix: fix 'poll for write'/ connected DGRAM sockets")
    Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
    Reported-by: syzbot <syzkaller@googlegroups.com>
    Signed-off-by: Kuniyuki Iwashima <kuniyu@amazon.com>
    Reviewed-by: Eric Dumazet <edumazet@google.com>
    Reviewed-by: Michal Kubiak <michal.kubiak@intel.com>
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e3eeefdca32dcaaca1e81254af84c789debfc08a
Author: Kuniyuki Iwashima <kuniyu@amazon.com>
Date:   Tue May 9 17:34:55 2023 -0700

    af_unix: Fix a data race of sk->sk_receive_queue->qlen.

    [ Upstream commit 679ed006d416ea0cecfe24a99d365d1dea69c683 ]

    KCSAN found a data race of sk->sk_receive_queue->qlen where recvmsg()
    updates qlen under the queue lock and sendmsg() checks qlen under
    unix_state_sock(), not the queue lock, so the reader side needs
    READ_ONCE().

    BUG: KCSAN: data-race in __skb_try_recv_from_queue / unix_wait_for_peer

    write (marked) to 0xffff888019fe7c68 of 4 bytes by task 49792 on cpu 0:
     __skb_unlink include/linux/skbuff.h:2347 [inline]
     __skb_try_recv_from_queue+0x3de/0x470 net/core/datagram.c:197
     __skb_try_recv_datagram+0xf7/0x390 net/core/datagram.c:263
     __unix_dgram_recvmsg+0x109/0x8a0 net/unix/af_unix.c:2452
     unix_dgram_recvmsg+0x94/0xa0 net/unix/af_unix.c:2549
     sock_recvmsg_nosec net/socket.c:1019 [inline]
     ____sys_recvmsg+0x3a3/0x3b0 net/socket.c:2720
     ___sys_recvmsg+0xc8/0x150 net/socket.c:2764
     do_recvmmsg+0x182/0x560 net/socket.c:2858
     __sys_recvmmsg net/socket.c:2937 [inline]
     __do_sys_recvmmsg net/socket.c:2960 [inline]
     __se_sys_recvmmsg net/socket.c:2953 [inline]
     __x64_sys_recvmmsg+0x153/0x170 net/socket.c:2953
     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 0xffff888019fe7c68 of 4 bytes by task 49793 on cpu 1:
     skb_queue_len include/linux/skbuff.h:2127 [inline]
     unix_recvq_full net/unix/af_unix.c:229 [inline]
     unix_wait_for_peer+0x154/0x1a0 net/unix/af_unix.c:1445
     unix_dgram_sendmsg+0x13bc/0x14b0 net/unix/af_unix.c:2048
     sock_sendmsg_nosec net/socket.c:724 [inline]
     sock_sendmsg+0x148/0x160 net/socket.c:747
     ____sys_sendmsg+0x20e/0x620 net/socket.c:2503
     ___sys_sendmsg+0xc6/0x140 net/socket.c:2557
     __sys_sendmmsg+0x11d/0x370 net/socket.c:2643
     __do_sys_sendmmsg net/socket.c:2672 [inline]
```

```
      __se_sys_sendmmsg net/socket.c:2669 [inline]
      __x64_sys_sendmmsg+0x58/0x70 net/socket.c:2669
      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: 0x0000000b -> 0x00000001

    Reported by Kernel Concurrency Sanitizer on:
    CPU: 1 PID: 49793 Comm: syz-executor.0 Not tainted 6.3.0-rc7-02330-gca6270c12e20 #2
    Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS rel-1.16.0-0-gd239552ce722-
prebuilt.qemu.org 04/01/2014

    Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
    Reported-by: syzbot <syzkaller@googlegroups.com>
    Signed-off-by: Kuniyuki Iwashima <kuniyu@amazon.com>
    Reviewed-by: Eric Dumazet <edumazet@google.com>
    Reviewed-by: Michal Kubiak <michal.kubiak@intel.com>
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 30ba51797dd3d159007f790e0afd041b535e5d18
Author: Eric Dumazet <edumazet@google.com>
Date:   Tue May 9 17:31:31 2023 +0000

    net: datagram: fix data-races in datagram_poll()

    [ Upstream commit 5bca1d081f44c9443e61841842ce4e9179d327b6 ]

    datagram_poll() runs locklessly, we should add READ_ONCE()
    annotations while reading sk->sk_err, sk->sk_shutdown and sk->sk_state.

    Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
    Signed-off-by: Eric Dumazet <edumazet@google.com>
    Reviewed-by: Kuniyuki Iwashima <kuniyu@amazon.com>
    Link: https://lore.kernel.org/r/20230509173131.3263780-1-edumazet@google.com
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c43a96fc00b662cef1ef0eb22d40441ce2abae8f
Author: Vitaly Prosyak <vitaly.prosyak@amd.com>
Date:   Wed May 10 09:51:11 2023 -0400

    drm/sched: Check scheduler work queue before calling timeout handling

    [ Upstream commit 2da5bffe9eaa5819a868e8eaaa11b3fd0f16a691 ]

    During an IGT GPU reset test we see again oops despite of
    commit 0c8c901aaaebc9 (drm/sched: Check scheduler ready before calling
    timeout handling).

    It uses ready condition whether to call drm_sched_fault which unwind
    the TDR leads to GPU reset.
    However it looks the ready condition is overloaded with other meanings,
    for example, for the following stack is related GPU reset :

    0  gfx_v9_0_cp_gfx_start
    1  gfx_v9_0_cp_gfx_resume
    2  gfx_v9_0_cp_resume
    3  gfx_v9_0_hw_init
    4  gfx_v9_0_resume
    5  amdgpu_device_ip_resume_phase2

    does the following:
            /* start the ring */
```

```
             gfx_v9_0_cp_gfx_start(adev);
             ring->sched.ready = true;

    The same approach is for other ASICs as well :
    gfx_v8_0_cp_gfx_resume
    gfx_v10_0_kiq_resume, etc...

    As a result, our GPU reset test causes GPU fault which calls unconditionally
gfx_v9_0_fault
    and then drm_sched_fault. However now it depends on whether the interrupt service
routine
    drm_sched_fault is executed after gfx_v9_0_cp_gfx_start is completed which sets the
ready
    field of the scheduler to true even  for uninitialized schedulers and causes oops vs
    no fault or when ISR  drm_sched_fault is completed prior  gfx_v9_0_cp_gfx_start and
    NULL pointer dereference does not occur.

    Use the field timeout_wq  to prevent oops for uninitialized schedulers.
    The field could be initialized by the work queue of resetting the domain.

    v1: Corrections to commit message (Luben)

    Fixes: 11b3b9f461c5c4 ("drm/sched: Check scheduler ready before calling timeout
handling")
    Signed-off-by: Vitaly Prosyak <vitaly.prosyak@amd.com>
    Link: https://lore.kernel.org/r/20230510135111.58631-1-vitaly.prosyak@amd.com
    Reviewed-by: Luben Tuikov <luben.tuikov@amd.com>
    Signed-off-by: Luben Tuikov <luben.tuikov@amd.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0f4024d5ede69f9a49b737df967f43f2c027dd41
Author: Colin Foster <colin.foster@in-advantage.com>
Date:   Tue May 9 21:48:51 2023 -0700

    net: mscc: ocelot: fix stat counter register values

    [ Upstream commit cdc2e28e214fe9315cdd7e069c1c8e2428f93427 ]

    Commit d4c367650704 ("net: mscc: ocelot: keep ocelot_stat_layout by reg
    address, not offset") organized the stats counters for Ocelot chips, namely
    the VSC7512 and VSC7514. A few of the counter offsets were incorrect, and
    were caught by this warning:

    WARNING: CPU: 0 PID: 24 at drivers/net/ethernet/mscc/ocelot_stats.c:909
    ocelot_stats_init+0x1fc/0x2d8
    reg 0x5000078 had address 0x220 but reg 0x5000079 has address 0x214,
    bulking broken!

    Fix these register offsets.

    Fixes: d4c367650704 ("net: mscc: ocelot: keep ocelot_stat_layout by reg address, not
offset")
    Signed-off-by: Colin Foster <colin.foster@in-advantage.com>
    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 3cd16c6a6a6b68bba02fbbc54b9906f44640ffde
Author: t.feng <fengtao40@huawei.com>
Date:   Wed May 10 11:50:44 2023 +0800

    ipvlan:Fix out-of-bounds caused by unclear skb->cb

    [ Upstream commit 90cbed5247439a966b645b34eb0a2e037836ea8e ]
```

```
    If skb enqueue the qdisc, fq_skb_cb(skb)->time_to_send is changed which
    is actually skb->cb, and IPCB(skb_in)->opt will be used in
    __ip_options_echo. It is possible that memcpy is out of bounds and lead
    to stack overflow.
    We should clear skb->cb before ip_local_out or ip6_local_out.

    v2:
    1. clean the stack info
    2. use IPCB/IP6CB instead of skb->cb

    crash on stable-5.10(reproduce in kasan kernel).
    Stack info:
    [ 2203.651571] BUG: KASAN: stack-out-of-bounds in
    __ip_options_echo+0x589/0x800
    [ 2203.653327] Write of size 4 at addr ffff88811a388f27 by task
    swapper/3/0
    [ 2203.655460] CPU: 3 PID: 0 Comm: swapper/3 Kdump: loaded Not tainted
    5.10.0-60.18.0.50.h856.kasan.eulerosv2r11.x86_64 #1
    [ 2203.655466] Hardware name: QEMU Standard PC (i440FX + PIIX, 1996),
    BIOS rel-1.10.2-0-g5f4c7b1-20181220_000000-szxrtosci10000 04/01/2014
    [ 2203.655475] Call Trace:
    [ 2203.655481]  <IRQ>
    [ 2203.655501]  dump_stack+0x9c/0xd3
    [ 2203.655514]  print_address_description.constprop.0+0x19/0x170
    [ 2203.655530]  __kasan_report.cold+0x6c/0x84
    [ 2203.655586]  kasan_report+0x3a/0x50
    [ 2203.655594]  check_memory_region+0xfd/0x1f0
    [ 2203.655601]  memcpy+0x39/0x60
    [ 2203.655608]  __ip_options_echo+0x589/0x800
    [ 2203.655654]  __icmp_send+0x59a/0x960
    [ 2203.655755]  nf_send_unreach+0x129/0x3d0 [nf_reject_ipv4]
    [ 2203.655763]  reject_tg+0x77/0x1bf [ipt_REJECT]
    [ 2203.655772]  ipt_do_table+0x691/0xa40 [ip_tables]
    [ 2203.655821]  nf_hook_slow+0x69/0x100
    [ 2203.655828]  __ip_local_out+0x21e/0x2b0
    [ 2203.655857]  ip_local_out+0x28/0x90
    [ 2203.655868]  ipvlan_process_v4_outbound+0x21e/0x260 [ipvlan]
    [ 2203.655931]  ipvlan_xmit_mode_l3+0x3bd/0x400 [ipvlan]
    [ 2203.655967]  ipvlan_queue_xmit+0xb3/0x190 [ipvlan]
    [ 2203.655977]  ipvlan_start_xmit+0x2e/0xb0 [ipvlan]
    [ 2203.655984]  xmit_one.constprop.0+0xe1/0x280
    [ 2203.655992]  dev_hard_start_xmit+0x62/0x100
    [ 2203.656000]  sch_direct_xmit+0x215/0x640
    [ 2203.656028]  __qdisc_run+0x153/0x1f0
    [ 2203.656069]  __dev_queue_xmit+0x77f/0x1030
    [ 2203.656173]  ip_finish_output2+0x59b/0xc20
    [ 2203.656244]  __ip_finish_output.part.0+0x318/0x3d0
    [ 2203.656312]  ip_finish_output+0x168/0x190
    [ 2203.656320]  ip_output+0x12d/0x220
    [ 2203.656357]  __ip_queue_xmit+0x392/0x880
    [ 2203.656380]  __tcp_transmit_skb+0x1088/0x11c0
    [ 2203.656436]  __tcp_retransmit_skb+0x475/0xa30
    [ 2203.656505]  tcp_retransmit_skb+0x2d/0x190
    [ 2203.656512]  tcp_retransmit_timer+0x3af/0x9a0
    [ 2203.656519]  tcp_write_timer_handler+0x3ba/0x510
    [ 2203.656529]  tcp_write_timer+0x55/0x180
    [ 2203.656542]  call_timer_fn+0x3f/0x1d0
    [ 2203.656555]  expire_timers+0x160/0x200
    [ 2203.656562]  run_timer_softirq+0x1f4/0x480
    [ 2203.656606]  __do_softirq+0xfd/0x402
    [ 2203.656613]  asm_call_irq_on_stack+0x12/0x20
    [ 2203.656617]  </IRQ>
    [ 2203.656623]  do_softirq_own_stack+0x37/0x50
    [ 2203.656631]  irq_exit_rcu+0x134/0x1a0
    [ 2203.656639]  sysvec_apic_timer_interrupt+0x36/0x80
```

```
[ 2203.656646]  asm_sysvec_apic_timer_interrupt+0x12/0x20
[ 2203.656654] RIP: 0010:default_idle+0x13/0x20
[ 2203.656663] Code: 89 f0 5d 41 5c 41 5d 41 5e c3 cc cc cc cc cc cc cc
cc cc cc cc cc cc 0f 1f 44 00 00 0f 1f 44 00 00 0f 00 2d 9f 32 57 00 fb
f4 <c3> cc cc cc cc 0f 1f 84 00 00 00 00 00 0f 1f 44 00 00 41 54 be 08
[ 2203.656668] RSP: 0018:ffff88810036fe78 EFLAGS: 00000256
[ 2203.656676] RAX: ffffffffaf2a87f0 RBX: ffff888100360000 RCX:
ffffffffaf290191
[ 2203.656681] RDX: 0000000000098b5e RSI: 0000000000000004 RDI:
ffff88811a3c4f60
[ 2203.656686] RBP: 0000000000000000 R08: 0000000000000001 R09:
ffff88811a3c4f63
[ 2203.656690] R10: ffffed10234789ec R11: 0000000000000001 R12:
0000000000000003
[ 2203.656695] R13: ffff888100360000 R14: 0000000000000000 R15:
0000000000000000
[ 2203.656729]  default_idle_call+0x5a/0x150
[ 2203.656735]  cpuidle_idle_call+0x1c6/0x220
[ 2203.656780]  do_idle+0xab/0x100
[ 2203.656786]  cpu_startup_entry+0x19/0x20
[ 2203.656793]  secondary_startup_64_no_verify+0xc2/0xcb

[ 2203.657409] The buggy address belongs to the page:
[ 2203.658648] page:0000000027a9842f refcount:1 mapcount:0
mapping:0000000000000000 index:0x0 pfn:0x11a388
[ 2203.658665] flags:
0x17ffffc0001000(reserved|node=0|zone=2|lastcpupid=0x1fffff)
[ 2203.658675] raw: 0017ffffc0001000 ffffea000468e208 ffffea000468e208
0000000000000000
[ 2203.658682] raw: 0000000000000000 0000000000000000 00000001ffffffff
0000000000000000
[ 2203.658686] page dumped because: kasan: bad access detected

To reproduce(ipvlan with IPVLAN_MODE_L3):
Env setting:
=========================================================
modprobe ipvlan ipvlan_default_mode=1
sysctl net.ipv4.conf.eth0.forwarding=1
iptables -t nat -A POSTROUTING -s 20.0.0.0/255.255.255.0 -o eth0 -j
MASQUERADE
ip link add gw link eth0 type ipvlan
ip -4 addr add 20.0.0.254/24 dev gw
ip netns add net1
ip link add ipv1 link eth0 type ipvlan
ip link set ipv1 netns net1
ip netns exec net1 ip link set ipv1 up
ip netns exec net1 ip -4 addr add 20.0.0.4/24 dev ipv1
ip netns exec net1 route add default gw 20.0.0.254
ip netns exec net1 tc qdisc add dev ipv1 root netem loss 10%
ifconfig gw up
iptables -t filter -A OUTPUT -p tcp --dport 8888 -j REJECT --reject-with
icmp-port-unreachable
=========================================================
And then excute the shell(curl any address of eth0 can reach):

for((i=1;i<=100000;i++))
do
        ip netns exec net1 curl x.x.x.x:8888
done
=========================================================

Fixes: 2ad7bf363841 ("ipvlan: Initial check-in of the IPVLAN driver.")
Signed-off-by: "t.feng" <fengtao40@huawei.com>
Suggested-by: Florian Westphal <fw@strlen.de>
Reviewed-by: Paolo Abeni <pabeni@redhat.com>
```

```
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c29389c9556243875223ffd482a928e90e629be9
Author: Ziwei Xiao <ziweixiao@google.com>
Date:   Tue May 9 15:51:23 2023 -0700

    gve: Remove the code of clearing PBA bit

    [ Upstream commit f4c2e67c1773d2a2632381ee30e9139c1e744c16 ]

    Clearing the PBA bit from the driver is race prone and it may lead to
    dropped interrupt events. This could potentially lead to the traffic
    being completely halted.

    Fixes: 5e8c5adf95f8 ("gve: DQO: Add core netdev features")
    Signed-off-by: Ziwei Xiao <ziweixiao@google.com>
    Signed-off-by: Bailey Forrest <bcf@google.com>
    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 d23691497c70db5306d4417a4809858faac1f40f
Author: Eric Dumazet <edumazet@google.com>
Date:   Tue May 9 20:36:56 2023 +0000

    tcp: add annotations around sk->sk_shutdown accesses

    [ Upstream commit e14cadfd80d76f01bfaa1a8d745b1db19b57d6be ]

    Now sk->sk_shutdown is no longer a bitfield, we can add
    standard READ_ONCE()/WRITE_ONCE() annotations to silence
    KCSAN reports like the following:

    BUG: KCSAN: data-race in tcp_disconnect / tcp_poll

    write to 0xffff88814588582c of 1 bytes by task 3404 on cpu 1:
    tcp_disconnect+0x4d6/0xdb0 net/ipv4/tcp.c:3121
    __inet_stream_connect+0x5dd/0x6e0 net/ipv4/af_inet.c:715
    inet_stream_connect+0x48/0x70 net/ipv4/af_inet.c:727
    __sys_connect_file net/socket.c:2001 [inline]
    __sys_connect+0x19b/0x1b0 net/socket.c:2018
    __do_sys_connect net/socket.c:2028 [inline]
    __se_sys_connect net/socket.c:2025 [inline]
    __x64_sys_connect+0x41/0x50 net/socket.c:2025
    do_syscall_x64 arch/x86/entry/common.c:50 [inline]
    do_syscall_64+0x41/0xc0 arch/x86/entry/common.c:80
    entry_SYSCALL_64_after_hwframe+0x63/0xcd

    read to 0xffff88814588582c of 1 bytes by task 3374 on cpu 0:
    tcp_poll+0x2e6/0x7d0 net/ipv4/tcp.c:562
    sock_poll+0x253/0x270 net/socket.c:1383
    vfs_poll include/linux/poll.h:88 [inline]
    io_poll_check_events io_uring/poll.c:281 [inline]
    io_poll_task_func+0x15a/0x820 io_uring/poll.c:333
    handle_tw_list io_uring/io_uring.c:1184 [inline]
    tctx_task_work+0x1fe/0x4d0 io_uring/io_uring.c:1246
    task_work_run+0x123/0x160 kernel/task_work.c:179
    get_signal+0xe64/0xff0 kernel/signal.c:2635
    arch_do_signal_or_restart+0x89/0x2a0 arch/x86/kernel/signal.c:306
    exit_to_user_mode_loop+0x6f/0xe0 kernel/entry/common.c:168
    exit_to_user_mode_prepare+0x6c/0xb0 kernel/entry/common.c:204
    __syscall_exit_to_user_mode_work kernel/entry/common.c:286 [inline]
    syscall_exit_to_user_mode+0x26/0x140 kernel/entry/common.c:297
    do_syscall_64+0x4d/0xc0 arch/x86/entry/common.c:86
```

```
    entry_SYSCALL_64_after_hwframe+0x63/0xcd

    value changed: 0x03 -> 0x00

    Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
    Reported-by: syzbot <syzkaller@googlegroups.com>
    Signed-off-by: Eric Dumazet <edumazet@google.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 15eaeb8941f12fcc2713c4bf6eb8f76a37854b4d
Author: Eric Dumazet <edumazet@google.com>
Date:   Tue May 9 13:18:57 2023 +0000

    net: add vlan_get_protocol_and_depth() helper

    [ Upstream commit 4063384ef762cc5946fc7a3f89879e76c6ec51e2 ]

    Before blamed commit, pskb_may_pull() was used instead
    of skb_header_pointer() in __vlan_get_protocol() and friends.

    Few callers depended on skb->head being populated with MAC header,
    syzbot caught one of them (skb_mac_gso_segment())

    Add vlan_get_protocol_and_depth() to make the intent clearer
    and use it where sensible.

    This is a more generic fix than commit e9d3f80935b6
    ("net/af_packet: make sure to pull mac header") which was
    dealing with a similar issue.

    kernel BUG at include/linux/skbuff.h:2655 !
    invalid opcode: 0000 [#1] SMP KASAN
    CPU: 0 PID: 1441 Comm: syz-executor199 Not tainted 6.1.24-syzkaller #0
    Hardware name: Google Google Compute Engine/Google Compute Engine, BIOS Google
04/14/2023
    RIP: 0010:__skb_pull include/linux/skbuff.h:2655 [inline]
    RIP: 0010:skb_mac_gso_segment+0x68f/0x6a0 net/core/gro.c:136
    Code: fd 48 8b 5c 24 10 44 89 6b 70 48 c7 c7 c0 ae 0d 86 44 89 e6 e8 a1 91 d0 00 48 c7
c7 00 af 0d 86 48 89 de 31 d2 e8 d1 4a e9 ff <0f> 0b 66 2e 0f 1f 84 00 00 00 00 00 0f 1f 44
00 00 55 48 89 e5 41
    RSP: 0018:ffffc90001bd7520 EFLAGS: 00010286
    RAX: ffffffff8469736a RBX: ffff88810f31dac0 RCX: ffff888115a18b00
    RDX: 0000000000000000 RSI: 0000000000000000 RDI: 0000000000000000
    RBP: ffffc90001bd75e8 R08: ffffffff84697183 R09: fffff5200037adf9
    R10: 0000000000000000 R11: dffffc0000000001 R12: 0000000000000012
    R13: 000000000000fee5 R14: 0000000000005865 R15: 000000000000fed7
    FS: 000055555633f300(0000) GS:ffff8881f6a00000(0000) knlGS:0000000000000000
    CS: 0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    CR2: 0000000020000000 CR3: 0000000116fea000 CR4: 00000000003506f0
    DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000
    DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400
    Call Trace:
    <TASK>
    [<ffffffff847018dd>] __skb_gso_segment+0x32d/0x4c0 net/core/dev.c:3419
    [<ffffffff8470398a>] skb_gso_segment include/linux/netdevice.h:4819 [inline]
    [<ffffffff8470398a>] validate_xmit_skb+0x3aa/0xee0 net/core/dev.c:3725
    [<ffffffff84707042>] __dev_queue_xmit+0x1332/0x3300 net/core/dev.c:4313
    [<ffffffff851a9ec7>] dev_queue_xmit+0x17/0x20 include/linux/netdevice.h:3029
    [<ffffffff851b4a82>] packet_snd net/packet/af_packet.c:3111 [inline]
    [<ffffffff851b4a82>] packet_sendmsg+0x49d2/0x6470 net/packet/af_packet.c:3142
    [<ffffffff84669a12>] sock_sendmsg_nosec net/socket.c:716 [inline]
    [<ffffffff84669a12>] sock_sendmsg net/socket.c:736 [inline]
    [<ffffffff84669a12>] __sys_sendto+0x472/0x5f0 net/socket.c:2139
    [<ffffffff84669c75>] __do_sys_sendto net/socket.c:2151 [inline]
```

```
     [<ffffffff84669c75>] __se_sys_sendto net/socket.c:2147 [inline]
     [<ffffffff84669c75>] __x64_sys_sendto+0xe5/0x100 net/socket.c:2147
     [<ffffffff8551d40f>] do_syscall_x64 arch/x86/entry/common.c:50 [inline]
     [<ffffffff8551d40f>] do_syscall_64+0x2f/0x50 arch/x86/entry/common.c:80
     [<ffffffff85600087>] entry_SYSCALL_64_after_hwframe+0x63/0xcd

     Fixes: 469aceddfa3e ("vlan: consolidate VLAN parsing code and limit max parsing depth")
     Reported-by: syzbot <syzkaller@googlegroups.com>
     Signed-off-by: Eric Dumazet <edumazet@google.com>
     Cc: Toke Høiland-Jørgensen <toke@redhat.com>
     Cc: Willem de Bruijn <willemb@google.com>
     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 a21fb2be5d0fd947ed3328131dc4668fc24da669
  Author: Eric Dumazet <edumazet@google.com>
  Date:   Tue May 9 18:29:48 2023 +0000

     net: deal with most data-races in sk_wait_event()

     [ Upstream commit d0ac89f6f9879fae316c155de77b5173b3e2c9c9 ]

     __condition is evaluated twice in sk_wait_event() macro.

     First invocation is lockless, and reads can race with writes,
     as spotted by syzbot.

     BUG: KCSAN: data-race in sk_stream_wait_connect / tcp_disconnect

     write to 0xffff88812d83d6a0 of 4 bytes by task 9065 on cpu 1:
     tcp_disconnect+0x2cd/0xdb0
     inet_shutdown+0x19e/0x1f0 net/ipv4/af_inet.c:911
     __sys_shutdown_sock net/socket.c:2343 [inline]
     __sys_shutdown net/socket.c:2355 [inline]
     __do_sys_shutdown net/socket.c:2363 [inline]
     __se_sys_shutdown+0xf8/0x140 net/socket.c:2361
     __x64_sys_shutdown+0x31/0x40 net/socket.c:2361
     do_syscall_x64 arch/x86/entry/common.c:50 [inline]
     do_syscall_64+0x41/0xc0 arch/x86/entry/common.c:80
     entry_SYSCALL_64_after_hwframe+0x63/0xcd

     read to 0xffff88812d83d6a0 of 4 bytes by task 9040 on cpu 0:
     sk_stream_wait_connect+0x1de/0x3a0 net/core/stream.c:75
     tcp_sendmsg_locked+0x2e4/0x2120 net/ipv4/tcp.c:1266
     tcp_sendmsg+0x30/0x50 net/ipv4/tcp.c:1484
     inet6_sendmsg+0x63/0x80 net/ipv6/af_inet6.c:651
     sock_sendmsg_nosec net/socket.c:724 [inline]
     sock_sendmsg net/socket.c:747 [inline]
     __sys_sendto+0x246/0x300 net/socket.c:2142
     __do_sys_sendto net/socket.c:2154 [inline]
     __se_sys_sendto net/socket.c:2150 [inline]
     __x64_sys_sendto+0x78/0x90 net/socket.c:2150
     do_syscall_x64 arch/x86/entry/common.c:50 [inline]
     do_syscall_64+0x41/0xc0 arch/x86/entry/common.c:80
     entry_SYSCALL_64_after_hwframe+0x63/0xcd

     value changed: 0x00000000 -> 0x00000068

     Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
     Reported-by: syzbot <syzkaller@googlegroups.com>
     Signed-off-by: Eric Dumazet <edumazet@google.com>
     Signed-off-by: David S. Miller <davem@davemloft.net>
     Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit d8ea6171485efcc6ce37c824d73b2c57d6198203
Author: Eric Dumazet <edumazet@google.com>
Date:   Tue May 9 16:35:53 2023 +0000

    net: annotate sk->sk_err write from do_recvmmsg()

    [ Upstream commit e05a5f510f26607616fecdd4ac136310c8bea56b ]

    do_recvmmsg() can write to sk->sk_err from multiple threads.

    As said before, many other points reading or writing sk_err
    need annotations.

    Fixes: 34b88a68f26a ("net: Fix use after free in the recvmmsg exit path")
    Signed-off-by: Eric Dumazet <edumazet@google.com>
    Reported-by: syzbot <syzkaller@googlegroups.com>
    Reviewed-by: Kuniyuki Iwashima <kuniyu@amazon.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 02e7afd659a4c9ce1e98fc01ab4c510f3de1f0b3
Author: Eric Dumazet <edumazet@google.com>
Date:   Tue May 9 16:56:34 2023 +0000

    netlink: annotate accesses to nlk->cb_running

    [ Upstream commit a939d14919b799e6fff8a9c80296ca229ba2f8a4 ]

    Both netlink_recvmsg() and netlink_native_seq_show() read
    nlk->cb_running locklessly. Use READ_ONCE() there.

    Add corresponding WRITE_ONCE() to netlink_dump() and
    __netlink_dump_start()

    syzbot reported:
    BUG: KCSAN: data-race in __netlink_dump_start / netlink_recvmsg

    write to 0xffff88813ea4db59 of 1 bytes by task 28219 on cpu 0:
    __netlink_dump_start+0x3af/0x4d0 net/netlink/af_netlink.c:2399
    netlink_dump_start include/linux/netlink.h:308 [inline]
    rtnetlink_rcv_msg+0x70f/0x8c0 net/core/rtnetlink.c:6130
    netlink_rcv_skb+0x126/0x220 net/netlink/af_netlink.c:2577
    rtnetlink_rcv+0x1c/0x20 net/core/rtnetlink.c:6192
    netlink_unicast_kernel net/netlink/af_netlink.c:1339 [inline]
    netlink_unicast+0x56f/0x640 net/netlink/af_netlink.c:1365
    netlink_sendmsg+0x665/0x770 net/netlink/af_netlink.c:1942
    sock_sendmsg_nosec net/socket.c:724 [inline]
    sock_sendmsg net/socket.c:747 [inline]
    sock_write_iter+0x1aa/0x230 net/socket.c:1138
    call_write_iter include/linux/fs.h:1851 [inline]
    new_sync_write fs/read_write.c:491 [inline]
    vfs_write+0x463/0x760 fs/read_write.c:584
    ksys_write+0xeb/0x1a0 fs/read_write.c:637
    __do_sys_write fs/read_write.c:649 [inline]
    __se_sys_write fs/read_write.c:646 [inline]
    __x64_sys_write+0x42/0x50 fs/read_write.c:646
    do_syscall_x64 arch/x86/entry/common.c:50 [inline]
    do_syscall_64+0x41/0xc0 arch/x86/entry/common.c:80
    entry_SYSCALL_64_after_hwframe+0x63/0xcd

    read to 0xffff88813ea4db59 of 1 bytes by task 28222 on cpu 1:
    netlink_recvmsg+0x3b4/0x730 net/netlink/af_netlink.c:2022
    sock_recvmsg_nosec+0x4c/0x80 net/socket.c:1017
    ____sys_recvmsg+0x2db/0x310 net/socket.c:2718
    ___sys_recvmsg net/socket.c:2762 [inline]
```

```
    do_recvmmsg+0x2e5/0x710 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+0xe2/0x160 net/socket.c:2951
    do_syscall_x64 arch/x86/entry/common.c:50 [inline]
    do_syscall_64+0x41/0xc0 arch/x86/entry/common.c:80
    entry_SYSCALL_64_after_hwframe+0x63/0xcd

    value changed: 0x00 -> 0x01

    Fixes: 16b304f3404f ("netlink: Eliminate kmalloc in netlink dump operation.")
    Reported-by: syzbot <syzkaller@googlegroups.com>
    Signed-off-by: Eric Dumazet <edumazet@google.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 637fb1fcc28bca46541f2a34bd470d991b8974e6
Author: Hangbin Liu <liuhangbin@gmail.com>
Date:   Tue May 9 11:11:57 2023 +0800

    bonding: fix send_peer_notif overflow

    [ Upstream commit 9949e2efb54eb3001cb2f6512ff3166dddbfb75d ]

    Bonding send_peer_notif was defined as u8. Since commit 07a4ddec3ce9
    ("bonding: add an option to specify a delay between peer notifications").
    the bond->send_peer_notif will be num_peer_notif multiplied by
    peer_notif_delay, which is u8 * u32. This would cause the send_peer_notif
    overflow easily. e.g.

      ip link add bond0 type bond mode 1 miimon 100 num_grat_arp 30 peer_notify_delay 1000

    To fix the overflow, let's set the send_peer_notif to u32 and limit
    peer_notif_delay to 300s.

    Reported-by: Liang Li <liali@redhat.com>
    Closes: https://bugzilla.redhat.com/show_bug.cgi?id=2090053
    Fixes: 07a4ddec3ce9 ("bonding: add an option to specify a delay between peer
notifications")
    Signed-off-by: Hangbin Liu <liuhangbin@gmail.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 687ca20ce5f93be7a88fdad3997fe7b5781d444b
Author: Florian Westphal <fw@strlen.de>
Date:   Thu May 4 14:55:02 2023 +0200

    netfilter: conntrack: fix possible bug_on with enable_hooks=1

    [ Upstream commit e72eeab542dbf4f544e389e64fa13b82a1b6d003 ]

    I received a bug report (no reproducer so far) where we trip over

    712         rcu_read_lock();
    713         ct_hook = rcu_dereference(nf_ct_hook);
    714         BUG_ON(ct_hook == NULL);  // here

    In nf_conntrack_destroy().

    First turn this BUG_ON into a WARN.  I think it was triggered
    via enable_hooks=1 flag.

    When this flag is turned on, the conntrack hooks are registered
    before nf_ct_hook pointer gets assigned.
```

```
    This opens a short window where packets enter the conntrack machinery,
    can have skb->_nfct set up and a subsequent kfree_skb might occur
    before nf_ct_hook is set.

    Call nf_conntrack_init_end() to set nf_ct_hook before we register the
    pernet ops.

    Fixes: ba3fbe663635 ("netfilter: nf_conntrack: provide modparam to always register
conntrack hooks")
    Signed-off-by: Florian Westphal <fw@strlen.de>
    Signed-off-by: Pablo Neira Ayuso <pablo@netfilter.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 94032527efbac13be702c76afb9d872c0cca7a43
Author: Florian Westphal <fw@strlen.de>
Date:   Thu May 4 14:20:21 2023 +0200

    netfilter: nf_tables: always release netdev hooks from notifier

    [ Upstream commit dc1c9fd4a8bbe1e06add9053010b652449bfe411 ]

    This reverts "netfilter: nf_tables: skip netdev events generated on netns removal".

    The problem is that when a veth device is released, the veth release
    callback will also queue the peer netns device for removal.

    Its possible that the peer netns is also slated for removal.  In this
    case, the device memory is already released before the pre_exit hook of
    the peer netns runs:

    BUG: KASAN: slab-use-after-free in nf_hook_entry_head+0x1b8/0x1d0
    Read of size 8 at addr ffff88812c0124f0 by task kworker/u8:1/45
    Workqueue: netns cleanup_net
    Call Trace:
     nf_hook_entry_head+0x1b8/0x1d0
     __nf_unregister_net_hook+0x76/0x510
     nft_netdev_unregister_hooks+0xa0/0x220
     __nft_release_hook+0x184/0x490
     nf_tables_pre_exit_net+0x12f/0x1b0
     ..

    Order is:
    1. First netns is released, veth_dellink() queues peer netns device
       for removal
    2. peer netns is queued for removal
    3. peer netns device is released, unreg event is triggered
    4. unreg event is ignored because netns is going down
    5. pre_exit hook calls nft_netdev_unregister_hooks but device memory
       might be free'd already.

    Fixes: 68a3765c659f ("netfilter: nf_tables: skip netdev events generated on netns
removal")
    Signed-off-by: Florian Westphal <fw@strlen.de>
    Signed-off-by: Pablo Neira Ayuso <pablo@netfilter.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 31b2c10eee2bf2dbee4b9a8998b64ab71f3e1fd5
Author: Florian Fainelli <f.fainelli@gmail.com>
Date:   Mon May 8 16:17:49 2023 -0700

    net: phy: bcm7xx: Correct read from expansion register

    [ Upstream commit 582dbb2cc1a0a7427840f5b1e3c65608e511b061 ]

    Since the driver works in the "legacy" addressing mode, we need to write
```

```
    to the expansion register (0x17) with bits 11:8 set to 0xf to properly
    select the expansion register passed as argument.

    Fixes: f68d08c437f9 ("net: phy: bcm7xxx: Add EPHY entry for 72165")
    Signed-off-by: Florian Fainelli <f.fainelli@gmail.com>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Link: https://lore.kernel.org/r/20230508231749.1681169-1-f.fainelli@gmail.com
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8319220054e5ea5f506d8d4c4b5e234f668ffc3b
Author: Kuniyuki Iwashima <kuniyu@amazon.com>
Date:   Mon May 8 10:55:43 2023 -0700

    net: Fix load-tearing on sk->sk_stamp in sock_recv_cmsgs().

    [ Upstream commit dfd9248c071a3710c24365897459538551cb7167 ]

    KCSAN found a data race in sock_recv_cmsgs() where the read access
    to sk->sk_stamp needs READ_ONCE().

    BUG: KCSAN: data-race in packet_recvmsg / packet_recvmsg

    write (marked) to 0xffff88803c81f258 of 8 bytes by task 19171 on cpu 0:
     sock_write_timestamp include/net/sock.h:2670 [inline]
     sock_recv_cmsgs include/net/sock.h:2722 [inline]
     packet_recvmsg+0xb97/0xd00 net/packet/af_packet.c:3489
     sock_recvmsg_nosec net/socket.c:1019 [inline]
     sock_recvmsg+0x11a/0x130 net/socket.c:1040
     sock_read_iter+0x176/0x220 net/socket.c:1118
     call_read_iter include/linux/fs.h:1845 [inline]
     new_sync_read fs/read_write.c:389 [inline]
     vfs_read+0x5e0/0x630 fs/read_write.c:470
     ksys_read+0x163/0x1a0 fs/read_write.c:613
     __do_sys_read fs/read_write.c:623 [inline]
     __se_sys_read fs/read_write.c:621 [inline]
     __x64_sys_read+0x41/0x50 fs/read_write.c:621
     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 0xffff88803c81f258 of 8 bytes by task 19183 on cpu 1:
     sock_recv_cmsgs include/net/sock.h:2721 [inline]
     packet_recvmsg+0xb64/0xd00 net/packet/af_packet.c:3489
     sock_recvmsg_nosec net/socket.c:1019 [inline]
     sock_recvmsg+0x11a/0x130 net/socket.c:1040
     sock_read_iter+0x176/0x220 net/socket.c:1118
     call_read_iter include/linux/fs.h:1845 [inline]
     new_sync_read fs/read_write.c:389 [inline]
     vfs_read+0x5e0/0x630 fs/read_write.c:470
     ksys_read+0x163/0x1a0 fs/read_write.c:613
     __do_sys_read fs/read_write.c:623 [inline]
     __se_sys_read fs/read_write.c:621 [inline]
     __x64_sys_read+0x41/0x50 fs/read_write.c:621
     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: 0xffffffffc4653600 -> 0x0000000000000000

    Reported by Kernel Concurrency Sanitizer on:
    CPU: 1 PID: 19183 Comm: syz-executor.5 Not tainted 6.3.0-rc7-02330-gca6270c12e20 #2
    Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS rel-1.16.0-0-gd239552ce722-
prebuilt.qemu.org 04/01/2014
```

```
    Fixes: 6c7c98bad488 ("sock: avoid dirtying sk_stamp, if possible")
    Reported-by: syzbot <syzkaller@googlegroups.com>
    Signed-off-by: Kuniyuki Iwashima <kuniyu@amazon.com>
    Reviewed-by: Eric Dumazet <edumazet@google.com>
    Link: https://lore.kernel.org/r/20230508175543.55756-1-kuniyu@amazon.com
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 996eef225dccc060bb62e99d158ef9508eb77f2e
Author: Marek Vasut <marex@denx.de>
Date:   Sun May 7 01:58:45 2023 +0200

    net: stmmac: Initialize MAC_ONEUS_TIC_COUNTER register

    [ Upstream commit 8efbdbfa99381a017dd2c0f6375a7d80a8118b74 ]

    Initialize MAC_ONEUS_TIC_COUNTER register with correct value derived
    from CSR clock, otherwise EEE is unstable on at least NXP i.MX8M Plus
    and Micrel KSZ9131RNX PHY, to the point where not even ARP request can
    be sent out.

    i.MX 8M Plus Applications Processor Reference Manual, Rev. 1, 06/2021
    11.7.6.1.34 One-microsecond Reference Timer (MAC_ONEUS_TIC_COUNTER)
    defines this register as:
    "
    This register controls the generation of the Reference time (1 microsecond
    tic) for all the LPI timers. This timer has to be programmed by the software
    initially.
    ...
    The application must program this counter so that the number of clock cycles
    of CSR clock is 1us. (Subtract 1 from the value before programming).
    For example if the CSR clock is 100MHz then this field needs to be programmed
    to value 100 - 1 = 99 (which is 0x63).
    This is required to generate the 1US events that are used to update some of
    the EEE related counters.
    "

    The reset value is 0x63 on i.MX8M Plus, which means expected CSR clock are
    100 MHz. However, the i.MX8M Plus "enet_qos_root_clk" are 266 MHz instead,
    which means the LPI timers reach their count much sooner on this platform.

    This is visible using a scope by monitoring e.g. exit from LPI mode on TX_CTL
    line from MAC to PHY. This should take 30us per STMMAC_DEFAULT_TWT_LS setting,
    during which the TX_CTL line transitions from tristate to low, and 30 us later
    from low to high. On i.MX8M Plus, this transition takes 11 us, which matches
    the 30us * 100/266 formula for misconfigured MAC_ONEUS_TIC_COUNTER register.

    Configure MAC_ONEUS_TIC_COUNTER based on CSR clock, so that the LPI timers
    have correct 1us reference. This then fixes EEE on i.MX8M Plus with Micrel
    KSZ9131RNX PHY.

    Fixes: 477286b53f55 ("stmmac: add GMAC4 core support")
    Signed-off-by: Marek Vasut <marex@denx.de>
    Tested-by: Harald Seiler <hws@denx.de>
    Reviewed-by: Francesco Dolcini <francesco.dolcini@toradex.com>
    Tested-by: Francesco Dolcini <francesco.dolcini@toradex.com> # Toradex Verdin iMX8MP
    Reviewed-by: Jesse Brandeburg <jesse.brandeburg@intel.com>
    Link: https://lore.kernel.org/r/20230506235845.246105-1-marex@denx.de
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 96c26eaa685c83d5873aadf2423469df01fe7898
Author: Roy Novich <royno@nvidia.com>
Date:   Sun May 7 16:57:43 2023 +0300
```

```
    linux/dim: Do nothing if no time delta between samples

    [ Upstream commit 162bd18eb55adf464a0fa2b4144b8d61c75ff7c2 ]

    Add return value for dim_calc_stats. This is an indication for the
    caller if curr_stats was assigned by the function. Avoid using
    curr_stats uninitialized over {rdma/net}_dim, when no time delta between
    samples. Coverity reported this potential use of an uninitialized
    variable.

    Fixes: 4c4dbb4a7363 ("net/mlx5e: Move dynamic interrupt coalescing code to
include/linux")
    Fixes: cb3c7fd4f839 ("net/mlx5e: Support adaptive RX coalescing")
    Signed-off-by: Roy Novich <royno@nvidia.com>
    Reviewed-by: Aya Levin <ayal@nvidia.com>
    Reviewed-by: Saeed Mahameed <saeedm@nvidia.com>
    Signed-off-by: Tariq Toukan <tariqt@nvidia.com>
    Reviewed-by: Leon Romanovsky <leonro@nvidia.com>
    Reviewed-by: Michal Kubiak <michal.kubiak@intel.com>
    Link: https://lore.kernel.org/r/20230507135743.138993-1-tariqt@nvidia.com
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9c035b4d3778acea89d6ac3a56c10ddf88d9b221
Author: Thomas Gleixner <tglx@linutronix.de>
Date:   Sat May 6 18:40:57 2023 +0200

    tick/broadcast: Make broadcast device replacement work correctly

    [ Upstream commit f9d36cf445ffff0b913ba187a3eff78028f9b1fb ]

    When a tick broadcast clockevent device is initialized for one shot mode
    then tick_broadcast_setup_oneshot() OR's the periodic broadcast mode
    cpumask into the oneshot broadcast cpumask.

    This is required when switching from periodic broadcast mode to oneshot
    broadcast mode to ensure that CPUs which are waiting for periodic
    broadcast are woken up on the next tick.

    But it is subtly broken, when an active broadcast device is replaced and
    the system is already in oneshot (NOHZ/HIGHRES) mode. Victor observed
    this and debugged the issue.

    Then the OR of the periodic broadcast CPU mask is wrong as the periodic
    cpumask bits are sticky after tick_broadcast_enable() set it for a CPU
    unless explicitly cleared via tick_broadcast_disable().

    That means that this sets all other CPUs which have tick broadcasting
    enabled at that point unconditionally in the oneshot broadcast mask.

    If the affected CPUs were already idle and had their bits set in the
    oneshot broadcast mask then this does no harm. But for non idle CPUs
    which were not set this corrupts their state.

    On their next invocation of tick_broadcast_enable() they observe the bit
    set, which indicates that the broadcast for the CPU is already set up.
    As a consequence they fail to update the broadcast event even if their
    earliest expiring timer is before the actually programmed broadcast
    event.

    If the programmed broadcast event is far in the future, then this can
    cause stalls or trigger the hung task detector.

    Avoid this by telling tick_broadcast_setup_oneshot() explicitly whether
    this is the initial switch over from periodic to oneshot broadcast which
```

```
    must take the periodic broadcast mask into account. In the case of
    initialization of a replacement device this prevents that the broadcast
    oneshot mask is modified.

    There is a second problem with broadcast device replacement in this
    function. The broadcast device is only armed when the previous state of
    the device was periodic.

    That is correct for the switch from periodic broadcast mode to oneshot
    broadcast mode as the underlying broadcast device could operate in
    oneshot state already due to lack of periodic state in hardware. In that
    case it is already armed to expire at the next tick.

    For the replacement case this is wrong as the device is in shutdown
    state. That means that any already pending broadcast event will not be
    armed.

    This went unnoticed because any CPU which goes idle will observe that
    the broadcast device has an expiry time of KTIME_MAX and therefore any
    CPUs next timer event will be earlier and cause a reprogramming of the
    broadcast device. But that does not guarantee that the events of the
    CPUs which were already in idle are delivered on time.

    Fix this by arming the newly installed device for an immediate event
    which will reevaluate the per CPU expiry times and reprogram the
    broadcast device accordingly. This is simpler than caching the last
    expiry time in yet another place or saving it before the device exchange
    and handing it down to the setup function. Replacement of broadcast
    devices is not a frequent operation and usually happens once somewhere
    late in the boot process.

    Fixes: 9c336c9935cf ("tick/broadcast: Allow late registered device to enter oneshot
mode")
    Reported-by: Victor Hassan <victor@allwinnertech.com>
    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Reviewed-by: Frederic Weisbecker <frederic@kernel.org>
    Link: https://lore.kernel.org/r/87pm7d2z1i.ffs@tglx
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5afaca177c115a5f62d0b99e293cd88b206b704c
Author: Keoseong Park <keosung.park@samsung.com>
Date:   Tue Apr 25 12:17:21 2023 +0900

    scsi: ufs: core: Fix I/O hang that occurs when BKOPS fails in W-LUN suspend

    [ Upstream commit 1a7edd041f2d252f251523ba3f2eaead076a8f8d ]

    Even when urgent BKOPS fails, the consumer will get stuck in runtime
    suspend status. Like commit 1a5665fc8d7a ("scsi: ufs: core: WLUN suspend
    SSU/enter hibern8 fail recovery"), trigger the error handler and return
    -EBUSY to break the suspend.

    Fixes: b294ff3e3449 ("scsi: ufs: core: Enable power management for wlun")
    Signed-off-by: Keoseong Park <keosung.park@samsung.com>
    Link:
https://lore.kernel.org/r/20230425031721epcms2p5d4de65616478c967d466626e20c42a3a@epcms2p5
    Reviewed-by: Avri Altman <avri.altman@wdc.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1166786def4b364d0e81d3a991cf83ebf4417971
Author: Yang Jihong <yangjihong1@huawei.com>
Date:   Tue Apr 25 10:32:17 2023 +0000

    perf/core: Fix perf_sample_data not properly initialized for different swevents in
```

```
perf_tp_event()

    [ Upstream commit 1d1bfe30dad50d4bea83cd38d73c441972ea0173 ]

    data->sample_flags may be modified in perf_prepare_sample(),
    in perf_tp_event(), different swevents use the same on-stack
    perf_sample_data, the previous swevent may change sample_flags in
    perf_prepare_sample(), as a result, some members of perf_sample_data are
    not correctly initialized when next swevent_event preparing sample
    (for example data->id, the value varies according to swevent).

    A simple scenario triggers this problem is as follows:

      # perf record -e sched:sched_switch --switch-output-event sched:sched_switch -a sleep
1
      [ perf record: dump data: Woken up 0 times ]
      [ perf record: Dump perf.data.2023041209014396 ]
      [ perf record: dump data: Woken up 0 times ]
      [ perf record: Dump perf.data.2023041209014662 ]
      [ perf record: dump data: Woken up 0 times ]
      [ perf record: Dump perf.data.2023041209014910 ]
      [ perf record: Woken up 0 times to write data ]
      [ perf record: Dump perf.data.2023041209015164 ]
      [ perf record: Captured and wrote 0.069 MB perf.data.<timestamp> ]
      # ls -l
      total 860
      -rw------- 1 root root  95694 Apr 12 09:01 perf.data.2023041209014396
      -rw------- 1 root root 606430 Apr 12 09:01 perf.data.2023041209014662
      -rw------- 1 root root  82246 Apr 12 09:01 perf.data.2023041209014910
      -rw------- 1 root root  82342 Apr 12 09:01 perf.data.2023041209015164
      # perf script -i perf.data.2023041209014396
      0x11d58 [0x80]: failed to process type: 9 [Bad address]

    Solution: Re-initialize perf_sample_data after each event is processed.
    Note that data->raw->frag.data may be accessed in perf_tp_event_match().
    Therefore, need to init sample_data and then go through swevent hlist to prevent
    reference of NULL pointer, reported by [1].

    After fix:

      # perf record -e sched:sched_switch --switch-output-event sched:sched_switch -a sleep
1
      [ perf record: dump data: Woken up 0 times ]
      [ perf record: Dump perf.data.2023041209442259 ]
      [ perf record: dump data: Woken up 0 times ]
      [ perf record: Dump perf.data.2023041209442514 ]
      [ perf record: dump data: Woken up 0 times ]
      [ perf record: Dump perf.data.2023041209442760 ]
      [ perf record: Woken up 0 times to write data ]
      [ perf record: Dump perf.data.2023041209443003 ]
      [ perf record: Captured and wrote 0.069 MB perf.data.<timestamp> ]
      # ls -l
      total 864
      -rw------- 1 root root 100166 Apr 12 09:44 perf.data.2023041209442259
      -rw------- 1 root root 606438 Apr 12 09:44 perf.data.2023041209442514
      -rw------- 1 root root  82246 Apr 12 09:44 perf.data.2023041209442760
      -rw------- 1 root root  82342 Apr 12 09:44 perf.data.2023041209443003
      # perf script -i perf.data.2023041209442259 | head -n 5
                 perf   232 [000]    66.846217: sched:sched_switch: prev_comm=perf
prev_pid=232 prev_prio=120 prev_state=D ==> next_comm=perf next_pid=234 next_prio=120
                 perf   234 [000]    66.846449: sched:sched_switch: prev_comm=perf
prev_pid=234 prev_prio=120 prev_state=S ==> next_comm=perf next_pid=232 next_prio=120
                 perf   232 [000]    66.846546: sched:sched_switch: prev_comm=perf
prev_pid=232 prev_prio=120 prev_state=R ==> next_comm=perf next_pid=234 next_prio=120
                 perf   234 [000]    66.846606: sched:sched_switch: prev_comm=perf
```

```
prev_pid=234 prev_prio=120 prev_state=S ==> next_comm=perf next_pid=232 next_prio=120
                  perf   232 [000]    66.846646: sched:sched_switch: prev_comm=perf
prev_pid=232 prev_prio=120 prev_state=R ==> next_comm=perf next_pid=234 next_prio=120
```

    [1] Link: https://lore.kernel.org/oe-lkp/202304250929.efef2caa-yujie.liu@intel.com

    Fixes: bb447c27a467 ("perf/core: Set data->sample_flags in perf_prepare_sample()")
    Signed-off-by: Yang Jihong <yangjihong1@huawei.com>
    Signed-off-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Link: https://lkml.kernel.org/r/20230425103217.130600-1-yangjihong1@huawei.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 091a480ecae03740f4dc909a2c64f45862a464a6
Author: Christophe JAILLET <christophe.jaillet@wanadoo.fr>
Date:   Fri May 5 20:39:33 2023 +0200

    net: mdio: mvusb: Fix an error handling path in mvusb_mdio_probe()

    [ Upstream commit 27c1eaa07283b0c94becf8241f95368267cf558b ]

    Should of_mdiobus_register() fail, a previous usb_get_dev() call should be
    undone as in the .disconnect function.

    Fixes: 04e37d92fbed ("net: phy: add marvell usb to mdio controller")
    Signed-off-by: Christophe JAILLET <christophe.jaillet@wanadoo.fr>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Reviewed-by: Andrew Lunn <andrew@lunn.ch>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 70a76d6816148819d0464f71aafa126c84826628
Author: Eric Dumazet <edumazet@google.com>
Date:   Fri May 5 17:06:18 2023 +0000

    net: skb_partial_csum_set() fix against transport header magic value

    [ Upstream commit 424f8416bb39936df6365442d651ee729b283460 ]

    skb->transport_header uses the special 0xFFFF value
    to mark if the transport header was set or not.

    We must prevent callers to accidentaly set skb->transport_header
    to 0xFFFF. Note that only fuzzers can possibly do this today.

    syzbot reported:

    WARNING: CPU: 0 PID: 2340 at include/linux/skbuff.h:2847 skb_transport_offset
include/linux/skbuff.h:2956 [inline]
    WARNING: CPU: 0 PID: 2340 at include/linux/skbuff.h:2847
virtio_net_hdr_to_skb+0xbcc/0x10c0 include/linux/virtio_net.h:103
    Modules linked in:
    CPU: 0 PID: 2340 Comm: syz-executor.0 Not tainted 6.3.0-syzkaller #0
    Hardware name: Google Google Compute Engine/Google Compute Engine, BIOS Google
04/14/2023
    RIP: 0010:skb_transport_header include/linux/skbuff.h:2847 [inline]
    RIP: 0010:skb_transport_offset include/linux/skbuff.h:2956 [inline]
    RIP: 0010:virtio_net_hdr_to_skb+0xbcc/0x10c0 include/linux/virtio_net.h:103
    Code: 41 39 df 0f 82 c3 04 00 00 48 8b 7c 24 10 44 89 e6 e8 08 6e 59 ff 48 85 c0 74 54
e8 ce 36 7e fc e9 37 f8 ff ff e8 c4 36 7e fc <0f> 0b e9 93 f8 ff ff 44 89 f7 44 89 e6 e8 32
38 7e fc 45 39 e6 0f
    RSP: 0018:ffffc90004497880 EFLAGS: 00010293
    RAX: ffffffff84fea55c RBX: 000000000000ffff RCX: ffff888120be2100
    RDX: 0000000000000000 RSI: 000000000000ffff RDI: 000000000000ffff
    RBP: ffffc90004497990 R08: ffffffff84fe9de5 R09: 0000000000000034
    R10: ffffea00048ebd80 R11: 0000000000000034 R12: ffff88811dc2d9c8

```
    R13: dffffc0000000000 R14: ffff88811dc2d9ae R15: 1ffff11023b85b35
    FS: 00007f9211a59700(0000) GS:ffff8881f6c00000(0000) knlGS:0000000000000000
    CS: 0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    CR2: 00000000200002c0 CR3: 00000001215a5000 CR4: 00000000003506f0
    DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000
    DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400
    Call Trace:
    <TASK>
    packet_snd net/packet/af_packet.c:3076 [inline]
    packet_sendmsg+0x4590/0x61a0 net/packet/af_packet.c:3115
    sock_sendmsg_nosec net/socket.c:724 [inline]
    sock_sendmsg net/socket.c:747 [inline]
    __sys_sendto+0x472/0x630 net/socket.c:2144
    __do_sys_sendto net/socket.c:2156 [inline]
    __se_sys_sendto net/socket.c:2152 [inline]
    __x64_sys_sendto+0xe5/0x100 net/socket.c:2152
    do_syscall_x64 arch/x86/entry/common.c:50 [inline]
    do_syscall_64+0x2f/0x50 arch/x86/entry/common.c:80
    entry_SYSCALL_64_after_hwframe+0x63/0xcd
    RIP: 0033:0x7f9210c8c169
    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:00007f9211a59168 EFLAGS: 00000246 ORIG_RAX: 000000000000002c
    RAX: ffffffffffffffda RBX: 00007f9210dabf80 RCX: 00007f9210c8c169
    RDX: 000000000000ffed RSI: 00000000200000c0 RDI: 0000000000000003
    RBP: 00007f9210ce7ca1 R08: 0000000020000540 R09: 0000000000000014
    R10: 0000000000000000 R11: 0000000000000246 R12: 0000000000000000
    R13: 00007ffe135d65cf R14: 00007f9211a59300 R15: 0000000000022000

    Fixes: 66e4c8d95008 ("net: warn if transport header was not set")
    Signed-off-by: Eric Dumazet <edumazet@google.com>
    Reported-by: syzbot <syzkaller@googlegroups.com>
    Cc: Willem de Bruijn <willemb@google.com>
    Reviewed-by: Willem de Bruijn <willemb@google.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a77ccc77d941c03e4252bb9a865ad451ce944ccf
Author: Randy Dunlap <rdunlap@infradead.org>
Date:   Sun Apr 23 06:48:45 2023 +0100

    ARM: 9296/1: HP Jornada 7XX: fix kernel-doc warnings

    [ Upstream commit 46dd6078dbc7e363a8bb01209da67015a1538929 ]

    Fix kernel-doc warnings from the kernel test robot:

    jornada720_ssp.c:24: warning: Function parameter or member 'jornada_ssp_lock' not
 described in 'DEFINE_SPINLOCK'
    jornada720_ssp.c:24: warning: expecting prototype for arch/arm/mac(). Prototype was for
 DEFINE_SPINLOCK() instead
    jornada720_ssp.c:34: warning: Function parameter or member 'byte' not described in
 'jornada_ssp_reverse'
    jornada720_ssp.c:57: warning: Function parameter or member 'byte' not described in
 'jornada_ssp_byte'
    jornada720_ssp.c:85: warning: Function parameter or member 'byte' not described in
 'jornada_ssp_inout'

    Link: lore.kernel.org/r/202304210535.tWby3jWF-lkp@intel.com

    Fixes: 69ebb22277a5 ("[ARM] 4506/1: HP Jornada 7XX: Addition of SSP Platform Driver")
    Signed-off-by: Randy Dunlap <rdunlap@infradead.org>
    Reported-by: kernel test robot <lkp@intel.com>
    Cc: Arnd Bergmann <arnd@arndb.de>
```

```
    Cc: Kristoffer Ericson <Kristoffer.ericson@gmail.com>
    Cc: patches@armlinux.org.uk
    Signed-off-by: Russell King (Oracle) <rmk+kernel@armlinux.org.uk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 581e9760aa6d77db60772b1843193303b26d7aa8
Author: Saravana Kannan <saravanak@google.com>
Date:   Thu Mar 9 22:39:09 2023 -0800

    drm/mipi-dsi: Set the fwnode for mipi_dsi_device

    [ Upstream commit a26cc2934331b57b5a7164bff344f0a2ec245fc0 ]

    After commit 3fb16866b51d ("driver core: fw_devlink: Make cycle
    detection more robust"), fw_devlink prints an error when consumer
    devices don't have their fwnode set. This used to be ignored silently.

    Set the fwnode mipi_dsi_device so fw_devlink can find them and properly
    track their dependencies.

    This fixes errors like this:
    [    0.334054] nwl-dsi 30a00000.mipi-dsi: Failed to create device link with regulator-
lcd-1v8
    [    0.346964] nwl-dsi 30a00000.mipi-dsi: Failed to create device link with backlight-
dsi

    Reported-by: Martin Kepplinger <martin.kepplinger@puri.sm>
    Link:
https://lore.kernel.org/lkml/2a8e407f4f18c9350f8629a2b5fa18673355b2ae.camel@puri.sm/
    Fixes: 068a00233969 ("drm: Add MIPI DSI bus support")
    Signed-off-by: Saravana Kannan <saravanak@google.com>
    Tested-by: Martin Kepplinger <martin.kepplinger@puri.sm>
    Link: https://lore.kernel.org/r/20230310063910.2474472-1-saravanak@google.com
    Signed-off-by: Maxime Ripard <maxime@cerno.tech>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 49cacb5d9782cd70e5690eae96c77ac5df339f20
Author: Kees Cook <keescook@chromium.org>
Date:   Sat Feb 4 10:43:10 2023 -0800

    drm/nouveau/disp: More DP_RECEIVER_CAP_SIZE array fixes

    [ Upstream commit 25feda6fbd0cfefcb69308fb20d4d4815a107c5e ]

    More arrays (and arguments) for dcpd were set to 16, when it looks like
    DP_RECEIVER_CAP_SIZE (15) should be used. Fix the remaining cases, seen
    with GCC 13:

    ../drivers/gpu/drm/nouveau/nvif/outp.c: In function 'nvif_outp_acquire_dp':
    ../include/linux/fortify-string.h:57:33: warning: array subscript 'unsigned char[16][0]'
is partly outside array bounds of 'u8[15]' {aka 'unsigned char[15]'} [-Warray-bounds=]
       57 | #define __underlying_memcpy     __builtin_memcpy
          |                                 ^
    ...
    ../drivers/gpu/drm/nouveau/nvif/outp.c:140:9: note: in expansion of macro 'memcpy'
      140 |         memcpy(args.dp.dpcd, dpcd, sizeof(args.dp.dpcd));
          |         ^~~~~
    ../drivers/gpu/drm/nouveau/nvif/outp.c:130:49: note: object 'dpcd' of size [0, 15]
      130 | nvif_outp_acquire_dp(struct nvif_outp *outp, u8 dpcd[DP_RECEIVER_CAP_SIZE],
          |                                              ~~~^~~~~~~~~~~~~~~~~~~~~~~~~

    Fixes: 813443721331 ("drm/nouveau/disp: move DP link config into acquire")
    Cc: Ben Skeggs <bskeggs@redhat.com>
    Cc: Lyude Paul <lyude@redhat.com>
    Cc: Karol Herbst <kherbst@redhat.com>
```

```
    Cc: David Airlie <airlied@gmail.com>
    Cc: Daniel Vetter <daniel@ffwll.ch>
    Cc: Dave Airlie <airlied@redhat.com>
    Cc: "Gustavo A. R. Silva" <gustavo@embeddedor.com>
    Cc: dri-devel@lists.freedesktop.org
    Cc: nouveau@lists.freedesktop.org
    Signed-off-by: Kees Cook <keescook@chromium.org>
    Reviewed-by: Gustavo A. R. Silva <gustavoars@kernel.org>
    Reviewed-by: Karol Herbst <kherbst@redhat.com>
    Signed-off-by: Karol Herbst <git@karolherbst.de>
    Link: https://patchwork.freedesktop.org/patch/msgid/20230204184307.never.825-
kees@kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4d1ad98912de7b08806d7b937750b7e85b52f143
Author: Jani Nikula <jani.nikula@intel.com>
Date:   Thu Apr 6 16:46:15 2023 +0300

    drm/dsc: fix DP_DSC_MAX_BPP_DELTA_* macro values

    [ Upstream commit 0d68683838f2850dd8ff31f1121e05bfb7a2def0 ]

    The macro values just don't match the specs. Fix them.

    Fixes: 1482ec00be4a ("drm: Add missing DP DSC extended capability definitions.")
    Cc: Vinod Govindapillai <vinod.govindapillai@intel.com>
    Cc: Stanislav Lisovskiy <stanislav.lisovskiy@intel.com>
    Signed-off-by: Jani Nikula <jani.nikula@intel.com>
    Reviewed-by: Ankit Nautiyal <ankit.k.nautiyal@intel.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20230406134615.1422509-2-
jani.nikula@intel.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 156787833d6e31e74665fbeafd78acec9fd11b7e
Author: Pierre Asselin <pa@panix.com>
Date:   Wed Apr 19 00:48:34 2023 -0400

    firmware/sysfb: Fix VESA format selection

    [ Upstream commit 1b617bc93178912fa36f87a957c15d1f1708c299 ]

    Some legacy BIOSes report no reserved bits in their 32-bit rgb mode,
    breaking the calculation of bits_per_pixel in  commit f35cd3fa7729
    ("firmware/sysfb: Fix EFI/VESA format selection").  However they report
    lfb_depth correctly for those modes.  Keep the computation but
    set bits_per_pixel to lfb_depth if the latter is larger.

    v2 fixes the warnings from a max3() macro with arguments of different
    types;  split the bits_per_pixel assignment to avoid uglyfing the code
    with too many typecasts.

    v3 fixes space and formatting blips pointed out by Javier, and change
    the bit_per_pixel assignment back to a single statement using two casts.

    v4 go back to v2 and use max_t()

    Signed-off-by: Pierre Asselin <pa@panix.com>
    Fixes: f35cd3fa7729 ("firmware/sysfb: Fix EFI/VESA format selection")
    Link: https://lore.kernel.org/r/4Psm6B6Lqkz1QXM@panix3.panix.com
    Link: https://lore.kernel.org/r/20230412150225.3757223-1-javierm@redhat.com
    Tested-by: Thomas Zimmermann <tzimmermann@suse.de>
    Signed-off-by: Thomas Zimmermann <tzimmermann@suse.de>
    Link: https://patchwork.freedesktop.org/patch/msgid/20230419044834.10816-1-pa@panix.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 251653fa974ea551a15d16cacfed7cde68cc7f87
Author: Sui Jingfeng <suijingfeng@loongson.cn>
Date:   Thu Apr 20 11:05:00 2023 +0800

    drm/fbdev-generic: prohibit potential out-of-bounds access

    [ Upstream commit c8687694bb1f5c48134f152f8c5c2e53483eb99d ]

    The fbdev test of IGT may write after EOF, which lead to out-of-bound
    access for drm drivers with fbdev-generic. For example, run fbdev test
    on a x86+ast2400 platform, with 1680x1050 resolution, will cause the
    linux kernel hang with the following call trace:

      Oops: 0000 [#1] PREEMPT SMP PTI
      [IGT] fbdev: starting subtest eof
      Workqueue: events drm_fb_helper_damage_work [drm_kms_helper]
      [IGT] fbdev: starting subtest nullptr

      RIP: 0010:memcpy_erms+0xa/0x20
      RSP: 0018:ffffa17d40167d98 EFLAGS: 00010246
      RAX: ffffa17d4eb7fa80 RBX: ffffa17d40e0aa80 RCX: 00000000000014c0
      RDX: 0000000000001a40 RSI: ffffa17d40e0b000 RDI: ffffa17d4eb80000
      RBP: ffffa17d40167e20 R08: 0000000000000000 R09: ffff89522ecff8c0
      R10: ffffa17d4e4c5000 R11: 0000000000000000 R12: ffffa17d4eb7fa80
      R13: 0000000000001a40 R14: 000000000000041a R15: ffffa17d40167e30
      FS:  0000000000000000(0000) GS:ffff895257380000(0000) knlGS:0000000000000000
      CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
      CR2: ffffa17d40e0b000 CR3: 00000001eaeca006 CR4: 00000000001706e0
      Call Trace:
       <TASK>
       ? drm_fbdev_generic_helper_fb_dirty+0x207/0x330 [drm_kms_helper]
       drm_fb_helper_damage_work+0x8f/0x170 [drm_kms_helper]
       process_one_work+0x21f/0x430
       worker_thread+0x4e/0x3c0
       ? __pfx_worker_thread+0x10/0x10
       kthread+0xf4/0x120
       ? __pfx_kthread+0x10/0x10
       ret_from_fork+0x2c/0x50
       </TASK>
      CR2: ffffa17d40e0b000
      ---[ end trace 0000000000000000 ]---

    The is because damage rectangles computed by
    drm_fb_helper_memory_range_to_clip() function is not guaranteed to be
    bound in the screen's active display area. Possible reasons are:

    1) Buffers are allocated in the granularity of page size, for mmap system
       call support. The shadow screen buffer consumed by fbdev emulation may
       also choosed be page size aligned.

    2) The DIV_ROUND_UP() used in drm_fb_helper_memory_range_to_clip()
       will introduce off-by-one error.

    For example, on a 16KB page size system, in order to store a 1920x1080
    XRGB framebuffer, we need allocate 507 pages. Unfortunately, the size
    1920*1080*4 can not be divided exactly by 16KB.

     1920 * 1080 * 4 = 8294400 bytes
     506 * 16 * 1024 = 8290304 bytes
     507 * 16 * 1024 = 8306688 bytes

     line_length = 1920*4 = 7680 bytes

     507 * 16 * 1024 / 7680 = 1081.6
```

```
       off / line_length = 507 * 16 * 1024 / 7680 = 1081
       DIV_ROUND_UP(507 * 16 * 1024, 7680) will yeild 1082

     memcpy_toio() typically issue the copy line by line, when copy the last
     line, out-of-bound access will be happen. Because:

       1082 * line_length = 1082 * 7680 = 8309760, and 8309760 > 8306688

     Note that userspace may still write to the invisiable area if a larger
     buffer than width x stride is exposed. But it is not a big issue as
     long as there still have memory resolve the access if not drafting so
     far.

       - Also limit the y1 (Daniel)
       - keep fix patch it to minimal (Daniel)
       - screen_size is page size aligned because of it need mmap (Thomas)
       - Adding fixes tag (Thomas)

     Signed-off-by: Sui Jingfeng <suijingfeng@loongson.cn>
     Fixes: aa15c677cc34 ("drm/fb-helper: Fix vertical damage clipping")
     Reviewed-by: Thomas Zimmermann <tzimmermann@suse.de>
     Tested-by: Geert Uytterhoeven <geert+renesas@glider.be>
     Link: https://lore.kernel.org/dri-devel/ad44df29-3241-0d9e-e708-b0338bf3c623@189.cn/
     Signed-off-by: Thomas Zimmermann <tzimmermann@suse.de>
     Link: https://patchwork.freedesktop.org/patch/msgid/20230420030500.1578756-1-
  suijingfeng@loongson.cn
     Signed-off-by: Sasha Levin <sashal@kernel.org>
```