```
commit ed56826f177921da1cc66a927de22ca1666eb78c
Author: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
Date:   Sat Oct 5 15:12:40 2019 +0200

    Linux 5.3.4

commit d0b85a37c06b3495ecf8268c7670339af3a21fce
Author: Pi-Hsun Shih <pihsun@chromium.org>
Date:   Wed Sep 4 14:26:13 2019 +0800

    platform/chrome: cros_ec_rpmsg: Fix race with host command when probe failed

    [ Upstream commit 71cddb7097e2b0feb855d7fd7d59afd12cbee4bb ]

    Since the rpmsg_endpoint is created before probe is called, it's
    possible that a host event is received during cros_ec_register, and
    there would be some pending work in the host_event_work workqueue while
    cros_ec_register is called.

    If cros_ec_register fails, when the leftover work in host_event_work
    run, the ec_dev from the drvdata of the rpdev could be already set to
    NULL, causing kernel crash when trying to run cros_ec_get_next_event.

    Fix this by creating the rpmsg_endpoint by ourself, and when
    cros_ec_register fails (or on remove), destroy the endpoint first (to
    make sure there's no more new calls to cros_ec_rpmsg_callback), and then
    cancel all works in the host_event_work workqueue.

    Cc: stable@vger.kernel.org
    Fixes: 2de89fd98958 ("platform/chrome: cros_ec: Add EC host command support using
rpmsg")
    Signed-off-by: Pi-Hsun Shih <pihsun@chromium.org>
    Signed-off-by: Enric Balletbo i Serra <enric.balletbo@collabora.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit bec8c6dec6058e4657ceab0fdae5a4ef0c041d3c
Author: Lorenzo Bianconi <lorenzo@kernel.org>
Date:   Sun Sep 22 15:36:03 2019 +0200

    mt76: mt7615: fix mt7615 firmware path definitions

    [ Upstream commit 9d4d0d06bbf9f7e576b0ebbb2f77672d0fc7f503 ]

    mt7615 patch/n9/cr4 firmwares are available in mediatek folder in
    linux-firmware repository. Because of this mt7615 won't work on regular
    distributions like Ubuntu. Fix path definitions.  Moreover remove useless
    firmware name pointers and use definitions directly

    Fixes: 04b8e65922f6 ("mt76: add mac80211 driver for MT7615 PCIe-based chipsets")
    Cc: stable@vger.kernel.org
    Signed-off-by: Lorenzo Bianconi <lorenzo@kernel.org>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5dab55b417cadb24bae97692d07f9edeba062974
Author: Lorenzo Bianconi <lorenzo@kernel.org>
Date:   Tue Jul 2 11:24:51 2019 +0200

    mt76: mt7615: always release sem in mt7615_load_patch

    [ Upstream commit 2fc446487c364bf8bbd5f8f5f27e52d914fa1d72 ]

    Release patch semaphore even if request_firmware fails in
    mt7615_load_patch
```

```
    Signed-off-by: Lorenzo Bianconi <lorenzo@kernel.org>
    Signed-off-by: Felix Fietkau <nbd@nbd.name>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 88688a6cd741e5110fbe79db62e727a3ec19d6c1
Author: NeilBrown <neilb@suse.de>
Date:   Mon Sep 9 16:30:02 2019 +1000

    md/raid0: avoid RAID0 data corruption due to layout confusion.

    [ Upstream commit c84a1372df929033cb1a0441fb57bd3932f39ac9 ]

    If the drives in a RAID0 are not all the same size, the array is
    divided into zones.
    The first zone covers all drives, to the size of the smallest.
    The second zone covers all drives larger than the smallest, up to
    the size of the second smallest - etc.

    A change in Linux 3.14 unintentionally changed the layout for the
    second and subsequent zones.  All the correct data is still stored, but
    each chunk may be assigned to a different device than in pre-3.14 kernels.
    This can lead to data corruption.

    It is not possible to determine what layout to use - it depends which
    kernel the data was written by.
    So we add a module parameter to allow the old (0) or new (1) layout to be
    specified, and refused to assemble an affected array if that parameter is
    not set.

    Fixes: 20d0189b1012 ("block: Introduce new bio_split()")
    cc: stable@vger.kernel.org (3.14+)
    Acked-by: Guoqing Jiang <guoqing.jiang@cloud.ionos.com>
    Signed-off-by: NeilBrown <neilb@suse.de>
    Signed-off-by: Song Liu <songliubraving@fb.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e49770f69315a2f054cd3e9b5a57e62a96b3db49
Author: Alex Deucher <alexander.deucher@amd.com>
Date:   Fri Sep 20 15:13:24 2019 -0500

    drm/amdgpu/display: fix 64 bit divide

    commit dd9212a885ca4a95443905c7c3781122a4d664e8 upstream.

    Use proper helper for 32 bit.

    Reviewed-by: Harry Wentland <harry.wentland@amd.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 7c779ef422b893e295114cc4055a57e820ca0182
Author: Zhan Liu <zhan.liu@amd.com>
Date:   Thu Aug 22 14:54:18 2019 -0400

    drm/amd/display: Add missing HBM support and raise Vega20's uclk.

    commit c02d6a161395dfc0c2fdabb9e976a229017288d8 upstream.

    [Why]
    When more than 2 displays are connected to the graphics card,
    only the minimum memory clock is needed. However, when more
    displays are connected, the minimum memory clock is not
    sufficient enough to support the overwhelming bandwidth.
    System will hang under this circumstance.
```

```
    Also, the old code didn't address HBM cards, which has 2
    pseudo channels. We need to add the HBM part here.

    [How]
    When graphics card connects to 2 or more displays,
    switch to high memory clock. Also, choose memory
    multiplier based on whether its regular DRAM or HBM.

    Signed-off-by: Zhan Liu <zhan.liu@amd.com>
    Reviewed-by: Roman Li <Roman.Li@amd.com>
    Acked-by: Leo Li <sunpeng.li@amd.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1f430ed084055bb1d2d524af43b2ba67741b22a9
Author: Charlene Liu <charlene.liu@amd.com>
Date:   Tue Aug 20 20:33:46 2019 -0400

    drm/amd/display: dce11.x /dce12 update formula input

    commit c46e5df4ac898108da66a880c4e18f69c74f6c1b upstream.

    [Description]
    1. OUTSTANDING_REQUEST_LIMIT update from 0xFF to 0x1F (HW doc update)
    2. using memory type to convert UMC's MCLK to Yclk.

    Signed-off-by: Charlene Liu <charlene.liu@amd.com>
    Reviewed-by: Dmytro Laktyushkin <Dmytro.Laktyushkin@amd.com>
    Acked-by: Bhawanpreet Lakha <Bhawanpreet.Lakha@amd.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 911ec3ba25fb83445f5cfde8d80132cc888024ea
Author: Kai-Heng Feng <kai.heng.feng@canonical.com>
Date:   Mon Sep 2 16:33:42 2019 +0800

    drm/amd/display: Restore backlight brightness after system resume

    commit bb264220d9316f6bd7c1fd84b8da398c93912931 upstream.

    Laptops with AMD APU doesn't restore display backlight brightness after
    system resume.

    This issue started when DC was introduced.

    Let's use BL_CORE_SUSPENDRESUME so the backlight core calls
    update_status callback after system resume to restore the backlight
    level.

    Tested on Dell Inspiron 3180 (Stoney Ridge) and Dell Latitude 5495
    (Raven Ridge).

    Cc: <stable@vger.kernel.org>
    Signed-off-by: Kai-Heng Feng <kai.heng.feng@canonical.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 4e679ab4b95aee9eae6f6a2fc60b9072389d6fb7
Author: Pavel Shilovsky <pshilov@microsoft.com>
Date:   Thu Sep 26 12:31:20 2019 -0700

    CIFS: Fix oplock handling for SMB 2.1+ protocols

    commit a016e2794fc3a245a91946038dd8f34d65e53cc3 upstream.
```

```
    There may be situations when a server negotiates SMB 2.1
    protocol version or higher but responds to a CREATE request
    with an oplock rather than a lease.

    Currently the client doesn't handle such a case correctly:
    when another CREATE comes in the server sends an oplock
    break to the initial CREATE and the client doesn't send
    an ack back due to a wrong caching level being set (READ
    instead of RWH). Missing an oplock break ack makes the
    server wait until the break times out which dramatically
    increases the latency of the second CREATE.

    Fix this by properly detecting oplocks when using SMB 2.1
    protocol version and higher.

    Cc: <stable@vger.kernel.org>
    Signed-off-by: Pavel Shilovsky <pshilov@microsoft.com>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Reviewed-by: Ronnie Sahlberg <lsahlber@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 9ea37d18a5bbedea0c65fcfa7c97ddec13ea618b
Author: Murphy Zhou <jencce.kernel@gmail.com>
Date:   Sat Sep 21 19:26:00 2019 +0800

    CIFS: fix max ea value size

    commit 63d37fb4ce5ae7bf1e58f906d1bf25f036fe79b2 upstream.

    It should not be larger then the slab max buf size. If user
    specifies a larger size, it passes this check and goes
    straightly to SMB2_set_info_init performing an insecure memcpy.

    Signed-off-by: Murphy Zhou <jencce.kernel@gmail.com>
    Reviewed-by: Aurelien Aptel <aaptel@suse.com>
    CC: Stable <stable@vger.kernel.org>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 6bc7cc6a78a544df678972b89803a3a9efa7b6d7
Author: Chris Brandt <chris.brandt@renesas.com>
Date:   Thu Sep 26 07:19:09 2019 -0500

    i2c: riic: Clear NACK in tend isr

    commit a71e2ac1f32097fbb2beab098687a7a95c84543e upstream.

    The NACKF flag should be cleared in INTRIICNAKI interrupt processing as
    description in HW manual.

    This issue shows up quickly when PREEMPT_RT is applied and a device is
    probed that is not plugged in (like a touchscreen controller). The result
    is endless interrupts that halt system boot.

    Fixes: 310c18a41450 ("i2c: riic: add driver")
    Cc: stable@vger.kernel.org
    Reported-by: Chien Nguyen <chien.nguyen.eb@rvc.renesas.com>
    Signed-off-by: Chris Brandt <chris.brandt@renesas.com>
    Signed-off-by: Wolfram Sang <wsa@the-dreams.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f3ffa1e89901609d27dc4384188b708ca416138c
Author: Laurent Vivier <lvivier@redhat.com>
Date:   Tue Sep 17 11:54:50 2019 +0200
```

```
    hwrng: core - don't wait on add_early_randomness()

    commit 78887832e76541f77169a24ac238fccb51059b63 upstream.

    add_early_randomness() is called by hwrng_register() when the
    hardware is added. If this hardware and its module are present
    at boot, and if there is no data available the boot hangs until
    data are available and can't be interrupted.

    For instance, in the case of virtio-rng, in some cases the host can be
    not able to provide enough entropy for all the guests.

    We can have two easy ways to reproduce the problem but they rely on
    misconfiguration of the hypervisor or the egd daemon:

    - if virtio-rng device is configured to connect to the egd daemon of the
    host but when the virtio-rng driver asks for data the daemon is not
    connected,

    - if virtio-rng device is configured to connect to the egd daemon of the
    host but the egd daemon doesn't provide data.

    The guest kernel will hang at boot until the virtio-rng driver provides
    enough data.

    To avoid that, call rng_get_data() in non-blocking mode (wait=0)
    from add_early_randomness().

    Signed-off-by: Laurent Vivier <lvivier@redhat.com>
    Fixes: d9e797261933 ("hwrng: add randomness to system from rng...")
    Cc: <stable@vger.kernel.org>
    Reviewed-by: Theodore Ts'o <tytso@mit.edu>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 5e311dddd22f4191c4b16927a4ed2f59e5feefaa
Author: Chao Yu <yuchao0@huawei.com>
Date:   Wed Sep 11 17:36:50 2019 +0800

    quota: fix wrong condition in is_quota_modification()

    commit 6565c182094f69e4ffdece337d395eb7ec760efc upstream.

    Quoted from
    commit 3da40c7b0898 ("ext4: only call ext4_truncate when size <= isize")

    " At LSF we decided that if we truncate up from isize we shouldn't trim
      fallocated blocks that were fallocated with KEEP_SIZE and are past the
     new i_size.  This patch fixes ext4 to do this. "

    And generic/092 of fstest have covered this case for long time, however
    is_quota_modification() didn't adjust based on that rule, so that in
    below condition, we will lose to quota block change:
    - fallocate blocks beyond EOF
    - remount
    - truncate(file_path, file_size)

    Fix it.

    Link: https://lore.kernel.org/r/20190911093650.35329-1-yuchao0@huawei.com
    Fixes: 3da40c7b0898 ("ext4: only call ext4_truncate when size <= isize")
    CC: stable@vger.kernel.org
    Signed-off-by: Chao Yu <yuchao0@huawei.com>
    Signed-off-by: Jan Kara <jack@suse.cz>
```

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

commit d92c8d1e740efcbf1bcde236ed8b8397aed88e4d
Author: Theodore Ts'o <tytso@mit.edu>
Date:   Fri Aug 23 22:38:00 2019 -0400

    ext4: fix punch hole for inline_data file systems

    commit c1e8220bd316d8ae8e524df39534b8a412a45d5e upstream.

    If a program attempts to punch a hole on an inline data file, we need
    to convert it to a normal file first.

    This was detected using ext4/032 using the adv configuration.  Simple
    reproducer:

    mke2fs -Fq -t ext4 -O inline_data /dev/vdc
    mount /vdc
    echo "" > /vdc/testfile
    xfs_io -c 'truncate 33554432' /vdc/testfile
    xfs_io -c 'fpunch 0 1048576' /vdc/testfile
    umount /vdc
    e2fsck -fy /dev/vdc

    Cc: stable@vger.kernel.org
    Signed-off-by: Theodore Ts'o <tytso@mit.edu>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 7587f2a2a93da4069d2514eda96bbe11a71595b9
Author: Rakesh Pandit <rakesh@tuxera.com>
Date:   Thu Aug 22 22:53:46 2019 -0400

    ext4: fix warning inside ext4_convert_unwritten_extents_endio

    commit e3d550c2c4f2f3dba469bc3c4b83d9332b4e99e1 upstream.

    Really enable warning when CONFIG_EXT4_DEBUG is set and fix missing
    first argument.  This was introduced in commit ff95ec22cd7f ("ext4:
    add warning to ext4_convert_unwritten_extents_endio") and splitting
    extents inside endio would trigger it.

    Fixes: ff95ec22cd7f ("ext4: add warning to ext4_convert_unwritten_extents_endio")
    Signed-off-by: Rakesh Pandit <rakesh@tuxera.com>
    Signed-off-by: Theodore Ts'o <tytso@mit.edu>
    Cc: stable@kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e2c0ac2eb750e50e233794b9160689ff7266b013
Author: Christophe Kerello <christophe.kerello@st.com>
Date:   Tue Jul 9 11:41:45 2019 +0200

    mtd: rawnand: stm32_fmc2: avoid warnings when building with W=1 option

    commit b410f4eb01a1950ed73ae40859d0978b1a924380 upstream.

    This patch solves warnings detected by setting W=1 when building.

    Warnings type detected:
    drivers/mtd/nand/raw/stm32_fmc2_nand.c: In function ‘stm32_fmc2_calc_timings’:
    drivers/mtd/nand/raw/stm32_fmc2_nand.c:1417:23: warning: comparison is
    always false due to limited range of data type [-Wtype-limits]
      else if (tims->twait > FMC2_PMEM_PATT_TIMING_MASK)

    Signed-off-by: Christophe Kerello <christophe.kerello@st.com>
    Cc: stable@vger.kernel.org
```

```
    Fixes: 2cd457f328c1 ("mtd: rawnand: stm32_fmc2: add STM32 FMC2 NAND flash controller
driver")
    Signed-off-by: Miquel Raynal <miquel.raynal@bootlin.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 5eee4abada05f9cb0cfe3f34fd173850244b532b
Author: Tony Camuso <tcamuso@redhat.com>
Date:   Thu Aug 22 08:24:53 2019 -0400

    ipmi: move message error checking to avoid deadlock

    commit 383035211c79d4d98481a09ad429b31c7dbf22bd upstream.

    V1->V2: in handle_one_rcv_msg, if data_size > 2, set requeue to zero and
            goto out instead of calling ipmi_free_msg.
            Kosuke Tatsukawa <tatsu@ab.jp.nec.com>

    In the source stack trace below, function set_need_watch tries to
    take out the same si_lock that was taken earlier by ipmi_thread.

    ipmi_thread() [drivers/char/ipmi/ipmi_si_intf.c:995]
     smi_event_handler() [drivers/char/ipmi/ipmi_si_intf.c:765]
      handle_transaction_done() [drivers/char/ipmi/ipmi_si_intf.c:555]
       deliver_recv_msg() [drivers/char/ipmi/ipmi_si_intf.c:283]
        ipmi_smi_msg_received() [drivers/char/ipmi/ipmi_msghandler.c:4503]
         intf_err_seq() [drivers/char/ipmi/ipmi_msghandler.c:1149]
          smi_remove_watch() [drivers/char/ipmi/ipmi_msghandler.c:999]
           set_need_watch() [drivers/char/ipmi/ipmi_si_intf.c:1066]

    Upstream commit e1891cffd4c4896a899337a243273f0e23c028df adds code to
    ipmi_smi_msg_received() to call smi_remove_watch() via intf_err_seq()
    and this seems to be causing the deadlock.

    commit e1891cffd4c4896a899337a243273f0e23c028df
    Author: Corey Minyard <cminyard@mvista.com>
    Date:   Wed Oct 24 15:17:04 2018 -0500
        ipmi: Make the smi watcher be disabled immediately when not needed

    The fix is to put all messages in the queue and move the message
    checking code out of ipmi_smi_msg_received and into handle_one_recv_msg,
    which processes the message checking after ipmi_thread releases its
    locks.

    Additionally,Kosuke Tatsukawa <tatsu@ab.jp.nec.com> reported that
    handle_new_recv_msgs calls ipmi_free_msg when handle_one_rcv_msg returns
    zero, so that the call to ipmi_free_msg in handle_one_rcv_msg introduced
    another panic when "ipmitool sensor list" was run in a loop. He
    submitted this part of the patch.

    +free_msg:
    +               requeue = 0;
    +               goto out;

    Reported by: Osamu Samukawa <osa-samukawa@tg.jp.nec.com>
    Characterized by: Kosuke Tatsukawa <tatsu@ab.jp.nec.com>
    Signed-off-by: Tony Camuso <tcamuso@redhat.com>
    Fixes: e1891cffd4c4 ("ipmi: Make the smi watcher be disabled immediately when not
needed")
    Cc: stable@vger.kernel.org # 5.1
    Signed-off-by: Corey Minyard <cminyard@mvista.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ff6d7192d95268df6b18e745a5bd50d04c648bc6
Author: Jan Kara <jack@suse.cz>
Date:   Thu Aug 29 09:04:12 2019 -0700
```

```
    xfs: Fix stale data exposure when readahead races with hole punch

    commit 40144e49ff84c3bd6bd091b58115257670be8803 upstream.

    Hole puching currently evicts pages from page cache and then goes on to
    remove blocks from the inode. This happens under both XFS_IOLOCK_EXCL
    and XFS_MMAPLOCK_EXCL which provides appropriate serialization with
    racing reads or page faults. However there is currently nothing that
    prevents readahead triggered by fadvise() or madvise() from racing with
    the hole punch and instantiating page cache page after hole punching has
    evicted page cache in xfs_flush_unmap_range() but before it has removed
    blocks from the inode. This page cache page will be mapping soon to be
    freed block and that can lead to returning stale data to userspace or
    even filesystem corruption.

    Fix the problem by protecting handling of readahead requests by
    XFS_IOLOCK_SHARED similarly as we protect reads.

    CC: stable@vger.kernel.org
    Link: https://lore.kernel.org/linux-
fsdevel/CAOQ4uxjQNmxqmtA_VbYW0Su9rKRk2zobJmahcyeaEVOFKVQ5dw@mail.gmail.com/
    Reported-by: Amir Goldstein <amir73il@gmail.com>
    Signed-off-by: Jan Kara <jack@suse.cz>
    Reviewed-by: Darrick J. Wong <darrick.wong@oracle.com>
    Signed-off-by: Darrick J. Wong <darrick.wong@oracle.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 46b4c5d9df6d94c45de2b42d264b848884ee25a6
Author: Jan Kara <jack@suse.cz>
Date:   Thu Aug 29 09:04:11 2019 -0700

    mm: Handle MADV_WILLNEED through vfs_fadvise()

    commit 692fe62433d4ca47605b39f7c416efd6679ba694 upstream.

    Currently handling of MADV_WILLNEED hint calls directly into readahead
    code. Handle it by calling vfs_fadvise() instead so that filesystem can
    use its ->fadvise() callback to acquire necessary locks or otherwise
    prepare for the request.

    Suggested-by: Amir Goldstein <amir73il@gmail.com>
    Reviewed-by: Boaz Harrosh <boazh@netapp.com>
    CC: stable@vger.kernel.org
    Signed-off-by: Jan Kara <jack@suse.cz>
    Reviewed-by: Darrick J. Wong <darrick.wong@oracle.com>
    Signed-off-by: Darrick J. Wong <darrick.wong@oracle.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit fbe7950a17c7786130d1e2bb0fa91a5256f7e186
Author: Jan Kara <jack@suse.cz>
Date:   Thu Aug 29 09:04:11 2019 -0700

    fs: Export generic_fadvise()

    commit cf1ea0592dbf109e7e7935b7d5b1a47a1ba04174 upstream.

    Filesystems will need to call this function from their fadvise handlers.

    CC: stable@vger.kernel.org
    Reviewed-by: Darrick J. Wong <darrick.wong@oracle.com>
    Signed-off-by: Jan Kara <jack@suse.cz>
    Signed-off-by: Darrick J. Wong <darrick.wong@oracle.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit 12bfec2132f8b1f8d474a7b4a62239fcd0e190d0
Author: Tetsuo Handa <penguin-kernel@I-love.SAKURA.ne.jp>
Date:   Mon Aug 26 22:13:25 2019 +0900

    /dev/mem: Bail out upon SIGKILL.

    commit 8619e5bdeee8b2c685d686281f2d2a6017c4bc15 upstream.

    syzbot found that a thread can stall for minutes inside read_mem() or
    write_mem() after that thread was killed by SIGKILL [1]. Reading from
    iomem areas of /dev/mem can be slow, depending on the hardware.
    While reading 2GB at one read() is legal, delaying termination of killed
    thread for minutes is bad. Thus, allow reading/writing /dev/mem and
    /dev/kmem to be preemptible and killable.

      [ 1335.912419][T20577] read_mem: sz=4096 count=2134565632
      [ 1335.943194][T20577] read_mem: sz=4096 count=2134561536
      [ 1335.978280][T20577] read_mem: sz=4096 count=2134557440
      [ 1336.011147][T20577] read_mem: sz=4096 count=2134553344
      [ 1336.041897][T20577] read_mem: sz=4096 count=2134549248

    Theoretically, reading/writing /dev/mem and /dev/kmem can become
    "interruptible". But this patch chose "killable". Future patch will make
    them "interruptible" so that we can revert to "killable" if some program
    regressed.

    [1] https://syzkaller.appspot.com/bug?id=a0e3436829698d5824231251fad9d8e998f94f5e

    Signed-off-by: Tetsuo Handa <penguin-kernel@I-love.SAKURA.ne.jp>
    Cc: stable <stable@vger.kernel.org>
    Reported-by: syzbot <syzbot+8ab2d0f39fb79fe6ca40@syzkaller.appspotmail.com>
    Link: https://lore.kernel.org/r/1566825205-10703-1-git-send-email-penguin-kernel@I-
love.SAKURA.ne.jp
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c8afd2daaadf3959a963968ac28115fa384cf770
Author: Denis Kenzior <denkenz@gmail.com>
Date:   Wed Aug 28 16:11:10 2019 -0500

    cfg80211: Purge frame registrations on iftype change

    commit c1d3ad84eae35414b6b334790048406bd6301b12 upstream.

    Currently frame registrations are not purged, even when changing the
    interface type.  This can lead to potentially weird situations where
    frames possibly not allowed on a given interface type remain registered
    due to the type switching happening after registration.

    The kernel currently relies on userspace apps to actually purge the
    registrations themselves, this is not something that the kernel should
    rely on.

    Add a call to cfg80211_mlme_purge_registrations() to forcefully remove
    any registrations left over prior to switching the iftype.

    Cc: stable@vger.kernel.org
    Signed-off-by: Denis Kenzior <denkenz@gmail.com>
    Link: https://lore.kernel.org/r/20190828211110.15005-1-denkenz@gmail.com
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 251eac21ab86599f30e7ec1bed1d40d5e5553a3e
Author: NeilBrown <neilb@suse.com>
Date:   Tue Aug 20 10:21:09 2019 +1000
```

```
    md: only call set_in_sync() when it is expected to succeed.

    commit 480523feae581ab714ba6610388a3b4619a2f695 upstream.

    Since commit 4ad23a976413 ("MD: use per-cpu counter for
    writes_pending"), set_in_sync() is substantially more expensive: it
    can wait for a full RCU grace period which can be 10s of milliseconds.

    So we should only call it when the cost is justified.

    md_check_recovery() currently calls set_in_sync() every time it finds
    anything to do (on non-external active arrays).  For an array
    performing resync or recovery, this will be quite often.
    Each call will introduce a delay to the md thread, which can noticeable
    affect IO submission latency.

    In md_check_recovery() we only need to call set_in_sync() if
    'safemode' was non-zero at entry, meaning that there has been not
    recent IO.  So we save this "safemode was nonzero" state, and only
    call set_in_sync() if it was non-zero.

    This measurably reduces mean and maximum IO submission latency during
    resync/recovery.

    Reported-and-tested-by: Jack Wang <jinpu.wang@cloud.ionos.com>
    Fixes: 4ad23a976413 ("MD: use per-cpu counter for writes_pending")
    Cc: stable@vger.kernel.org (v4.12+)
    Signed-off-by: NeilBrown <neilb@suse.com>
    Signed-off-by: Song Liu <songliubraving@fb.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 8258a5ee93849d58986b34f5e95dbf263776b19b
Author: NeilBrown <neilb@suse.com>
Date:   Tue Aug 20 10:21:09 2019 +1000

    md: don't report active array_state until after revalidate_disk() completes.

    commit 9d4b45d6af442237560d0bb5502a012baa5234b7 upstream.

    Until revalidate_disk() has completed, the size of a new md array will
    appear to be zero.
    So we shouldn't report, through array_state, that the array is active
    until that time.
    udev rules check array_state to see if the array is ready.  As soon as
    it appear to be zero, fsck can be run.  If it find the size to be
    zero, it will fail.

    So add a new flag to provide an interlock between do_md_run() and
    array_state_show().  This flag is set while do_md_run() is active and
    it prevents array_state_show() from reporting that the array is
    active.

    Before do_md_run() is called, ->pers will be NULL so array is
    definitely not active.
    After do_md_run() is called, revalidate_disk() will have run and the
    array will be completely ready.

    We also move various sysfs_notify*() calls out of md_run() into
    do_md_run() after MD_NOT_READY is cleared.  This ensure the
    information is ready before the notification is sent.

    Prior to v4.12, array_state_show() was called with the
    mddev->reconfig_mutex held, which provided exclusion with do_md_run().
```

```
    Note that MD_NOT_READY cleared twice.  This is deliberate to cover
    both success and error paths with minimal noise.

    Fixes: b7b17c9b67e5 ("md: remove mddev_lock() from md_attr_show()")
    Cc: stable@vger.kernel.org (v4.12++)
    Signed-off-by: NeilBrown <neilb@suse.com>
    Signed-off-by: Song Liu <songliubraving@fb.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 54ec75ff0f299c50262eb3396a35c22c9f22e3c0
Author: Xiao Ni <xni@redhat.com>
Date:   Mon Jul 8 10:14:32 2019 +0800

    md/raid6: Set R5_ReadError when there is read failure on parity disk

    commit 143f6e733b73051cd22dcb80951c6c929da413ce upstream.

    7471fb77ce4d ("md/raid6: Fix anomily when recovering a single device in
    RAID6.") avoids rereading P when it can be computed from other members.
    However, this misses the chance to re-write the right data to P. This
    patch sets R5_ReadError if the re-read fails.

    Also, when re-read is skipped, we also missed the chance to reset
    rdev->read_errors to 0. It can fail the disk when there are many read
    errors on P member disk (other disks don't have read error)

    V2: upper layer read request don't read parity/Q data. So there is no
    need to consider such situation.

    This is Reported-by: kbuild test robot <lkp@intel.com>

    Fixes: 7471fb77ce4d ("md/raid6: Fix anomily when recovering a single device in RAID6.")
    Cc: <stable@vger.kernel.org> #4.4+
    Signed-off-by: Xiao Ni <xni@redhat.com>
    Signed-off-by: Song Liu <songliubraving@fb.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ef5668207c4a14bcae91e5530aeaced83ff2a9da
Author: Jarkko Nikula <jarkko.nikula@linux.intel.com>
Date:   Thu Aug 22 11:32:00 2019 +0300

    ACPI / LPSS: Save/restore LPSS private registers also on Lynxpoint

    commit 57b3006492a4c11b2d4a772b5b2905d544a32037 upstream.

    My assumption in commit b53548f9d9e4 ("spi: pxa2xx: Remove LPSS private
    register restoring during resume") that Intel Lynxpoint and compatible
    based chipsets may not need LPSS private registers saving and restoring
    over suspend/resume cycle turned out to be false on Intel Broadwell.

    Curtis Malainey sent a patch bringing above change back and reported the
    LPSS SPI Chip Select control was lost over suspend/resume cycle on
    Broadwell machine.

    Instead of reverting above commit lets add LPSS private register
    saving/restoring also for all LPSS SPI, I2C and UART controllers on
    Lynxpoint and compatible chipset to make sure context is not lost in
    case nothing else preserves it like firmware or if LPSS is always on.

    Fixes: b53548f9d9e4 ("spi: pxa2xx: Remove LPSS private register restoring during
resume")
    Reported-by: Curtis Malainey <cujomalainey@chromium.org>
    Tested-by: Curtis Malainey <cujomalainey@chromium.org>
    Cc: 5.0+ <stable@vger.kernel.org> # 5.0+
    Signed-off-by: Jarkko Nikula <jarkko.nikula@linux.intel.com>
```

```
    Reviewed-by: Andy Shevchenko <andriy.shevchenko@linux.intel.com>
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 2ea3445b0a6079fb632c9943da454edd3171478a
Author: Benjamin Coddington <bcodding@redhat.com>
Date:   Mon Sep 16 07:59:37 2019 -0400

    SUNRPC: Fix buffer handling of GSS MIC without slack

    commit 5f1bc39979d868a0358c683864bec3fc8395440b upstream.

    The GSS Message Integrity Check data for krb5i may lie partially in the XDR
    reply buffer's pages and tail.  If so, we try to copy the entire MIC into
    free space in the tail.  But as the estimations of the slack space required
    for authentication and verification have improved there may be less free
    space in the tail to complete this copy -- see commit 2c94b8eca1a2
    ("SUNRPC: Use au_rslack when computing reply buffer size").  In fact, there
    may only be room in the tail for a single copy of the MIC, and not part of
    the MIC and then another complete copy.

    The real world failure reported is that `ls` of a directory on NFS may
    sometimes return -EIO, which can be traced back to xdr_buf_read_netobj()
    failing to find available free space in the tail to copy the MIC.

    Fix this by checking for the case of the MIC crossing the boundaries of
    head, pages, and tail. If so, shift the buffer until the MIC is contained
    completely within the pages or tail.  This allows the remainder of the
    function to create a sub buffer that directly address the complete MIC.

    Signed-off-by: Benjamin Coddington <bcodding@redhat.com>
    Cc: stable@vger.kernel.org # v5.1
    Reviewed-by: Chuck Lever <chuck.lever@oracle.com>
    Signed-off-by: Anna Schumaker <Anna.Schumaker@Netapp.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 92ea3e24ca7b02827ad40f498272eaba1c1507af
Author: Trond Myklebust <trondmy@gmail.com>
Date:   Tue Sep 10 13:01:35 2019 -0400

    SUNRPC: Dequeue the request from the receive queue while we're re-encoding

    commit cc204d01262a69218b2d0db5cdea371de85871d9 upstream.

    Ensure that we dequeue the request from the transport receive queue
    while we're re-encoding to prevent issues like use-after-free when
    we release the bvec.

    Fixes: 7536908982047 ("SUNRPC: Ensure the bvecs are reset when we re-encode...")
    Signed-off-by: Trond Myklebust <trond.myklebust@hammerspace.com>
    Cc: stable@vger.kernel.org # v4.20+
    Signed-off-by: Anna Schumaker <Anna.Schumaker@Netapp.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 773ded9d9853fc78c29a454250dacd0cc215798f
Author: Qu Wenruo <wqu@suse.com>
Date:   Wed Sep 25 10:13:27 2019 +0800

    btrfs: Fix a regression which we can't convert to SINGLE profile

    commit fab273595507a9ec7035df6d5512a955d80a80ba upstream.

    [BUG]
    With v5.3 kernel, we can't convert to SINGLE profile:
```

```
       # btrfs balance start -f -dconvert=single $mnt
       ERROR: error during balancing '/mnt/btrfs': Invalid argument
       # dmesg -t | tail
       validate_convert_profile: data profile=0x1000000000000 allowed=0x20 is_valid=1
  final=0x1000000000000 ret=1
       BTRFS error (device dm-3): balance: invalid convert data profile single

      [CAUSE]
      With the extra debug output added, it shows that the @allowed bit is
      lacking the special in-memory only SINGLE profile bit.

      Thus we fail at that (profile & ~allowed) check.

      This regression is caused by commit 081db89b13cb ("btrfs: use raid_attr
      to get allowed profiles for balance conversion") and the fact that we
      don't use any bit to indicate SINGLE profile on-disk, but uses special
      in-memory only bit to help distinguish different profiles.

      [FIX]
      Add that BTRFS_AVAIL_ALLOC_BIT_SINGLE to @allowed, so the code should be
      the same as it was and fix the regression.

      Reported-by: Chris Murphy <lists@colorremedies.com>
      Fixes: 081db89b13cb ("btrfs: use raid_attr to get allowed profiles for balance
  conversion")
      CC: stable@vger.kernel.org # 5.3+
      Reviewed-by: Anand Jain <anand.jain@oracle.com>
      Signed-off-by: Qu Wenruo <wqu@suse.com>
      Reviewed-by: David Sterba <dsterba@suse.com>
      Signed-off-by: David Sterba <dsterba@suse.com>
      Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

  commit 0e007f8e0dba65bb03c3ea0531fde29aa484ab00
  Author: Filipe Manana <fdmanana@suse.com>
  Date:   Tue Sep 24 10:49:54 2019 +0100

      Btrfs: fix race setting up and completing qgroup rescan workers

      commit 13fc1d271a2e3ab8a02071e711add01fab9271f6 upstream.

      There is a race between setting up a qgroup rescan worker and completing
      a qgroup rescan worker that can lead to callers of the qgroup rescan wait
      ioctl to either not wait for the rescan worker to complete or to hang
      forever due to missing wake ups. The following diagram shows a sequence
      of steps that illustrates the race.

              CPU 1                                                   CPU 2
  CPU 3

       btrfs_ioctl_quota_rescan()
        btrfs_qgroup_rescan()
         qgroup_rescan_init()
          mutex_lock(&fs_info->qgroup_rescan_lock)
          spin_lock(&fs_info->qgroup_lock)

          fs_info->qgroup_flags |=
            BTRFS_QGROUP_STATUS_FLAG_RESCAN

          init_completion(
            &fs_info->qgroup_rescan_completion)

          fs_info->qgroup_rescan_running = true

          mutex_unlock(&fs_info->qgroup_rescan_lock)
          spin_unlock(&fs_info->qgroup_lock)
```

```
        btrfs_init_work()
         --> starts the worker

                                                          btrfs_qgroup_rescan_worker()
                                                           mutex_lock(&fs_info-
>qgroup_rescan_lock)

                                                           fs_info->qgroup_flags &=

~BTRFS_QGROUP_STATUS_FLAG_RESCAN

                                                           mutex_unlock(&fs_info-
>qgroup_rescan_lock)

                                                           starts transaction, updates
qgroup status
                                                           item, etc

btrfs_ioctl_quota_rescan()

btrfs_qgroup_rescan()

qgroup_rescan_init()

mutex_lock(&fs_info->qgroup_rescan_lock)

spin_lock(&fs_info->qgroup_lock)

fs_info->qgroup_flags |=

BTRFS_QGROUP_STATUS_FLAG_RESCAN

init_completion(

&fs_info->qgroup_rescan_completion)

fs_info->qgroup_rescan_running = true

mutex_unlock(&fs_info->qgroup_rescan_lock)

spin_unlock(&fs_info->qgroup_lock)

btrfs_init_work()

--> starts another worker

                                                           mutex_lock(&fs_info-
>qgroup_rescan_lock)

                                                           fs_info->qgroup_rescan_running
= false

                                                           mutex_unlock(&fs_info-
>qgroup_rescan_lock)

                                                           complete_all(&fs_info-
>qgroup_rescan_completion)
```

```
    Before the rescan worker started by the task at CPU 3 completes, if
    another task calls btrfs_ioctl_quota_rescan(), it will get -EINPROGRESS
    because the flag BTRFS_QGROUP_STATUS_FLAG_RESCAN is set at
    fs_info->qgroup_flags, which is expected and correct behaviour.

    However if other task calls btrfs_ioctl_quota_rescan_wait() before the
    rescan worker started by the task at CPU 3 completes, it will return
    immediately without waiting for the new rescan worker to complete,
    because fs_info->qgroup_rescan_running is set to false by CPU 2.

    This race is making test case btrfs/171 (from fstests) to fail often:

      btrfs/171 9s ... - output mismatch (see
/home/fdmanana/git/hub/xfstests/results//btrfs/171.out.bad)
    #      --- tests/btrfs/171.out     2018-09-16 21:30:48.505104287 +0100
    #      +++ /home/fdmanana/git/hub/xfstests/results//btrfs/171.out.bad      2019-09-19
02:01:36.938486039 +0100
    #      @@ -1,2 +1,3 @@
    #       QA output created by 171
    #      +ERROR: quota rescan failed: Operation now in progress
    #       Silence is golden
    #      ...
    #      (Run 'diff -u /home/fdmanana/git/hub/xfstests/tests/btrfs/171.out
/home/fdmanana/git/hub/xfstests/results//btrfs/171.out.bad'  to see the entire diff)

    That is because the test calls the btrfs-progs commands "qgroup quota
    rescan -w", "qgroup assign" and "qgroup remove" in a sequence that makes
    calls to the rescan start ioctl fail with -EINPROGRESS (note the "btrfs"
    commands 'qgroup assign' and 'qgroup remove' often call the rescan start
    ioctl after calling the qgroup assign ioctl,
    btrfs_ioctl_qgroup_assign()), since previous waits didn't actually wait
    for a rescan worker to complete.

    Another problem the race can cause is missing wake ups for waiters,
    since the call to complete_all() happens outside a critical section and
    after clearing the flag BTRFS_QGROUP_STATUS_FLAG_RESCAN. In the sequence
    diagram above, if we have a waiter for the first rescan task (executed
    by CPU 2), then fs_info->qgroup_rescan_completion.wait is not empty, and
    if after the rescan worker clears BTRFS_QGROUP_STATUS_FLAG_RESCAN and
    before it calls complete_all() against
    fs_info->qgroup_rescan_completion, the task at CPU 3 calls
    init_completion() against fs_info->qgroup_rescan_completion which
    re-initilizes its wait queue to an empty queue, therefore causing the
    rescan worker at CPU 2 to call complete_all() against an empty queue,
    never waking up the task waiting for that rescan worker.

    Fix this by clearing BTRFS_QGROUP_STATUS_FLAG_RESCAN and setting
    fs_info->qgroup_rescan_running to false in the same critical section,
    delimited by the mutex fs_info->qgroup_rescan_lock, as well as doing the
    call to complete_all() in that same critical section. This gives the
    protection needed to avoid rescan wait ioctl callers not waiting for a
    running rescan worker and the lost wake ups problem, since setting that
    rescan flag and boolean as well as initializing the wait queue is done
    already in a critical section delimited by that mutex (at
    qgroup_rescan_init()).

    Fixes: 57254b6ebce4ce ("Btrfs: add ioctl to wait for qgroup rescan completion")
    Fixes: d2c609b834d62f ("btrfs: properly track when rescan worker is running")
    CC: stable@vger.kernel.org # 4.4+
    Reviewed-by: Josef Bacik <josef@toxicpanda.com>
    Signed-off-by: Filipe Manana <fdmanana@suse.com>
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 88b870760d1912a1e4f94bae7ac9965f9a0da91e
```

```
Author: Qu Wenruo <wqu@suse.com>
Date:   Mon Sep 16 20:02:39 2019 +0800

    btrfs: qgroup: Fix reserved data space leak if we have multiple reserve calls

    commit d4e204948fe3e0dc8e1fbf3f8f3290c9c2823be3 upstream.

    [BUG]
    The following script can cause btrfs qgroup data space leak:

      mkfs.btrfs -f $dev
      mount $dev -o nospace_cache $mnt

      btrfs subv create $mnt/subv
      btrfs quota en $mnt
      btrfs quota rescan -w $mnt
      btrfs qgroup limit 128m $mnt/subv

      for (( i = 0; i < 3; i++)); do
              # Create 3 64M holes for latter fallocate to fail
              truncate -s 192m $mnt/subv/file
              xfs_io -c "pwrite 64m 4k" $mnt/subv/file > /dev/null
              xfs_io -c "pwrite 128m 4k" $mnt/subv/file > /dev/null
              sync

              # it's supposed to fail, and each failure will leak at least 64M
              # data space
              xfs_io -f -c "falloc 0 192m" $mnt/subv/file &> /dev/null
              rm $mnt/subv/file
              sync
      done

      # Shouldn't fail after we removed the file
      xfs_io -f -c "falloc 0 64m" $mnt/subv/file

    [CAUSE]
    Btrfs qgroup data reserve code allow multiple reservations to happen on
    a single extent_changeset:
    E.g:
            btrfs_qgroup_reserve_data(inode, &data_reserved, 0, SZ_1M);
            btrfs_qgroup_reserve_data(inode, &data_reserved, SZ_1M, SZ_2M);
            btrfs_qgroup_reserve_data(inode, &data_reserved, 0, SZ_4M);

    Btrfs qgroup code has its internal tracking to make sure we don't
    double-reserve in above example.

    The only pattern utilizing this feature is in the main while loop of
    btrfs_fallocate() function.

    However btrfs_qgroup_reserve_data()'s error handling has a bug in that
    on error it clears all ranges in the io_tree with EXTENT_QGROUP_RESERVED
    flag but doesn't free previously reserved bytes.

    This bug has a two fold effect:
    - Clearing EXTENT_QGROUP_RESERVED ranges
      This is the correct behavior, but it prevents
      btrfs_qgroup_check_reserved_leak() to catch the leakage as the
      detector is purely EXTENT_QGROUP_RESERVED flag based.

    - Leak the previously reserved data bytes.

    The bug manifests when N calls to btrfs_qgroup_reserve_data are made and
    the last one fails, leaking space reserved in the previous ones.

    [FIX]
```

```
    Also free previously reserved data bytes when btrfs_qgroup_reserve_data
    fails.

    Fixes: 524725537023 ("btrfs: qgroup: Introduce btrfs_qgroup_reserve_data function")
    CC: stable@vger.kernel.org # 4.4+
    Signed-off-by: Qu Wenruo <wqu@suse.com>
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 53daf6b7a0117b0d8aa54f2e65878fb7f4477bd0
Author: Qu Wenruo <wqu@suse.com>
Date:   Mon Sep 16 20:02:38 2019 +0800

    btrfs: qgroup: Fix the wrong target io_tree when freeing reserved data space

    commit bab32fc069ce8829c416e8737c119f62a57970f9 upstream.

    [BUG]
    Under the following case with qgroup enabled, if some error happened
    after we have reserved delalloc space, then in error handling path, we
    could cause qgroup data space leakage:

    From btrfs_truncate_block() in inode.c:

            ret = btrfs_delalloc_reserve_space(inode, &data_reserved,
                                               block_start, blocksize);
            if (ret)
                    goto out;

     again:
            page = find_or_create_page(mapping, index, mask);
            if (!page) {
                    btrfs_delalloc_release_space(inode, data_reserved,
                                                 block_start, blocksize, true);
                    btrfs_delalloc_release_extents(BTRFS_I(inode), blocksize, true);
                    ret = -ENOMEM;
                    goto out;
            }

    [CAUSE]
    In the above case, btrfs_delalloc_reserve_space() will call
    btrfs_qgroup_reserve_data() and mark the io_tree range with
    EXTENT_QGROUP_RESERVED flag.

    In the error handling path, we have the following call stack:
    btrfs_delalloc_release_space()
    |- btrfs_free_reserved_data_space()
       |- btrsf_qgroup_free_data()
          |- __btrfs_qgroup_release_data(reserved=@reserved, free=1)
             |- qgroup_free_reserved_data(reserved=@reserved)
                |- clear_record_extent_bits();
                |- freed += changeset.bytes_changed;

    However due to a completion bug, qgroup_free_reserved_data() will clear
    EXTENT_QGROUP_RESERVED flag in BTRFS_I(inode)->io_failure_tree, other
    than the correct BTRFS_I(inode)->io_tree.
    Since io_failure_tree is never marked with that flag,
    btrfs_qgroup_free_data() will not free any data reserved space at all,
    causing a leakage.

    This type of error handling can only be triggered by errors outside of
    qgroup code. So EDQUOT error from qgroup can't trigger it.

    [FIX]
    Fix the wrong target io_tree.
```

```
    Reported-by: Josef Bacik <josef@toxicpanda.com>
    Fixes: bc42bda22345 ("btrfs: qgroup: Fix qgroup reserved space underflow by only freeing
reserved ranges")
    CC: stable@vger.kernel.org # 4.14+
    Reviewed-by: Nikolay Borisov <nborisov@suse.com>
    Signed-off-by: Qu Wenruo <wqu@suse.com>
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 79b8b06b726c17525140fa5b1376a8f4b8a78249
Author: Dennis Zhou <dennis@kernel.org>
Date:   Fri Sep 13 14:54:07 2019 +0100

    btrfs: adjust dirty_metadata_bytes after writeback failure of extent buffer

    commit eb5b64f142504a597d67e2109d603055ff765e52 upstream.

    Before, if a eb failed to write out, we would end up triggering a
    BUG_ON(). As of f4340622e0226 ("btrfs: extent_io: Move the BUG_ON() in
    flush_write_bio() one level up"), we no longer BUG_ON(), so we should
    make life consistent and add back the unwritten bytes to
    dirty_metadata_bytes.

    Fixes: f4340622e022 ("btrfs: extent_io: Move the BUG_ON() in flush_write_bio() one level
up")
    CC: stable@vger.kernel.org # 5.2+
    Reviewed-by: Filipe Manana <fdmanana@kernel.org>
    Signed-off-by: Dennis Zhou <dennis@kernel.org>
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 8f1bc5f02cc00f7ba10555048b790a1d6864e4b1
Author: Nikolay Borisov <nborisov@suse.com>
Date:   Wed Sep 4 19:33:58 2019 +0300

    btrfs: Relinquish CPUs in btrfs_compare_trees

    commit 6af112b11a4bc1b560f60a618ac9c1dcefe9836e upstream.

    When doing any form of incremental send the parent and the child trees
    need to be compared via btrfs_compare_trees. This  can result in long
    loop chains without ever relinquishing the CPU. This causes softlockup
    detector to trigger when comparing trees with a lot of items. Example
    report:

    watchdog: BUG: soft lockup - CPU#0 stuck for 24s! [snapperd:16153]
    CPU: 0 PID: 16153 Comm: snapperd Not tainted 5.2.9-1-default #1 openSUSE Tumbleweed
(unreleased)
    Hardware name: QEMU KVM Virtual Machine, BIOS 0.0.0 02/06/2015
    pstate: 40000005 (nZcv daif -PAN -UAO)
    pc : __ll_sc_arch_atomic_sub_return+0x14/0x20
    lr : btrfs_release_extent_buffer_pages+0xe0/0x1e8 [btrfs]
    sp : ffff00001273b7e0
    Call trace:
     __ll_sc_arch_atomic_sub_return+0x14/0x20
     release_extent_buffer+0xdc/0x120 [btrfs]
     free_extent_buffer.part.0+0xb0/0x118 [btrfs]
     free_extent_buffer+0x24/0x30 [btrfs]
     btrfs_release_path+0x4c/0xa0 [btrfs]
     btrfs_free_path.part.0+0x20/0x40 [btrfs]
     btrfs_free_path+0x24/0x30 [btrfs]
     get_inode_info+0xa8/0xf8 [btrfs]
     finish_inode_if_needed+0xe0/0x6d8 [btrfs]
     changed_cb+0x9c/0x410 [btrfs]
```

```
       btrfs_compare_trees+0x284/0x648 [btrfs]
       send_subvol+0x33c/0x520 [btrfs]
       btrfs_ioctl_send+0x8a0/0xaf0 [btrfs]
       btrfs_ioctl+0x199c/0x2288 [btrfs]
       do_vfs_ioctl+0x4b0/0x820
       ksys_ioctl+0x84/0xb8
       __arm64_sys_ioctl+0x28/0x38
       el0_svc_common.constprop.0+0x7c/0x188
       el0_svc_handler+0x34/0x90
       el0_svc+0x8/0xc

    Fix this by adding a call to cond_resched at the beginning of the main
    loop in btrfs_compare_trees.

    Fixes: 7069830a9e38 ("Btrfs: add btrfs_compare_trees function")
    CC: stable@vger.kernel.org # 4.4+
    Reviewed-by: Johannes Thumshirn <jthumshirn@suse.de>
    Signed-off-by: Nikolay Borisov <nborisov@suse.com>
    Reviewed-by: David Sterba <dsterba@suse.com>
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f59d80e2f12b695d38e646f00de1e46e123dfcc9
Author: Filipe Manana <fdmanana@suse.com>
Date:   Mon Aug 12 19:14:29 2019 +0100

    Btrfs: fix use-after-free when using the tree modification log

    commit efad8a853ad2057f96664328a0d327a05ce39c76 upstream.

    At ctree.c:get_old_root(), we are accessing a root's header owner field
    after we have freed the respective extent buffer. This results in an
    use-after-free that can lead to crashes, and when CONFIG_DEBUG_PAGEALLOC
    is set, results in a stack trace like the following:

      [ 3876.799331] stack segment: 0000 [#1] SMP DEBUG_PAGEALLOC PTI
      [ 3876.799363] CPU: 0 PID: 15436 Comm: pool Not tainted 5.3.0-rc3-btrfs-next-54 #1
      [ 3876.799385] Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS rel-1.12.0-
0-ga698c8995f-prebuilt.qemu.org 04/01/2014
      [ 3876.799433] RIP: 0010:btrfs_search_old_slot+0x652/0xd80 [btrfs]
      (...)
      [ 3876.799502] RSP: 0018:ffff9f08c1a2f9f0 EFLAGS: 00010286
      [ 3876.799518] RAX: ffff8dd300000000 RBX: ffff8dd85a7a9348 RCX: 000000038da26000
      [ 3876.799538] RDX: 0000000000000000 RSI: ffffe522ce368980 RDI: 0000000000000246
      [ 3876.799559] RBP: dae1922adadad000 R08: 0000000008020000 R09: ffffe522c0000000
      [ 3876.799579] R10: ffff8dd57fd788c8 R11: 000000007511b030 R12: ffff8dd781ddc000
      [ 3876.799599] R13: ffff8dd9e6240578 R14: ffff8dd6896f7a88 R15: ffff8dd688cf90b8
      [ 3876.799620] FS:  00007f23ddd97700(0000) GS:ffff8dda20200000(0000)
knlGS:0000000000000000
      [ 3876.799643] CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
      [ 3876.799660] CR2: 00007f23d4024000 CR3: 0000000710bb0005 CR4: 00000000003606f0
      [ 3876.799682] DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000
      [ 3876.799703] DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400
      [ 3876.799723] Call Trace:
      [ 3876.799735]  ? do_raw_spin_unlock+0x49/0xc0
      [ 3876.799749]  ? _raw_spin_unlock+0x24/0x30
      [ 3876.799779]  resolve_indirect_refs+0x1eb/0xc80 [btrfs]
      [ 3876.799810]  find_parent_nodes+0x38d/0x1180 [btrfs]
      [ 3876.799841]  btrfs_check_shared+0x11a/0x1d0 [btrfs]
      [ 3876.799870]  ? extent_fiemap+0x598/0x6e0 [btrfs]
      [ 3876.799895]  extent_fiemap+0x598/0x6e0 [btrfs]
      [ 3876.799913]  do_vfs_ioctl+0x45a/0x700
      [ 3876.799926]  ksys_ioctl+0x70/0x80
      [ 3876.799938]  ? trace_hardirqs_off_thunk+0x1a/0x20
      [ 3876.799953]  __x64_sys_ioctl+0x16/0x20
```

```
       [ 3876.799965]  do_syscall_64+0x62/0x220
       [ 3876.799977]  entry_SYSCALL_64_after_hwframe+0x49/0xbe
       [ 3876.799993] RIP: 0033:0x7f23e0013dd7
       (...)
       [ 3876.800056] RSP: 002b:00007f23ddd96ca8 EFLAGS: 00000246 ORIG_RAX: 0000000000000010
       [ 3876.800078] RAX: ffffffffffffffda RBX: 00007f23d80210f8 RCX: 00007f23e0013dd7
       [ 3876.800099] RDX: 00007f23d80210f8 RSI: 00000000c020660b RDI: 0000000000000003
       [ 3876.800626] RBP: 000055fa2a2a2440 R08: 0000000000000000 R09: 00007f23ddd96d7c
       [ 3876.801143] R10: 00007f23d8022000 R11: 0000000000000246 R12: 00007f23ddd96d80
       [ 3876.801662] R13: 00007f23ddd96d78 R14: 00007f23d80210f0 R15: 00007f23ddd96d80
       (...)
       [ 3876.805107] ---[ end trace e53161e179ef04f9 ]---

    Fix that by saving the root's header owner field into a local variable
    before freeing the root's extent buffer, and then use that local variable
    when needed.

    Fixes: 30b0463a9394d9 ("Btrfs: fix accessing the root pointer in tree mod log
functions")
    CC: stable@vger.kernel.org # 3.10+
    Reviewed-by: Nikolay Borisov <nborisov@suse.com>
    Reviewed-by: Anand Jain <anand.jain@oracle.com>
    Signed-off-by: Filipe Manana <fdmanana@suse.com>
    Reviewed-by: David Sterba <dsterba@suse.com>
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit d6122f68dd50650b7468107e2103bb3e2084e7a0
Author: Christophe Leroy <christophe.leroy@c-s.fr>
Date:   Wed Aug 21 15:05:55 2019 +0000

    btrfs: fix allocation of free space cache v1 bitmap pages

    commit 3acd48507dc43eeeb0a1fe965b8bad91cab904a7 upstream.

    Various notifications of type "BUG kmalloc-4096 () : Redzone
    overwritten" have been observed recently in various parts of the kernel.
    After some time, it has been made a relation with the use of BTRFS
    filesystem and with SLUB_DEBUG turned on.

    [   22.809700] BUG kmalloc-4096 (Tainted: G        W        ): Redzone overwritten

    [   22.810286] INFO: 0xbe1a5921-0xfbfc06cd. First byte 0x0 instead of 0xcc
    [   22.810866] INFO: Allocated in __load_free_space_cache+0x588/0x780 [btrfs] age=22
cpu=0 pid=224
    [   22.811193]  __slab_alloc.constprop.26+0x44/0x70
    [   22.811345]  kmem_cache_alloc_trace+0xf0/0x2ec
    [   22.811588]  __load_free_space_cache+0x588/0x780 [btrfs]
    [   22.811848]  load_free_space_cache+0xf4/0x1b0 [btrfs]
    [   22.812090]  cache_block_group+0x1d0/0x3d0 [btrfs]
    [   22.812321]  find_free_extent+0x680/0x12a4 [btrfs]
    [   22.812549]  btrfs_reserve_extent+0xec/0x220 [btrfs]
    [   22.812785]  btrfs_alloc_tree_block+0x178/0x5f4 [btrfs]
    [   22.813032]  __btrfs_cow_block+0x150/0x5d4 [btrfs]
    [   22.813262]  btrfs_cow_block+0x194/0x298 [btrfs]
    [   22.813484]  commit_cowonly_roots+0x44/0x294 [btrfs]
    [   22.813718]  btrfs_commit_transaction+0x63c/0xc0c [btrfs]
    [   22.813973]  close_ctree+0xf8/0x2a4 [btrfs]
    [   22.814107]  generic_shutdown_super+0x80/0x110
    [   22.814250]  kill_anon_super+0x18/0x30
    [   22.814437]  btrfs_kill_super+0x18/0x90 [btrfs]
    [   22.814590] INFO: Freed in proc_cgroup_show+0xc0/0x248 age=41 cpu=0 pid=83
    [   22.814841]  proc_cgroup_show+0xc0/0x248
    [   22.814967]  proc_single_show+0x54/0x98
    [   22.815086]  seq_read+0x278/0x45c
```

```
     [   22.815190]  __vfs_read+0x28/0x17c
     [   22.815289]  vfs_read+0xa8/0x14c
     [   22.815381]  ksys_read+0x50/0x94
     [   22.815475]  ret_from_syscall+0x0/0x38

     Commit 69d2480456d1 ("btrfs: use copy_page for copying pages instead of
     memcpy") changed the way bitmap blocks are copied. But allthough bitmaps
     have the size of a page, they were allocated with kzalloc().

     Most of the time, kzalloc() allocates aligned blocks of memory, so
     copy_page() can be used. But when some debug options like SLAB_DEBUG are
     activated, kzalloc() may return unaligned pointer.

     On powerpc, memcpy(), copy_page() and other copying functions use
     'dcbz' instruction which provides an entire zeroed cacheline to avoid
     memory read when the intention is to overwrite a full line. Functions
     like memcpy() are writen to care about partial cachelines at the start
     and end of the destination, but copy_page() assumes it gets pages. As
     pages are naturally cache aligned, copy_page() doesn't care about
     partial lines. This means that when copy_page() is called with a
     misaligned pointer, a few leading bytes are zeroed.

     To fix it, allocate bitmaps through kmem_cache instead of using kzalloc()
     The cache pool is created with PAGE_SIZE alignment constraint.

     Reported-by: Erhard F. <erhard_f@mailbox.org>
     Bugzilla: https://bugzilla.kernel.org/show_bug.cgi?id=204371
     Fixes: 69d2480456d1 ("btrfs: use copy_page for copying pages instead of memcpy")
     Cc: stable@vger.kernel.org # 4.19+
     Signed-off-by: Christophe Leroy <christophe.leroy@c-s.fr>
     Reviewed-by: David Sterba <dsterba@suse.com>
     [ rename to btrfs_free_space_bitmap ]
     Signed-off-by: David Sterba <dsterba@suse.com>
     Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit bf8c7875650b6857e17eb0d42fc4b768b60827a1
Author: Mark Salyzyn <salyzyn@android.com>
Date:   Thu Aug 29 11:30:14 2019 -0700

     ovl: filter of trusted xattr results in audit

     commit 5c2e9f346b815841f9bed6029ebcb06415caf640 upstream.

     When filtering xattr list for reading, presence of trusted xattr
     results in a security audit log.  However, if there is other content
     no errno will be set, and if there isn't, the errno will be -ENODATA
     and not -EPERM as is usually associated with a lack of capability.
     The check does not block the request to list the xattrs present.

     Switch to ns_capable_noaudit to reflect a more appropriate check.

     Signed-off-by: Mark Salyzyn <salyzyn@android.com>
     Cc: linux-security-module@vger.kernel.org
     Cc: kernel-team@android.com
     Cc: stable@vger.kernel.org # v3.18+
     Fixes: a082c6f680da ("ovl: filter trusted xattr for non-admin")
     Signed-off-by: Miklos Szeredi <mszeredi@redhat.com>
     Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit df6b54da8d4df261a60bd30f25447141e53c6b90
Author: Ding Xiang <dingxiang@cmss.chinamobile.com>
Date:   Mon Sep 9 16:29:56 2019 +0800

     ovl: Fix dereferencing possible ERR_PTR()
```

```
    commit 97f024b9171e74c4443bbe8a8dce31b917f97ac5 upstream.

    if ovl_encode_real_fh() fails, no memory was allocated
    and the error in the error-valued pointer should be returned.

    Fixes: 9b6faee07470 ("ovl: check ERR_PTR() return value from ovl_encode_fh()")
    Signed-off-by: Ding Xiang <dingxiang@cmss.chinamobile.com>
    Cc: <stable@vger.kernel.org> # v4.16+
    Signed-off-by: Miklos Szeredi <mszeredi@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3cf65975b7537efa6e5faf760b30804f60905ff9
Author: Steve French <stfrench@microsoft.com>
Date:   Sun Sep 22 00:55:46 2019 -0500

    smb3: fix leak in "open on server" perf counter

    commit d2f15428d6a0ebfc0edc364094d7c4a2de7037ed upstream.

    We were not bumping up the "open on server" (num_remote_opens)
    counter (in some cases) on opens of the share root so
    could end up showing as a negative value.

    CC: Stable <stable@vger.kernel.org>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Reviewed-by: Pavel Shilovsky <pshilov@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit d6af8dcf1fc1940060c953323586bb8956fde7d6
Author: Steve French <stfrench@microsoft.com>
Date:   Thu Sep 12 17:52:54 2019 -0500

    smb3: fix unmount hang in open_shroot

    commit 96d9f7ed00b86104bf03adeffc8980897e9694ab upstream.

    An earlier patch "CIFS: fix deadlock in cached root handling"
    did not completely address the deadlock in open_shroot. This
    patch addresses the deadlock.

    In testing the recent patch:
      smb3: improve handling of share deleted (and share recreated)
    we were able to reproduce the open_shroot deadlock to one
    of the target servers in unmount in a delete share scenario.

    Fixes: 7e5a70ad88b1e ("CIFS: fix deadlock in cached root handling")

    This is version 2 of this patch. An earlier version of this
    patch "smb3: fix unmount hang in open_shroot" had a problem
    found by Dan.

    Reported-by: kbuild test robot <lkp@intel.com>
    Reported-by: Dan Carpenter <dan.carpenter@oracle.com>

    Suggested-by: Pavel Shilovsky <pshilov@microsoft.com>
    Reviewed-by: Pavel Shilovsky <pshilov@microsoft.com>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    CC: Aurelien Aptel <aaptel@suse.com>
    CC: Stable <stable@vger.kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 777ffe16167f6b53f206d1cea6ea76179088fa23
Author: Steve French <stfrench@microsoft.com>
Date:   Wed Sep 11 21:46:20 2019 -0500
```

```
    smb3: allow disabling requesting leases

    commit 3e7a02d47872081f4b6234a9f72500f1d10f060c upstream.

    In some cases to work around server bugs or performance
    problems it can be helpful to be able to disable requesting
    SMB2.1/SMB3 leases on a particular mount (not to all servers
    and all shares we are mounted to). Add new mount parm
    "nolease" which turns off requesting leases on directory
    or file opens.  Currently the only way to disable leases is
    globally through a module load parameter. This is more
    granular.

    Suggested-by: Pavel Shilovsky <pshilov@microsoft.com>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Reviewed-by: Ronnie Sahlberg <lsahlber@redhat.com>
    Reviewed-by: Pavel Shilovsky <pshilov@microsoft.com>
    CC: Stable <stable@vger.kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b780f0f77850e6b0ea7784885f65009db2c0ef19
Author: Yufen Yu <yuyufen@huawei.com>
Date:   Fri Sep 27 16:19:55 2019 +0800

    block: fix null pointer dereference in blk_mq_rq_timed_out()

    commit 8d6996630c03d7ceeabe2611378fea5ca1c3f1b3 upstream.

    We got a null pointer deference BUG_ON in blk_mq_rq_timed_out()
    as following:

    [  108.825472] BUG: kernel NULL pointer dereference, address: 0000000000000040
    [  108.827059] PGD 0 P4D 0
    [  108.827313] Oops: 0000 [#1] SMP PTI
    [  108.827657] CPU: 6 PID: 198 Comm: kworker/6:1H Not tainted 5.3.0-rc8+ #431
    [  108.829503] Workqueue: kblockd blk_mq_timeout_work
    [  108.829913] RIP: 0010:blk_mq_check_expired+0x258/0x330
    [  108.838191] Call Trace:
    [  108.838406]  bt_iter+0x74/0x80
    [  108.838665]  blk_mq_queue_tag_busy_iter+0x204/0x450
    [  108.839074]  ? __switch_to_asm+0x34/0x70
    [  108.839405]  ? blk_mq_stop_hw_queue+0x40/0x40
    [  108.839823]  ? blk_mq_stop_hw_queue+0x40/0x40
    [  108.840273]  ? syscall_return_via_sysret+0xf/0x7f
    [  108.840732]  blk_mq_timeout_work+0x74/0x200
    [  108.841151]  process_one_work+0x297/0x680
    [  108.841550]  worker_thread+0x29c/0x6f0
    [  108.841926]  ? rescuer_thread+0x580/0x580
    [  108.842344]  kthread+0x16a/0x1a0
    [  108.842666]  ? kthread_flush_work+0x170/0x170
    [  108.843100]  ret_from_fork+0x35/0x40

    The bug is caused by the race between timeout handle and completion for
    flush request.

    When timeout handle function blk_mq_rq_timed_out() try to read
    'req->q->mq_ops', the 'req' have completed and reinitiated by next
    flush request, which would call blk_rq_init() to clear 'req' as 0.

    After commit 12f5b93145 ("blk-mq: Remove generation seqeunce"),
    normal requests lifetime are protected by refcount. Until 'rq->ref'
    drop to zero, the request can really be free. Thus, these requests
    cannot been reused before timeout handle finish.

    However, flush request has defined .end_io and rq->end_io() is still
```

```
    called even if 'rq->ref' doesn't drop to zero. After that, the 'flush_rq'
    can be reused by the next flush request handle, resulting in null
    pointer deference BUG ON.

    We fix this problem by covering flush request with 'rq->ref'.
    If the refcount is not zero, flush_end_io() return and wait the
    last holder recall it. To record the request status, we add a new
    entry 'rq_status', which will be used in flush_end_io().

    Cc: Christoph Hellwig <hch@infradead.org>
    Cc: Keith Busch <keith.busch@intel.com>
    Cc: Bart Van Assche <bvanassche@acm.org>
    Cc: stable@vger.kernel.org # v4.18+
    Reviewed-by: Ming Lei <ming.lei@redhat.com>
    Reviewed-by: Bob Liu <bob.liu@oracle.com>
    Signed-off-by: Yufen Yu <yuyufen@huawei.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

    -------
    v2:
     - move rq_status from struct request to struct blk_flush_queue
    v3:
     - remove unnecessary '{}' pair.
    v4:
     - let spinlock to protect 'fq->rq_status'
    v5:
     - move rq_status after flush_running_idx member of struct blk_flush_queue
    Signed-off-by: Jens Axboe <axboe@kernel.dk>

commit 1e04eb03877c3e0a38c1be1845be97074a1198b6
Author: Damien Le Moal <damien.lemoal@wdc.com>
Date:   Wed Aug 28 13:40:20 2019 +0900

    block: mq-deadline: Fix queue restart handling

    commit cb8acabbe33b110157955a7425ee876fb81e6bbc upstream.

    Commit 7211aef86f79 ("block: mq-deadline: Fix write completion
    handling") added a call to blk_mq_sched_mark_restart_hctx() in
    dd_dispatch_request() to make sure that write request dispatching does
    not stall when all target zones are locked. This fix left a subtle race
    when a write completion happens during a dispatch execution on another
    CPU:

    CPU 0: Dispatch                 CPU1: write completion

    dd_dispatch_request()
        lock(&dd->lock);
        ...
        lock(&dd->zone_lock);       dd_finish_request()
        rq = find request           lock(&dd->zone_lock);
        unlock(&dd->zone_lock);
                                    zone write unlock
                                    unlock(&dd->zone_lock);
                                    ...
                                    __blk_mq_free_request
                                          check restart flag (not set)
                                          -> queue not run
        ...
        if (!rq && have writes)
            blk_mq_sched_mark_restart_hctx()
        unlock(&dd->lock)

    Since the dispatch context finishes after the write request completion
    handling, marking the queue as needing a restart is not seen from
```

```
    __blk_mq_free_request() and blk_mq_sched_restart() not executed leading
    to the dispatch stall under 100% write workloads.

    Fix this by moving the call to blk_mq_sched_mark_restart_hctx() from
    dd_dispatch_request() into dd_finish_request() under the zone lock to
    ensure full mutual exclusion between write request dispatch selection
    and zone unlock on write request completion.

    Fixes: 7211aef86f79 ("block: mq-deadline: Fix write completion handling")
    Cc: stable@vger.kernel.org
    Reported-by: Hans Holmberg <Hans.Holmberg@wdc.com>
    Reviewed-by: Hans Holmberg <hans.holmberg@wdc.com>
    Reviewed-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Damien Le Moal <damien.lemoal@wdc.com>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e715603042ec6698b737e0633b955d987205c4b4
Author: Stefan Assmann <sassmann@kpanic.de>
Date:   Wed Aug 21 16:09:29 2019 +0200

    i40e: check __I40E_VF_DISABLE bit in i40e_sync_filters_subtask

    commit a7542b87607560d0b89e7ff81d870bd6ff8835cb upstream.

    While testing VF spawn/destroy the following panic occurred.

    BUG: unable to handle kernel NULL pointer dereference at 0000000000000029
    [...]
    Workqueue: i40e i40e_service_task [i40e]
    RIP: 0010:i40e_sync_vsi_filters+0x6fd/0xc60 [i40e]
    [...]
    Call Trace:
     ? __switch_to_asm+0x35/0x70
     ? __switch_to_asm+0x41/0x70
     ? __switch_to_asm+0x35/0x70
     ? _cond_resched+0x15/0x30
     i40e_sync_filters_subtask+0x56/0x70 [i40e]
     i40e_service_task+0x382/0x11b0 [i40e]
     ? __switch_to_asm+0x41/0x70
     ? __switch_to_asm+0x41/0x70
     process_one_work+0x1a7/0x3b0
     worker_thread+0x30/0x390
     ? create_worker+0x1a0/0x1a0
     kthread+0x112/0x130
     ? kthread_bind+0x30/0x30
     ret_from_fork+0x35/0x40

    Investigation revealed a race where pf->vf[vsi->vf_id].trusted may get
    accessed by the watchdog via i40e_sync_filters_subtask() although
    i40e_free_vfs() already free'd pf->vf.
    To avoid this the call to i40e_sync_vsi_filters() in
    i40e_sync_filters_subtask() needs to be guarded by __I40E_VF_DISABLE,
    which is also used by i40e_free_vfs().

    Note: put the __I40E_VF_DISABLE check after the
    __I40E_MACVLAN_SYNC_PENDING check as the latter is more likely to
    trigger.

    CC: stable@vger.kernel.org
    Signed-off-by: Stefan Assmann <sassmann@kpanic.de>
    Tested-by: Andrew Bowers <andrewx.bowers@intel.com>
    Signed-off-by: Jeff Kirsher <jeffrey.t.kirsher@intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit 0a9f7e7fc5a2444c7fb4a42794a64476ba8c5f7e
Author: Rakesh Pillai <pillair@codeaurora.org>
Date:   Fri Mar 8 16:56:06 2019 +0530

    ath10k: fix channel info parsing for non tlv target

    commit 6be6c04bcc2e8770b8637632789ff15765124894 upstream.

    The tlv targets such as WCN3990 send more data in the chan info event, which is
    not sent by the non tlv targets. There is a minimum size check in the wmi event
    for non-tlv targets and hence we cannot update the common channel info
    structure as it was done in commit 13104929d2ec ("ath10k: fill the channel
    survey results for WCN3990 correctly"). This broke channel survey results on
    10.x firmware versions.

    If the common channel info structure is updated, the size check for chan info
    event for non-tlv targets will fail and return -EPROTO and we see the below
    error messages

       ath10k_pci 0000:01:00.0: failed to parse chan info event: -71

    Add tlv specific channel info structure and restore the original size of the
    common channel info structure to mitigate this issue.

    Tested HW: WCN3990
               QCA9887
    Tested FW: WLAN.HL.3.1-00784-QCAHLSWMTPLZ-1
               10.2.4-1.0-00037

    Fixes: 13104929d2ec ("ath10k: fill the channel survey results for WCN3990 correctly")
    Cc: stable@vger.kernel.org # 5.0
    Signed-off-by: Rakesh Pillai <pillair@codeaurora.org>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ee849d9c0691bf9eb41371715629d746d399dc14
Author: Jian-Hong Pan <jian-hong@endlessm.com>
Date:   Thu Jul 11 13:24:27 2019 +0800

    rtw88: pci: Use DMA sync instead of remapping in RX ISR

    commit 29b68a920f6abb7b5ba21ab4b779f62d536bac9b upstream.

    Since each skb in RX ring is reused instead of new allocation, we can
    treat the DMA in a more efficient way by DMA synchronization.

    Signed-off-by: Jian-Hong Pan <jian-hong@endlessm.com>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit d9f9e67330ba8c4724b3876a0f10d4cf7c922a54
Author: Jian-Hong Pan <jian-hong@endlessm.com>
Date:   Thu Jul 11 13:24:26 2019 +0800

    rtw88: pci: Rearrange the memory usage for skb in RX ISR

    commit ee6db78f5db9bfe426c57a1ec9713827ebccd2d4 upstream.

    Testing with RTL8822BE hardware, when available memory is low, we
    frequently see a kernel panic and system freeze.

    First, rtw_pci_rx_isr encounters a memory allocation failure (trimmed):

    rx routine starvation
```

```
    WARNING: CPU: 7 PID: 9871 at drivers/net/wireless/realtek/rtw88/pci.c:822
rtw_pci_rx_isr.constprop.25+0x35a/0x370 [rtwpci]
    [ 2356.580313] RIP: 0010:rtw_pci_rx_isr.constprop.25+0x35a/0x370 [rtwpci]

    Then we see a variety of different error conditions and kernel panics,
    such as this one (trimmed):

    rtw_pci 0000:02:00.0: pci bus timeout, check dma status
    skbuff: skb_over_panic: text:00000000091b6e66 len:415 put:415 head:00000000d2880c6f
data:000000007a02b1ea tail:0x1df end:0xc0 dev:<NULL>
    ------------[ cut here ]------------
    kernel BUG at net/core/skbuff.c:105!
    invalid opcode: 0000 [#1] SMP NOPTI
    RIP: 0010:skb_panic+0x43/0x45

    When skb allocation fails and the "rx routine starvation" is hit, the
    function returns immediately without updating the RX ring. At this
    point, the RX ring may continue referencing an old skb which was already
    handed off to ieee80211_rx_irqsafe(). When it comes to be used again,
    bad things happen.

    This patch allocates a new, data-sized skb first in RX ISR. After
    copying the data in, we pass it to the upper layers. However, if skb
    allocation fails, we effectively drop the frame. In both cases, the
    original, full size ring skb is reused.

    In addition, to fixing the kernel crash, the RX routine should now
    generally behave better under low memory conditions.

    Buglink: https://bugzilla.kernel.org/show_bug.cgi?id=204053
    Signed-off-by: Jian-Hong Pan <jian-hong@endlessm.com>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit d110bc8b3b4a3e318236c3b6341f113cc1e5b664
Author: Roberto Sassu <roberto.sassu@huawei.com>
Date:   Fri Sep 13 20:51:36 2019 +0200

    KEYS: trusted: correctly initialize digests and fix locking issue

    commit 9f75c82246313d4c2a6bc77e947b45655b3b5ad5 upstream.

    Commit 0b6cf6b97b7e ("tpm: pass an array of tpm_extend_digest structures to
    tpm_pcr_extend()") modifies tpm_pcr_extend() to accept a digest for each
    PCR bank. After modification, tpm_pcr_extend() expects that digests are
    passed in the same order as the algorithms set in chip->allocated_banks.

    This patch fixes two issues introduced in the last iterations of the patch
    set: missing initialization of the TPM algorithm ID in the tpm_digest
    structures passed to tpm_pcr_extend() by the trusted key module, and
    unreleased locks in the TPM driver due to returning from tpm_pcr_extend()
    without calling tpm_put_ops().

    Cc: stable@vger.kernel.org
    Fixes: 0b6cf6b97b7e ("tpm: pass an array of tpm_extend_digest structures to
tpm_pcr_extend()")
    Signed-off-by: Roberto Sassu <roberto.sassu@huawei.com>
    Suggested-by: Jarkko Sakkinen <jarkko.sakkinen@linux.intel.com>
    Reviewed-by: Jerry Snitselaar <jsnitsel@redhat.com>
    Reviewed-by: Jarkko Sakkinen <jarkko.sakkinen@linux.intel.com>
    Signed-off-by: Jarkko Sakkinen <jarkko.sakkinen@linux.intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 053b03efbdba3062fdf26baf26a49864b0766bc7
```

```
Author: Felix Fietkau <nbd@nbd.name>
Date:   Mon Jul 1 13:15:07 2019 +0200

    mt76: round up length on mt76_wr_copy

    commit 850e8f6fbd5d0003b0f1119d19a01c6fef1644e2 upstream.

    When beacon length is not a multiple of 4, the beacon could be sent with
    the last 1-3 bytes corrupted. The skb data is guaranteed to have enough
    room for reading beyond the end, because it is always followed by
    skb_shared_info, so rounding up is safe.
    All other callers of mt76_wr_copy have multiple-of-4 length already.

    Cc: stable@vger.kernel.org
    Signed-off-by: Felix Fietkau <nbd@nbd.name>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ddff6fc437fd75235108afc70ce2d35bb4bae79a
Author: Dave Rodgman <dave.rodgman@arm.com>
Date:   Wed Sep 25 16:48:24 2019 -0700

    lib/lzo/lzo1x_compress.c: fix alignment bug in lzo-rle

    commit 09b35b4192f6682dff96a093ab1930998cdb73b4 upstream.

    Fix an unaligned access which breaks on platforms where this is not
    permitted (e.g., Sparc).

    Link: http://lkml.kernel.org/r/20190912145502.35229-1-dave.rodgman@arm.com
    Signed-off-by: Dave Rodgman <dave.rodgman@arm.com>
    Cc: Dave Rodgman <dave.rodgman@arm.com>
    Cc: Markus F.X.J. Oberhumer <markus@oberhumer.com>
    Cc: Minchan Kim <minchan@kernel.org>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 0ceea3455e1ab8edb58d23a7f19e3be6b04ffafd
Author: Michal Hocko <mhocko@suse.com>
Date:   Wed Sep 25 16:45:53 2019 -0700

    memcg, kmem: do not fail __GFP_NOFAIL charges

    commit e55d9d9bfb69405bd7615c0f8d229d8fafb3e9b8 upstream.

    Thomas has noticed the following NULL ptr dereference when using cgroup
    v1 kmem limit:
    BUG: unable to handle kernel NULL pointer dereference at 0000000000000008
    PGD 0
    P4D 0
    Oops: 0000 [#1] PREEMPT SMP PTI
    CPU: 3 PID: 16923 Comm: gtk-update-icon Not tainted 4.19.51 #42
    Hardware name: Gigabyte Technology Co., Ltd. Z97X-Gaming G1/Z97X-Gaming G1, BIOS F9
07/31/2015
    RIP: 0010:create_empty_buffers+0x24/0x100
    Code: cd 0f 1f 44 00 00 0f 1f 44 00 00 41 54 49 89 d4 ba 01 00 00 00 55 53 48 89 fb e8
97 fe ff ff 48 89 c5 48 89 c2 eb 03 48 89 ca <48> 8b 4a 08 4c 09 22 48 85 c9 75 f1 48 89 6a
08 48 8b 43 18 48 8d
    RSP: 0018:ffff927ac1b37bf8 EFLAGS: 00010286
    RAX: 0000000000000000 RBX: fffff2d4429fd740 RCX: 0000000100097149
    RDX: 0000000000000000 RSI: 0000000000000082 RDI: ffff9075a99fbe00
    RBP: 0000000000000000 R08: fffff2d440949cc8 R09: 00000000000960c0
    R10: 0000000000000002 R11: 0000000000000000 R12: 0000000000000000
    R13: ffff907601f18360 R14: 0000000000002000 R15: 0000000000001000
```

```
FS:  00007fb55b288bc0(0000) GS:ffff90761f8c0000(0000) knlGS:0000000000000000
CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
CR2: 0000000000000008 CR3: 000000007aebc002 CR4: 00000000001606e0
Call Trace:
 create_page_buffers+0x4d/0x60
 __block_write_begin_int+0x8e/0x5a0
 ? ext4_inode_attach_jinode.part.82+0xb0/0xb0
 ? jbd2__journal_start+0xd7/0x1f0
 ext4_da_write_begin+0x112/0x3d0
 generic_perform_write+0xf1/0x1b0
 ? file_update_time+0x70/0x140
 __generic_file_write_iter+0x141/0x1a0
 ext4_file_write_iter+0xef/0x3b0
 __vfs_write+0x17e/0x1e0
 vfs_write+0xa5/0x1a0
 ksys_write+0x57/0xd0
 do_syscall_64+0x55/0x160
 entry_SYSCALL_64_after_hwframe+0x44/0xa9
```

Tetsuo then noticed that this is because the __memcg_kmem_charge_memcg fails __GFP_NOFAIL charge when the kmem limit is reached. This is a wrong behavior because nofail allocations are not allowed to fail. Normal charge path simply forces the charge even if that means to cross the limit. Kmem accounting should be doing the same.

Link: http://lkml.kernel.org/r/20190906125608.32129-1-mhocko@kernel.org
Signed-off-by: Michal Hocko <mhocko@suse.com>
Reported-by: Thomas Lindroth <thomas.lindroth@gmail.com>
Debugged-by: Tetsuo Handa <penguin-kernel@i-love.sakura.ne.jp>
Cc: Johannes Weiner <hannes@cmpxchg.org>
Cc: Vladimir Davydov <vdavydov.dev@gmail.com>
Cc: Andrey Ryabinin <aryabinin@virtuozzo.com>
Cc: Thomas Lindroth <thomas.lindroth@gmail.com>
Cc: Shakeel Butt <shakeelb@google.com>
Cc: <stable@vger.kernel.org>
Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1a11f8a707a92ba8355aaabbe8f167dfdc3a2abd
Author: Tetsuo Handa <penguin-kernel@i-love.sakura.ne.jp>
Date: Mon Sep 23 15:37:08 2019 -0700

    memcg, oom: don't require __GFP_FS when invoking memcg OOM killer

    commit f9c645621a28e37813a1de96d9cbd89cde94a1e4 upstream.

    Masoud Sharbiani noticed that commit 29ef680ae7c21110 ("memcg, oom: move
    out_of_memory back to the charge path") broke memcg OOM called from
    __xfs_filemap_fault() path. It turned out that try_charge() is retrying
    forever without making forward progress because mem_cgroup_oom(GFP_NOFS)
    cannot invoke the OOM killer due to commit 3da88fb3bacfaa33 ("mm, oom:
    move GFP_NOFS check to out_of_memory").

    Allowing forced charge due to being unable to invoke memcg OOM killer will
    lead to global OOM situation. Also, just returning -ENOMEM will be risky
    because OOM path is lost and some paths (e.g. get_user_pages()) will leak
    -ENOMEM. Therefore, invoking memcg OOM killer (despite GFP_NOFS) will be
    the only choice we can choose for now.

    Until 29ef680ae7c21110, we were able to invoke memcg OOM killer when
    GFP_KERNEL reclaim failed [1]. But since 29ef680ae7c21110, we need to
    invoke memcg OOM killer when GFP_NOFS reclaim failed [2]. Although in the
    past we did invoke memcg OOM killer for GFP_NOFS [3], we might get
    pre-mature memcg OOM reports due to this patch.

```
    [1]

     leaker invoked oom-killer: gfp_mask=0x6200ca(GFP_HIGHUSER_MOVABLE), nodemask=(null),
 order=0, oom_score_adj=0
     CPU: 0 PID: 2746 Comm: leaker Not tainted 4.18.0+ #19
     Hardware name: VMware, Inc. VMware Virtual Platform/440BX Desktop Reference Platform,
 BIOS 6.00 04/13/2018
     Call Trace:
      dump_stack+0x63/0x88
      dump_header+0x67/0x27a
      ? mem_cgroup_scan_tasks+0x91/0xf0
      oom_kill_process+0x210/0x410
      out_of_memory+0x10a/0x2c0
      mem_cgroup_out_of_memory+0x46/0x80
      mem_cgroup_oom_synchronize+0x2e4/0x310
      ? high_work_func+0x20/0x20
      pagefault_out_of_memory+0x31/0x76
      mm_fault_error+0x55/0x115
      ? handle_mm_fault+0xfd/0x220
      __do_page_fault+0x433/0x4e0
      do_page_fault+0x22/0x30
      ? page_fault+0x8/0x30
      page_fault+0x1e/0x30
     RIP: 0033:0x4009f0
     Code: 03 00 00 00 e8 71 fd ff ff 48 83 f8 ff 49 89 c6 74 74 48 89 c6 bf c0 0c 40 00 31
 c0 e8 69 fd ff ff 45 85 ff 7e 21 31 c9 66 90 <41> 0f be 14 0e 01 d3 f7 c1 ff 0f 00 00 75 05
 41 c6 04 0e 2a 48 83
     RSP: 002b:00007ffe29ae96f0 EFLAGS: 00010206
     RAX: 000000000000001b RBX: 0000000000000000 RCX: 0000000001ce1000
     RDX: 0000000000000000 RSI: 000000007fffffe5 RDI: 0000000000000000
     RBP: 000000000000000c R08: 0000000000000000 R09: 00007f94be09220d
     R10: 0000000000000002 R11: 0000000000000246 R12: 00000000000186a0
     R13: 0000000000000003 R14: 00007f949d845000 R15: 0000000002800000
     Task in /leaker killed as a result of limit of /leaker
     memory: usage 524288kB, limit 524288kB, failcnt 158965
     memory+swap: usage 0kB, limit 9007199254740988kB, failcnt 0
     kmem: usage 2016kB, limit 9007199254740988kB, failcnt 0
     Memory cgroup stats for /leaker: cache:844KB rss:521136KB rss_huge:0KB shmem:0KB
 mapped_file:0KB dirty:132KB writeback:0KB inactive_anon:0KB active_anon:521224KB
 inactive_file:1012KB active_file:8KB unevictable:0KB
     Memory cgroup out of memory: Kill process 2746 (leaker) score 998 or sacrifice child
     Killed process 2746 (leaker) total-vm:536704kB, anon-rss:521176kB, file-rss:1208kB,
 shmem-rss:0kB
     oom_reaper: reaped process 2746 (leaker), now anon-rss:0kB, file-rss:0kB, shmem-rss:0kB

    [2]

     leaker invoked oom-killer: gfp_mask=0x600040(GFP_NOFS), nodemask=(null), order=0,
 oom_score_adj=0
     CPU: 1 PID: 2746 Comm: leaker Not tainted 4.18.0+ #20
     Hardware name: VMware, Inc. VMware Virtual Platform/440BX Desktop Reference Platform,
 BIOS 6.00 04/13/2018
     Call Trace:
      dump_stack+0x63/0x88
      dump_header+0x67/0x27a
      ? mem_cgroup_scan_tasks+0x91/0xf0
      oom_kill_process+0x210/0x410
      out_of_memory+0x109/0x2d0
      mem_cgroup_out_of_memory+0x46/0x80
      try_charge+0x58d/0x650
      ? __radix_tree_replace+0x81/0x100
      mem_cgroup_try_charge+0x7a/0x100
      __add_to_page_cache_locked+0x92/0x180
      add_to_page_cache_lru+0x4d/0xf0
```

```
       iomap_readpages_actor+0xde/0x1b0
       ? iomap_zero_range_actor+0x1d0/0x1d0
       iomap_apply+0xaf/0x130
       iomap_readpages+0x9f/0x150
       ? iomap_zero_range_actor+0x1d0/0x1d0
       xfs_vm_readpages+0x18/0x20 [xfs]
       read_pages+0x60/0x140
       __do_page_cache_readahead+0x193/0x1b0
       ondemand_readahead+0x16d/0x2c0
       page_cache_async_readahead+0x9a/0xd0
       filemap_fault+0x403/0x620
       ? alloc_set_pte+0x12c/0x540
       ? _cond_resched+0x14/0x30
       __xfs_filemap_fault+0x66/0x180 [xfs]
       xfs_filemap_fault+0x27/0x30 [xfs]
       __do_fault+0x19/0x40
       __handle_mm_fault+0x8e8/0xb60
       handle_mm_fault+0xfd/0x220
       __do_page_fault+0x238/0x4e0
       do_page_fault+0x22/0x30
       ? page_fault+0x8/0x30
       page_fault+0x1e/0x30
     RIP: 0033:0x4009f0
     Code: 03 00 00 00 e8 71 fd ff ff 48 83 f8 ff 49 89 c6 74 74 48 89 c6 bf c0 0c 40 00 31
 c0 e8 69 fd ff ff 45 85 ff 7e 21 31 c9 66 90 <41> 0f be 14 0e 01 d3 f7 c1 ff 0f 00 00 75 05
 41 c6 04 0e 2a 48 83
     RSP: 002b:00007ffda45c9290 EFLAGS: 00010206
     RAX: 000000000000001b RBX: 0000000000000000 RCX: 0000000001a1e000
     RDX: 0000000000000000 RSI: 000000007fffffe5 RDI: 0000000000000000
     RBP: 000000000000000c R08: 0000000000000000 R09: 00007f6d061ff20d
     R10: 0000000000000002 R11: 0000000000000246 R12: 00000000000186a0
     R13: 0000000000000003 R14: 00007f6ce59b2000 R15: 0000000002800000
     Task in /leaker killed as a result of limit of /leaker
     memory: usage 524288kB, limit 524288kB, failcnt 7221
     memory+swap: usage 0kB, limit 9007199254740988kB, failcnt 0
     kmem: usage 1944kB, limit 9007199254740988kB, failcnt 0
     Memory cgroup stats for /leaker: cache:3632KB rss:518232KB rss_huge:0KB shmem:0KB
 mapped_file:0KB dirty:0KB writeback:0KB inactive_anon:0KB active_anon:518408KB
 inactive_file:3908KB active_file:12KB unevictable:0KB
     Memory cgroup out of memory: Kill process 2746 (leaker) score 992 or sacrifice child
     Killed process 2746 (leaker) total-vm:536704kB, anon-rss:518264kB, file-rss:1188kB,
 shmem-rss:0kB
     oom_reaper: reaped process 2746 (leaker), now anon-rss:0kB, file-rss:0kB, shmem-rss:0kB

    [3]

     leaker invoked oom-killer: gfp_mask=0x50, order=0, oom_score_adj=0
     leaker cpuset=/ mems_allowed=0
     CPU: 1 PID: 3206 Comm: leaker Not tainted 3.10.0-957.27.2.el7.x86_64 #1
     Hardware name: VMware, Inc. VMware Virtual Platform/440BX Desktop Reference Platform,
 BIOS 6.00 04/13/2018
     Call Trace:
      [<ffffffffaf364147>] dump_stack+0x19/0x1b
      [<ffffffffaf35eb6a>] dump_header+0x90/0x229
      [<ffffffffaedbb456>] ? find_lock_task_mm+0x56/0xc0
      [<ffffffffaee32a38>] ? try_get_mem_cgroup_from_mm+0x28/0x60
      [<ffffffffaedbb904>] oom_kill_process+0x254/0x3d0
      [<ffffffffaee36c36>] mem_cgroup_oom_synchronize+0x546/0x570
      [<ffffffffaee360b0>] ? mem_cgroup_charge_common+0xc0/0xc0
      [<ffffffffaedbc194>] pagefault_out_of_memory+0x14/0x90
      [<ffffffffaf35d072>] mm_fault_error+0x6a/0x157
      [<ffffffffaf3717c8>] __do_page_fault+0x3c8/0x4f0
      [<ffffffffaf371925>] do_page_fault+0x35/0x90
      [<ffffffffaf36d768>] page_fault+0x28/0x30
     Task in /leaker killed as a result of limit of /leaker
```

```
      memory: usage 524288kB, limit 524288kB, failcnt 20628
      memory+swap: usage 524288kB, limit 9007199254740988kB, failcnt 0
      kmem: usage 0kB, limit 9007199254740988kB, failcnt 0
      Memory cgroup stats for /leaker: cache:840KB rss:523448KB rss_huge:0KB mapped_file:0KB
 swap:0KB inactive_anon:0KB active_anon:523448KB inactive_file:464KB active_file:376KB
 unevictable:0KB
      Memory cgroup out of memory: Kill process 3206 (leaker) score 970 or sacrifice child
      Killed process 3206 (leaker) total-vm:536692kB, anon-rss:523304kB, file-rss:412kB,
 shmem-rss:0kB

    Bisected by Masoud Sharbiani.

    Link: http://lkml.kernel.org/r/cbe54ed1-b6ba-a056-8899-2dc42526371d@i-love.sakura.ne.jp
    Fixes: 3da88fb3bacfaa33 ("mm, oom: move GFP_NOFS check to out_of_memory") [necessary
 after 29ef680ae7c21110]
    Signed-off-by: Tetsuo Handa <penguin-kernel@I-love.SAKURA.ne.jp>
    Reported-by: Masoud Sharbiani <msharbiani@apple.com>
    Tested-by: Masoud Sharbiani <msharbiani@apple.com>
    Acked-by: Michal Hocko <mhocko@suse.com>
    Cc: David Rientjes <rientjes@google.com>
    Cc: <stable@vger.kernel.org>    [4.19+]
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

 commit 3d92c2909f1977a3ac1ede01e5729f30ae755ac0
 Author: Yafang Shao <laoar.shao@gmail.com>
 Date:   Mon Sep 23 15:36:54 2019 -0700

    mm/compaction.c: clear total_{migrate,free}_scanned before scanning a new zone

    commit a94b525241c0fff3598809131d7cfcfe1d572d8c upstream.

    total_{migrate,free}_scanned will be added to COMPACTMIGRATE_SCANNED and
    COMPACTFREE_SCANNED in compact_zone().  We should clear them before
    scanning a new zone.  In the proc triggered compaction, we forgot clearing
    them.

    [laoar.shao@gmail.com: introduce a helper compact_zone_counters_init()]
      Link: http://lkml.kernel.org/r/1563869295-25748-1-git-send-email-laoar.shao@gmail.com
    [akpm@linux-foundation.org: expand compact_zone_counters_init() into its single
 callsite, per mhocko]
    [vbabka@suse.cz: squash compact_zone() list_head init as well]
      Link: http://lkml.kernel.org/r/1fb6f7da-f776-9e42-22f8-bbb79b030b98@suse.cz
    [akpm@linux-foundation.org: kcompactd_do_work(): avoid unnecessary initialization of
 cc.zone]
    Link: http://lkml.kernel.org/r/1563789275-9639-1-git-send-email-laoar.shao@gmail.com
    Fixes: 7f354a548d1c ("mm, compaction: add vmstats for kcompactd work")
    Signed-off-by: Yafang Shao <laoar.shao@gmail.com>
    Signed-off-by: Vlastimil Babka <vbabka@suse.cz>
    Reviewed-by: Vlastimil Babka <vbabka@suse.cz>
    Cc: David Rientjes <rientjes@google.com>
    Cc: Yafang Shao <shaoyafang@didiglobal.com>
    Cc: Mel Gorman <mgorman@techsingularity.net>
    Cc: Michal Hocko <mhocko@suse.com>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

 commit b8ce9ad765e2ca90394d2a6eaa1a3eff9066e7d0
 Author: Vitaly Wool <vitalywool@gmail.com>
 Date:   Mon Sep 23 15:36:51 2019 -0700

    z3fold: fix memory leak in kmem cache
```

```
    commit 63398413c00c7836ea87a1fa205c91d2199b25cf upstream.

    Currently there is a leak in init_z3fold_page() -- it allocates handles
    from kmem cache even for headless pages, but then they are never used and
    never freed, so eventually kmem cache may get exhausted.  This patch
    provides a fix for that.

    Link: http://lkml.kernel.org/r/20190917185352.44cf285d3ebd9e64548de5de@gmail.com
    Signed-off-by: Vitaly Wool <vitalywool@gmail.com>
    Reported-by: Markus Linnala <markus.linnala@gmail.com>
    Tested-by: Markus Linnala <markus.linnala@gmail.com>
    Cc: Dan Streetman <ddstreet@ieee.org>
    Cc: Henry Burns <henrywolfeburns@gmail.com>
    Cc: Shakeel Butt <shakeelb@google.com>
    Cc: Vlastimil Babka <vbabka@suse.cz>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit fe27842091cfc6c973058607f9c097f91d01773b
Author: Vitaly Wool <vitalywool@gmail.com>
Date:   Mon Sep 23 15:33:02 2019 -0700

    z3fold: fix retry mechanism in page reclaim

    commit 3f9d2b5766aea06042630ac60b7316fd0cebf06f upstream.

    z3fold_page_reclaim()'s retry mechanism is broken: on a second iteration
    it will have zhdr from the first one so that zhdr is no longer in line
    with struct page.  That leads to crashes when the system is stressed.

    Fix that by moving zhdr assignment up.

    While at it, protect against using already freed handles by using own
    local slots structure in z3fold_page_reclaim().

    Link: http://lkml.kernel.org/r/20190908162919.830388dc7404d1e2c80f4095@gmail.com
    Signed-off-by: Vitaly Wool <vitalywool@gmail.com>
    Reported-by: Markus Linnala <markus.linnala@gmail.com>
    Reported-by: Chris Murphy <bugzilla@colorremedies.com>
    Reported-by: Agustin Dall'Alba <agustin@dallalba.com.ar>
    Cc: "Maciej S. Szmigiero" <mail@maciej.szmigiero.name>
    Cc: Shakeel Butt <shakeelb@google.com>
    Cc: Henry Burns <henrywolfeburns@gmail.com>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e8a4472acb33a93373eaaf14969889a766aeb340
Author: Bob Peterson <rpeterso@redhat.com>
Date:   Thu Sep 12 13:54:27 2019 -0400

    gfs2: clear buf_in_tr when ending a transaction in sweep_bh_for_rgrps

    commit f0b444b349e33ae0d3dd93e25ca365482a5d17d4 upstream.

    In function sweep_bh_for_rgrps, which is a helper for punch_hole,
    it uses variable buf_in_tr to keep track of when it needs to commit
    pending block frees on a partial delete that overflows the
    transaction created for the delete. The problem is that the
    variable was initialized at the start of function sweep_bh_for_rgrps
    but it was never cleared, even when starting a new transaction.
```

```
    This patch reinitializes the variable when the transaction is
    ended, so the next transaction starts out with it cleared.

    Fixes: d552a2b9b33e ("GFS2: Non-recursive delete")
    Cc: stable@vger.kernel.org # v4.12+
    Signed-off-by: Bob Peterson <rpeterso@redhat.com>
    Signed-off-by: Andreas Gruenbacher <agruenba@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 596b280befccd657633c76d88e6ff4c2652d69e6
Author: Hans de Goede <hdegoede@redhat.com>
Date:   Sun Jul 21 15:19:18 2019 +0200

    efifb: BGRT: Improve efifb_bgrt_sanity_check

    commit 51677dfcc17f88ed754143df670ff064eae67f84 upstream.

    For various reasons, at least with x86 EFI firmwares, the xoffset and
    yoffset in the BGRT info are not always reliable.

    Extensive testing has shown that when the info is correct, the
    BGRT image is always exactly centered horizontally (the yoffset variable
    is more variable and not always predictable).

    This commit simplifies / improves the bgrt_sanity_check to simply
    check that the BGRT image is exactly centered horizontally and skips
    (re)drawing it when it is not.

    This fixes the BGRT image sometimes being drawn in the wrong place.

    Cc: stable@vger.kernel.org
    Fixes: 88fe4ceb2447 ("efifb: BGRT: Do not copy the boot graphics for non native
resolutions")
    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Cc: Peter Jones <pjones@redhat.com>,
    Signed-off-by: Bartlomiej Zolnierkiewicz <b.zolnierkie@samsung.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20190721131918.10115-1-
hdegoede@redhat.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3fbfa415a5dd8fc78076c1ead160a03a058cd9f6
Author: Mark Brown <broonie@kernel.org>
Date:   Wed Sep 4 13:42:50 2019 +0100

    regulator: Defer init completion for a while after late_initcall

    commit 55576cf1853798e86f620766e23b604c9224c19c upstream.

    The kernel has no way of knowing when we have finished instantiating
    drivers, between deferred probe and systems that build key drivers as
    modules we might be doing this long after userspace has booted. This has
    always been a bit of an issue with regulator_init_complete since it can
    power off hardware that's not had it's driver loaded which can result in
    user visible effects, the main case is powering off displays. Practically
    speaking it's not been an issue in real systems since most systems that
    use the regulator API are embedded and build in key drivers anyway but
    with Arm laptops coming on the market it's becoming more of an issue so
    let's do something about it.

    In the absence of any better idea just defer the powering off for 30s
    after late_initcall(), this is obviously a hack but it should mask the
    issue for now and it's no more arbitrary than late_initcall() itself.
    Ideally we'd have some heuristics to detect if we're on an affected
    system and tune or skip the delay appropriately, and there may be some
```

```
    need for a command line option to be added.

    Link: https://lore.kernel.org/r/20190904124250.25844-1-broonie@kernel.org
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Tested-by: Lee Jones <lee.jones@linaro.org>
    Cc: stable@vger.kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 0564ef45b2f67c017d074398434cde292ee16e6a
Author: Nadav Amit <namit@vmware.com>
Date:   Tue Aug 20 01:53:17 2019 -0700

    iommu/vt-d: Fix wrong analysis whether devices share the same bus

    commit 2c70010867f164d1b30e787e360e05d10cc40046 upstream.

    set_msi_sid_cb() is used to determine whether device aliases share the
    same bus, but it can provide false indications that aliases use the same
    bus when in fact they do not. The reason is that set_msi_sid_cb()
    assumes that pdev is fixed, while actually pci_for_each_dma_alias() can
    call fn() when pdev is set to a subordinate device.

    As a result, running an VM on ESX with VT-d emulation enabled can
    results in the log warning such as:

      DMAR: [INTR-REMAP] Request device [00:11.0] fault index 3b [fault reason 38] Blocked
an interrupt request due to source-id verification failure

    This seems to cause additional ata errors such as:
      ata3.00: qc timeout (cmd 0xa1)
      ata3.00: failed to IDENTIFY (I/O error, err_mask=0x4)

    These timeouts also cause boot to be much longer and other errors.

    Fix it by checking comparing the alias with the previous one instead.

    Fixes: 3f0c625c6ae71 ("iommu/vt-d: Allow interrupts from the entire bus for aliased
devices")
    Cc: stable@vger.kernel.org
    Cc: Logan Gunthorpe <logang@deltatee.com>
    Cc: David Woodhouse <dwmw2@infradead.org>
    Cc: Joerg Roedel <joro@8bytes.org>
    Cc: Jacob Pan <jacob.jun.pan@linux.intel.com>
    Signed-off-by: Nadav Amit <namit@vmware.com>
    Reviewed-by: Logan Gunthorpe <logang@deltatee.com>
    Signed-off-by: Joerg Roedel <jroedel@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b1a78a24172e3fb6ada01da392a08edb960e344d
Author: Thadeu Lima de Souza Cascardo <cascardo@canonical.com>
Date:   Tue Sep 3 14:18:02 2019 -0300

    alarmtimer: Use EOPNOTSUPP instead of ENOTSUPP

    commit f18ddc13af981ce3c7b7f26925f099e7c6929aba upstream.

    ENOTSUPP is not supposed to be returned to userspace. This was found on an
    OpenPower machine, where the RTC does not support set_alarm.

    On that system, a clock_nanosleep(CLOCK_REALTIME_ALARM, ...) results in
    "524 Unknown error 524"

    Replace it with EOPNOTSUPP which results in the expected "95 Operation not
    supported" error.
```

```
    Fixes: 1c6b39ad3f01 (alarmtimers: Return -ENOTSUPP if no RTC device is present)
    Signed-off-by: Thadeu Lima de Souza Cascardo <cascardo@canonical.com>
    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Cc: stable@vger.kernel.org
    Link: https://lkml.kernel.org/r/20190903171802.28314-1-cascardo@canonical.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 4cc03cdf28fbdc80783cd28411398ea16a78760f
Author: Will Deacon <will@kernel.org>
Date:   Wed Aug 21 14:17:00 2019 +0100

    iommu/arm-smmu-v3: Disable detection of ATS and PRI

    commit b5e86196b83fd68e065a7c811ab8925fb0dc3893 upstream.

    Detecting the ATS capability of the SMMU at probe time introduces a
    spinlock into the ->unmap() fast path, even when ATS is not actually
    in use. Furthermore, the ATC invalidation that exists is broken, as it
    occurs before invalidation of the main SMMU TLB which leaves a window
    where the ATC can be repopulated with stale entries.

    Given that ATS is both a new feature and a specialist sport, disable it
    for now whilst we fix it properly in subsequent patches. Since PRI
    requires ATS, disable that too.

    Cc: <stable@vger.kernel.org>
    Fixes: 9ce27afc0830 ("iommu/arm-smmu-v3: Add support for PCI ATS")
    Acked-by: Robin Murphy <robin.murphy@arm.com>
    Signed-off-by: Will Deacon <will@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 922b0e17473bb5577331cd4b42891bbb54dc8f60
Author: Shawn Lin <shawn.lin@rock-chips.com>
Date:   Fri Aug 30 08:26:47 2019 +0800

    arm64: dts: rockchip: limit clock rate of MMC controllers for RK3328

    commit 03e61929c0d227ed3e1c322fc3804216ea298b7e upstream.

    150MHz is a fundamental limitation of RK3328 Soc, w/o this limitation,
    eMMC, for instance, will run into 200MHz clock rate in HS200 mode, which
    makes the RK3328 boards not always boot properly. By adding it in
    rk3328.dtsi would also obviate the worry of missing it when adding new
    boards.

    Fixes: 52e02d377a72 ("arm64: dts: rockchip: add core dtsi file for RK3328 SoCs")
    Cc: stable@vger.kernel.org
    Cc: Robin Murphy <robin.murphy@arm.com>
    Cc: Liang Chen <cl@rock-chips.com>
    Signed-off-by: Shawn Lin <shawn.lin@rock-chips.com>
    Signed-off-by: Heiko Stuebner <heiko@sntech.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit d6984c21d8154ca00cdcf4bb91476bf7441d0215
Author: Will Deacon <will@kernel.org>
Date:   Thu Aug 22 15:03:45 2019 +0100

    arm64: tlb: Ensure we execute an ISB following walk cache invalidation

    commit 51696d346c49c6cf4f29e9b20d6e15832a2e3408 upstream.

    05f2d2f83b5a ("arm64: tlbflush: Introduce __flush_tlb_kernel_pgtable")
    added a new TLB invalidation helper which is used when freeing
    intermediate levels of page table used for kernel mappings, but is
    missing the required ISB instruction after completion of the TLBI
```

```
    instruction.

    Add the missing barrier.

    Cc: <stable@vger.kernel.org>
    Fixes: 05f2d2f83b5a ("arm64: tlbflush: Introduce __flush_tlb_kernel_pgtable")
    Reviewed-by: Mark Rutland <mark.rutland@arm.com>
    Signed-off-by: Will Deacon <will@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 2dcea73cea85bd395fcbc2e491a2c91c90a605a6
Author: Luis Araneda <luaraneda@gmail.com>
Date:   Thu Aug 8 08:52:43 2019 -0400

    ARM: zynq: Use memcpy_toio instead of memcpy on smp bring-up

    commit b7005d4ef4f3aa2dc24019ffba03a322557ac43d upstream.

    This fixes a kernel panic on memcpy when
    FORTIFY_SOURCE is enabled.

    The initial smp implementation on commit aa7eb2bb4e4a
    ("arm: zynq: Add smp support")
    used memcpy, which worked fine until commit ee333554fed5
    ("ARM: 8749/1: Kconfig: Add ARCH_HAS_FORTIFY_SOURCE")
    enabled overflow checks at runtime, producing a read
    overflow panic.

    The computed size of memcpy args are:
    - p_size (dst): 4294967295 = (size_t) -1
    - q_size (src): 1
    - size (len): 8

    Additionally, the memory is marked as __iomem, so one of
    the memcpy_* functions should be used for read/write.

    Fixes: aa7eb2bb4e4a ("arm: zynq: Add smp support")
    Signed-off-by: Luis Araneda <luaraneda@gmail.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Michal Simek <michal.simek@xilinx.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c2d30ffe4cabd6a4bb084304262df6fa282ac069
Author: Lihua Yao <ylhuajnu@outlook.com>
Date:   Sat Sep 7 03:30:01 2019 +0000

    ARM: samsung: Fix system restart on S3C6410

    commit 16986074035cc0205472882a00d404ed9d213313 upstream.

    S3C6410 system restart is triggered by watchdog reset.

    Cc: <stable@vger.kernel.org>
    Fixes: 9f55342cc2de ("ARM: dts: s3c64xx: Fix infinite interrupt in soft mode")
    Signed-off-by: Lihua Yao <ylhuajnu@outlook.com>
    Signed-off-by: Krzysztof Kozlowski <krzk@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ea82e3cf540a1c28721a3a880e1f1f7f8adb330b
Author: Gao Xiang <gaoxiang25@huawei.com>
Date:   Mon Aug 19 18:34:22 2019 +0800

    staging: erofs: cannot set EROFS_V_Z_INITED_BIT if fill_inode_lazy fails

    commit 3407a4198faf01c9d7596c45b8606834b8dfc2b7 upstream.
```

```
    As reported by erofs-utils fuzzer, unsupported compressed
    clustersize will make fill_inode_lazy fail, for such case
    we cannot set EROFS_V_Z_INITED_BIT since we need return
    failure for each z_erofs_map_blocks_iter().

    Fixes: 152a333a5895 ("staging: erofs: add compacted compression indexes support")
    Cc: <stable@vger.kernel.org> # 5.3+
    Signed-off-by: Gao Xiang <gaoxiang25@huawei.com>
    Reviewed-by: Chao Yu <yuchao0@huawei.com>
    Link: https://lore.kernel.org/r/20190819103426.87579-3-gaoxiang25@huawei.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 4115a85b27909bdf5d6f8be07602ca9be5ee1f92
Author: Amadeusz Sławiński <amadeuszx.slawinski@intel.com>
Date:   Tue Aug 27 16:17:08 2019 +0200

    ASoC: Intel: Fix use of potentially uninitialized variable

    commit 810f3b860850148788fc1ed8a6f5f807199fed65 upstream.

    If ipc->ops.reply_msg_match is NULL, we may end up using uninitialized
    mask value.

    reported by smatch:
    sound/soc/intel/common/sst-ipc.c:266 sst_ipc_reply_find_msg() error: uninitialized
symbol 'mask'.

    Signed-off-by: Amadeusz Sławiński <amadeuszx.slawinski@intel.com>
    Link: https://lore.kernel.org/r/20190827141712.21015-3-
amadeuszx.slawinski@linux.intel.com
    Reviewed-by: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com>
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Cc: stable@vger.kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a8e709239fe05cde40d6436b50e7bbd90ee6e26b
Author: Amadeusz Sławiński <amadeuszx.slawinski@intel.com>
Date:   Tue Aug 27 16:17:07 2019 +0200

    ASoC: Intel: Skylake: Use correct function to access iomem space

    commit 17d29ff98fd4b70e9ccdac5e95e18a087e2737ef upstream.

    For copying from __iomem, we should use __ioread32_copy.

    reported by sparse:
    sound/soc/intel/skylake/skl-debug.c:437:34: warning: incorrect type in argument 1
(different address spaces)
    sound/soc/intel/skylake/skl-debug.c:437:34:    expected void [noderef] <asn:2> *to
    sound/soc/intel/skylake/skl-debug.c:437:34:    got unsigned char *
    sound/soc/intel/skylake/skl-debug.c:437:51: warning: incorrect type in argument 2
(different address spaces)
    sound/soc/intel/skylake/skl-debug.c:437:51:    expected void const *from
    sound/soc/intel/skylake/skl-debug.c:437:51:    got void [noderef] <asn:2> *[assigned]
fw_reg_addr

    Signed-off-by: Amadeusz Sławiński <amadeuszx.slawinski@intel.com>
    Link: https://lore.kernel.org/r/20190827141712.21015-2-
amadeuszx.slawinski@linux.intel.com
    Reviewed-by: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com>
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Cc: stable@vger.kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit c3e6117ca60b3f4a33f72fa652d11e718e93dfd2
Author: Amadeusz Sławiński <amadeuszx.slawinski@intel.com>
Date:   Tue Aug 27 16:17:12 2019 +0200

    ASoC: Intel: NHLT: Fix debug print format

    commit 855a06da37a773fd073d51023ac9d07988c87da8 upstream.

    oem_table_id is 8 chars long, so we need to limit it, otherwise it
    may print some unprintable characters into dmesg.

    Signed-off-by: Amadeusz Sławiński <amadeuszx.slawinski@intel.com>
    Link: https://lore.kernel.org/r/20190827141712.21015-7-
amadeuszx.slawinski@linux.intel.com
    Reviewed-by: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com>
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Cc: stable@vger.kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 7ef3cd9cbdccbe66246491a468f8e48b430286e9
Author: Kees Cook <keescook@chromium.org>
Date:   Thu Sep 26 10:15:25 2019 -0700

    binfmt_elf: Do not move brk for INTERP-less ET_EXEC

    commit 7be3cb019db1cbd5fd5ffe6d64a23fefa4b6f229 upstream.

    When brk was moved for binaries without an interpreter, it should have
    been limited to ET_DYN only. In other words, the special case was an
    ET_DYN that lacks an INTERP, not just an executable that lacks INTERP.
    The bug manifested for giant static executables, where the brk would end
    up in the middle of the text area on 32-bit architectures.

    Reported-and-tested-by: Richard Kojedzinszky <richard@kojedz.in>
    Fixes: bbdc6076d2e5 ("binfmt_elf: move brk out of mmap when doing direct loader exec")
    Cc: stable@vger.kernel.org
    Signed-off-by: Kees Cook <keescook@chromium.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 8d2c9ff537f9ceedae3506fdc0bf43116c61e8cb
Author: Vladimir Oltean <olteanv@gmail.com>
Date:   Fri Aug 23 00:24:50 2019 +0300

    spi: spi-fsl-dspi: Exit the ISR with IRQ_NONE when it's not ours

    commit d41f36a6464a85c06ad920703d878e4491d2c023 upstream.

    The DSPI interrupt can be shared between two controllers at least on the
    LX2160A. In that case, the driver for one controller might misbehave and
    consume the other's interrupt. Fix this by actually checking if any of
    the bits in the status register have been asserted.

    Fixes: 13aed2392741 ("spi: spi-fsl-dspi: use IRQF_SHARED mode to request IRQ")
    Signed-off-by: Vladimir Oltean <olteanv@gmail.com>
    Link: https://lore.kernel.org/r/20190822212450.21420-2-olteanv@gmail.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Cc: stable@vger.kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 4e4a1dcb543d40ae76ecab03a7972d0268c65674
Author: Alexander Sverdlin <alexander.sverdlin@gmail.com>
Date:   Sat Aug 31 20:04:02 2019 +0200

    spi: ep93xx: Repair SPI CS lookup tables
```

```
    commit 4fbc485324d2975c54201091dfad0a7dd4902324 upstream.

    The actual device name of the SPI controller being registered on EP93xx is
    "spi0" (as seen by gpiod_find_lookup_table()). This patch fixes all
    relevant lookup tables and the following failure (seen on EDB9302):

    ep93xx-spi ep93xx-spi.0: failed to register SPI master
    ep93xx-spi: probe of ep93xx-spi.0 failed with error -22

    Fixes: 1dfbf334f1236 ("spi: ep93xx: Convert to use CS GPIO descriptors")
    Cc: stable@vger.kernel.org
    Signed-off-by: Alexander Sverdlin <alexander.sverdlin@gmail.com>
    Reviewed-by: Linus Walleij <linus.walleij@linaro.org>
    Reviewed-by: Lukasz Majewski <lukma@denx.de>
    Link: https://lore.kernel.org/r/20190831180402.10008-1-alexander.sverdlin@gmail.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 50af2e37b3fbde15ee7e6de9745731b09c2e37ce
Author: Guillaume Tucker <guillaume.tucker@collabora.com>
Date:   Wed Jul 24 11:19:22 2019 -0400

    media: vivid: fix device init when no_error_inj=1 and fb disabled

    commit 79e85d1d2c16ba4907bb9d6a4381516b729ff341 upstream.

    Add an extra condition to add the video output control class when the
    device has some hdmi outputs defined.  This is required to then always
    be able to add the display present control, which is enabled when
    there are some hdmi outputs.

    This fixes the corner case where no_error_inj is enabled and the
    device has no frame buffer but some hdmi outputs, as otherwise the
    video output control class would be added anyway.  Without this fix,
    the sanity checks fail in v4l2_ctrl_new() as name is NULL.

    Fixes: c533435ffb91 ("media: vivid: add display present control")
    Cc: stable@vger.kernel.org # for 5.3
    Signed-off-by: Guillaume Tucker <guillaume.tucker@collabora.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 18fb9b18fb3d2305ac1bbd650817bfe6b7357a17
Author: Arnd Bergmann <arnd@arndb.de>
Date:   Wed Jun 19 10:24:17 2019 -0300

    media: don't drop front-end reference count for ->detach

    commit 14e3cdbb00a885eedc95c0cf8eda8fe28d26d6b4 upstream.

    A bugfix introduce a link failure in configurations without CONFIG_MODULES:

    In file included from drivers/media/usb/dvb-usb/pctv452e.c:20:0:
    drivers/media/usb/dvb-usb/pctv452e.c: In function 'pctv452e_frontend_attach':
    drivers/media/dvb-frontends/stb0899_drv.h:151:36: error: weak declaration of
'stb0899_attach' being applied to a already existing, static definition

    The problem is that the !IS_REACHABLE() declaration of stb0899_attach()
    is a 'static inline' definition that clashes with the weak definition.

    I further observed that the bugfix was only done for one of the five users
    of stb0899_attach(), the other four still have the problem.  This reverts
    the bugfix and instead addresses the problem by not dropping the reference
```

```
    count when calling '->detach()', instead we call this function directly
    in dvb_frontend_put() before dropping the kref on the front-end.

    I first submitted this in early 2018, and after some discussion it
    was apparently discarded.  While there is a long-term plan in place,
    that plan is obviously not nearing completion yet, and the current
    kernel is still broken unless this patch is applied.

    Link: https://patchwork.kernel.org/patch/10140175/
    Link: https://patchwork.linuxtv.org/patch/54831/

    Cc: Max Kellermann <max.kellermann@gmail.com>
    Cc: Wolfgang Rohdewald <wolfgang@rohdewald.de>
    Cc: stable@vger.kernel.org
    Fixes: f686c14364ad ("[media] stb0899: move code to "detach" callback")
    Fixes: 6cdeaed3b142 ("media: dvb_usb_pctv452e: module refcount changes were unbalanced")
    Signed-off-by: Arnd Bergmann <arnd@arndb.de>
    Signed-off-by: Sean Young <sean@mess.org>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit fa829a21cfdd7b01b0ae739034b3d54e0c2a405b
Author: Francois Buergisser <fbuergisser@chromium.org>
Date:   Thu Jul 25 10:17:50 2019 -0400

    media: hantro: Set DMA max segment size

    commit c3c3509b86810293df5c524ef61421d8affc8bf0 upstream.

    The Hantro codec is typically used in platforms with an IOMMU,
    so we need to set a proper DMA segment size. Devices without an
    IOMMU will still fallback to default 64KiB segments.

    Cc: stable@vger.kernel.org
    Fixes: 775fec69008d3 ("media: add Rockchip VPU JPEG encoder driver")
    Signed-off-by: Francois Buergisser <fbuergisser@chromium.org>
    Signed-off-by: Ezequiel Garcia <ezequiel@collabora.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 29ed55517b8b7dc313ff402e085aead4766ba134
Author: Hans de Goede <hdegoede@redhat.com>
Date:   Sun Aug 18 12:03:23 2019 -0300

    media: sn9c20x: Add MSI MS-1039 laptop to flip_dmi_table

    commit 7e0bb5828311f811309bed5749528ca04992af2f upstream.

    Like a bunch of other MSI laptops the MS-1039 uses a 0c45:627b
    SN9C201 + OV7660 webcam which is mounted upside down.

    Add it to the sn9c20x flip_dmi_table to deal with this.

    Cc: stable@vger.kernel.org
    Reported-by: Rui Salvaterra <rsalvaterra@gmail.com>
    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 5579fd9d007aebd30f8a9f568f19dc756a411f2d
Author: Hans Verkuil <hverkuil-cisco@xs4all.nl>
Date:   Wed Sep 4 06:04:07 2019 -0300
```

```
    media: videobuf-core.c: poll_wait needs a non-NULL buf pointer

    commit 6f51fdfd8229d5358c2d6e272cf73478866e8ddc upstream.

    poll_wait uses &buf->done, but buf is NULL. Move the poll_wait to later
    in the function once buf is correctly set and only call it if it is
    non-NULL.

    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Reported-by: Dan Carpenter <dan.carpenter@oracle.com>
    Fixes: bb436cbeb918 ("media: videobuf: fix epoll() by calling poll_wait first")
    Cc: <stable@vger.kernel.org>      # for v5.1 and up
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 7c143549f168e8748ba603f0cc2c2c4aa8f0632b
Author: Sean Christopherson <sean.j.christopherson@intel.com>
Date:   Thu Sep 12 19:46:04 2019 -0700

    KVM: x86/mmu: Use fast invalidate mechanism to zap MMIO sptes

    commit 92f58b5c0181596d9f1e317b49ada2e728fb76eb upstream.

    Use the fast invalidate mechasim to zap MMIO sptes on a MMIO generation
    wrap.  The fast invalidate flow was reintroduced to fix a livelock bug
    in kvm_mmu_zap_all() that can occur if kvm_mmu_zap_all() is invoked when
    the guest has live vCPUs.  I.e. using kvm_mmu_zap_all() to handle the
    MMIO generation wrap is theoretically susceptible to the livelock bug.

    This effectively reverts commit 4771450c345dc ("Revert "KVM: MMU: drop
    kvm_mmu_zap_mmio_sptes""), i.e. restores the behavior of commit
    a8eca9dcc656a ("KVM: MMU: drop kvm_mmu_zap_mmio_sptes").

    Note, this actually fixes commit 571c5af06e303 ("KVM: x86/mmu:
    Voluntarily reschedule as needed when zapping MMIO sptes"), but there
    is no need to incrementally revert back to using fast invalidate, e.g.
    doing so doesn't provide any bisection or stability benefits.

    Fixes: 571c5af06e303 ("KVM: x86/mmu: Voluntarily reschedule as needed when zapping MMIO
sptes")
    Cc: stable@vger.kernel.org
    Signed-off-by: Sean Christopherson <sean.j.christopherson@intel.com>
    Signed-off-by: Paolo Bonzini <pbonzini@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 9e69dab2b47e68843e885fad82cd94207f436aff
Author: Jim Mattson <jmattson@google.com>
Date:   Thu Sep 12 09:55:03 2019 -0700

    kvm: x86: Add "significant index" flag to a few CPUID leaves

    commit a06dcd625d6181747fac7f4c140b5a4c397a778c upstream.

    According to the Intel SDM, volume 2, "CPUID," the index is
    significant (or partially significant) for CPUID leaves 0FH, 10H, 12H,
    17H, 18H, and 1FH.

    Add the corresponding flag to these CPUID leaves in do_host_cpuid().

    Signed-off-by: Jim Mattson <jmattson@google.com>
    Reviewed-by: Peter Shier <pshier@google.com>
    Reviewed-by: Steve Rutherford <srutherford@google.com>
    Fixes: a87f2d3a6eadab ("KVM: x86: Add Intel CPUID.1F cpuid emulation support")
    Reviewed-by: Krish Sadhukhan <krish.sadhukhan@oracle.com>
    Cc: stable@vger.kernel.org
```

```
    Signed-off-by: Paolo Bonzini <pbonzini@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit fe5fce29c2bd7242da5192fc728f9dc5516d28d5
Author: Alexander Graf <graf@amazon.com>
Date:   Thu Sep 5 14:58:18 2019 +0200

    KVM: x86: Disable posted interrupts for non-standard IRQs delivery modes

    commit fdcf756213756c23b533ca4974d1f48c6a4d4281 upstream.

    We can easily route hardware interrupts directly into VM context when
    they target the "Fixed" or "LowPriority" delivery modes.

    However, on modes such as "SMI" or "Init", we need to go via KVM code
    to actually put the vCPU into a different mode of operation, so we can
    not post the interrupt

    Add code in the VMX and SVM PI logic to explicitly refuse to establish
    posted mappings for advanced IRQ deliver modes. This reflects the logic
    in __apic_accept_irq() which also only ever passes Fixed and LowPriority
    interrupts as posted interrupts into the guest.

    This fixes a bug I have with code which configures real hardware to
    inject virtual SMIs into my guest.

    Signed-off-by: Alexander Graf <graf@amazon.com>
    Reviewed-by: Liran Alon <liran.alon@oracle.com>
    Reviewed-by: Sean Christopherson <sean.j.christopherson@intel.com>
    Reviewed-by: Wanpeng Li <wanpengli@tencent.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Paolo Bonzini <pbonzini@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 54dfbb9de30eca78ebd4e14dc0b32774e12ecdf3
Author: Sean Christopherson <sean.j.christopherson@intel.com>
Date:   Tue Sep 3 16:36:45 2019 -0700

    KVM: x86: Manually calculate reserved bits when loading PDPTRS

    commit 16cfacc8085782dab8e365979356ce1ca87fd6cc upstream.

    Manually generate the PDPTR reserved bit mask when explicitly loading
    PDPTRs.  The reserved bits that are being tracked by the MMU reflect the
    current paging mode, which is unlikely to be PAE paging in the vast
    majority of flows that use load_pdptrs(), e.g. CR0 and CR4 emulation,
    __set_sregs(), etc...  This can cause KVM to incorrectly signal a bad
    PDPTR, or more likely, miss a reserved bit check and subsequently fail
    a VM-Enter due to a bad VMCS.GUEST_PDPTR.

    Add a one off helper to generate the reserved bits instead of sharing
    code across the MMU's calculations and the PDPTR emulation.  The PDPTR
    reserved bits are basically set in stone, and pushing a helper into
    the MMU's calculation adds unnecessary complexity without improving
    readability.

    Oppurtunistically fix/update the comment for load_pdptrs().

    Note, the buggy commit also introduced a deliberate functional change,
    "Also remove bit 5-6 from rsvd_bits_mask per latest SDM.", which was
    effectively (and correctly) reverted by commit cd9ae5fe47df ("KVM: x86:
    Fix page-tables reserved bits").  A bit of SDM archaeology shows that
    the SDM from late 2008 had a bug (likely a copy+paste error) where it
    listed bits 6:5 as AVL and A for PDPTEs used for 4k entries but reserved
    for 2mb entries.  I.e. the SDM contradicted itself, and bits 6:5 are and
```

```
    always have been reserved.

    Fixes: 20c466b56168d ("KVM: Use rsvd_bits_mask in load_pdptrs()")
    Cc: stable@vger.kernel.org
    Cc: Nadav Amit <nadav.amit@gmail.com>
    Reported-by: Doug Reiland <doug.reiland@intel.com>
    Signed-off-by: Sean Christopherson <sean.j.christopherson@intel.com>
    Reviewed-by: Peter Xu <peterx@redhat.com>
    Signed-off-by: Paolo Bonzini <pbonzini@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3faf2647df7b1c5096607cc747255fe06ef181ec
Author: Jan Dakinevich <jan.dakinevich@virtuozzo.com>
Date:   Tue Aug 27 13:07:08 2019 +0000

    KVM: x86: set ctxt->have_exception in x86_decode_insn()

    commit c8848cee74ff05638e913582a476bde879c968ad upstream.

    x86_emulate_instruction() takes into account ctxt->have_exception flag
    during instruction decoding, but in practice this flag is never set in
    x86_decode_insn().

    Fixes: 6ea6e84309ca ("KVM: x86: inject exceptions produced by x86_decode_insn")
    Cc: stable@vger.kernel.org
    Cc: Denis Lunev <den@virtuozzo.com>
    Cc: Roman Kagan <rkagan@virtuozzo.com>
    Cc: Denis Plotnikov <dplotnikov@virtuozzo.com>
    Signed-off-by: Jan Dakinevich <jan.dakinevich@virtuozzo.com>
    Signed-off-by: Paolo Bonzini <pbonzini@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit cf6f9ff527b1cef7db145ee426b78faa1ab27223
Author: Jan Dakinevich <jan.dakinevich@virtuozzo.com>
Date:   Tue Aug 27 13:07:09 2019 +0000

    KVM: x86: always stop emulation on page fault

    commit 8530a79c5a9f4e29e6ffb35ec1a79d81f4968ec8 upstream.

    inject_emulated_exception() returns true if and only if nested page
    fault happens. However, page fault can come from guest page tables
    walk, either nested or not nested. In both cases we should stop an
    attempt to read under RIP and give guest to step over its own page
    fault handler.

    This is also visible when an emulated instruction causes a #GP fault
    and the VMware backdoor is enabled.  To handle the VMware backdoor,
    KVM intercepts #GP faults; with only the next patch applied,
    x86_emulate_instruction() injects a #GP but returns EMULATE_FAIL
    instead of EMULATE_DONE.   EMULATE_FAIL causes handle_exception_nmi()
    (or gp_interception() for SVM) to re-inject the original #GP because it
    thinks emulation failed due to a non-VMware opcode.  This patch prevents
    the issue as x86_emulate_instruction() will return EMULATE_DONE after
    injecting the #GP.

    Fixes: 6ea6e84309ca ("KVM: x86: inject exceptions produced by x86_decode_insn")
    Cc: stable@vger.kernel.org
    Cc: Denis Lunev <den@virtuozzo.com>
    Cc: Roman Kagan <rkagan@virtuozzo.com>
    Cc: Denis Plotnikov <dplotnikov@virtuozzo.com>
    Signed-off-by: Jan Dakinevich <jan.dakinevich@virtuozzo.com>
    Signed-off-by: Paolo Bonzini <pbonzini@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit 07e717d0c8050b44e32a85a650ede6068393c922
Author: Hans de Goede <hdegoede@redhat.com>
Date:   Fri Aug 23 19:48:14 2019 +0200

    platform/x86: intel_int0002_vgpio: Fix wakeups not working on Cherry Trail

    commit 1bd43d0077b9a32a8b8059036471f3fc82dae342 upstream.

    Commit 871f1f2bcb01 ("platform/x86: intel_int0002_vgpio: Only implement
    irq_set_wake on Bay Trail") removed the irq_set_wake method from the
    struct irq_chip used on Cherry Trail, but it did not set
    IRQCHIP_SKIP_SET_WAKE causing  kernel/irq/manage.c: set_irq_wake_real()
    to return -ENXIO.

    This causes the kernel to no longer see PME events reported through the
    INT0002 device as wakeup events. Which e.g. breaks wakeup by the (USB)
    keyboard on many Cherry Trail 2-in-1 devices.

    Cc: stable@vger.kernel.org
    Fixes: 871f1f2bcb01 ("platform/x86: intel_int0002_vgpio: Only implement irq_set_wake on
Bay Trail")
    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Andy Shevchenko <andriy.shevchenko@linux.intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f7eba5ed12458be5acc6529331ddc6e7f3291ca8
Author: Helge Deller <deller@gmx.de>
Date:   Thu Sep 5 16:44:17 2019 +0200

    parisc: Disable HP HSC-PCI Cards to prevent kernel crash

    commit 5fa1659105fac63e0f3c199b476025c2e04111ce upstream.

    The HP Dino PCI controller chip can be used in two variants: as on-board
    controller (e.g. in B160L), or on an Add-On card ("Card-Mode") to bridge
    PCI components to systems without a PCI bus, e.g. to a HSC/GSC bus.  One
    such Add-On card is the HP HSC-PCI Card which has one or more DEC Tulip
    PCI NIC chips connected to the on-card Dino PCI controller.

    Dino in Card-Mode has a big disadvantage: All PCI memory accesses need
    to go through the DINO_MEM_DATA register, so Linux drivers will not be
    able to use the ioremap() function. Without ioremap() many drivers will
    not work, one example is the tulip driver which then simply crashes the
    kernel if it tries to access the ports on the HP HSC card.

    This patch disables the HP HSC card if it finds one, and as such
    fixes the kernel crash on a HP D350/2 machine.

    Signed-off-by: Helge Deller <deller@gmx.de>
    Noticed-by: Phil Scarr <phil.scarr@pm.me>
    Cc: stable@vger.kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ec21729789eb246ff01ef48ffbe2b293572fba90
Author: Tejun Heo <tj@kernel.org>
Date:   Sun Sep 22 06:19:36 2019 -0700

    fuse: fix beyond-end-of-page access in fuse_parse_cache()

    commit e5854b1cdf6cb48a20e01e3bdad0476a4c60a077 upstream.

    With DEBUG_PAGEALLOC on, the following triggers.

      BUG: unable to handle page fault for address: ffff88859367c000
      #PF: supervisor read access in kernel mode
```

```
       #PF: error_code(0x0000) - not-present page
       PGD 3001067 P4D 3001067 PUD 406d3a8067 PMD 406d30c067 PTE 800ffffa6c983060
       Oops: 0000 [#1] SMP DEBUG_PAGEALLOC
       CPU: 38 PID: 3110657 Comm: python2.7
       RIP: 0010:fuse_readdir+0x88f/0xe7a [fuse]
       Code: 49 8b 4d 08 49 39 4e 60 0f 84 44 04 00 00 48 8b 43 08 43 8d 1c 3c 4d 01 7e 68 49
 89 dc 48 03 5c 24 38 49 89 46 60 8b 44 24 30 <8b> 4b 10 44 29 e0 48 89 ca 48 83 c1 1f 48 83
 e1 f8 83 f8 17 49 89
       RSP: 0018:ffffc90035edbde0 EFLAGS: 00010286
       RAX: 0000000000001000 RBX: ffff88859367bff0 RCX: 0000000000000000
       RDX: 0000000000000000 RSI: ffff88859367bfed RDI: 0000000000920907
       RBP: ffffc90035edbe90 R08: 000000000000014b R09: 0000000000000004
       R10: ffff88859367b000 R11: 0000000000000000 R12: 0000000000000ff0
       R13: ffffc90035edbee0 R14: ffff889fb8546180 R15: 0000000000000020
       FS:  00007f80b5f4a740(0000) GS:ffff889fffa00000(0000) knlGS:0000000000000000
       CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
       CR2: ffff88859367c000 CR3: 0000001c170c2001 CR4: 00000000003606e0
       DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000
       DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400
       Call Trace:
        iterate_dir+0x122/0x180
        __x64_sys_getdents+0xa6/0x140
        do_syscall_64+0x42/0x100
        entry_SYSCALL_64_after_hwframe+0x44/0xa9

    It's in fuse_parse_cache().  %rbx (ffff88859367bff0) is fuse_dirent
    pointer - addr + offset.  FUSE_DIRENT_SIZE() is trying to dereference
    namelen off of it but that derefs into the next page which is disabled
    by pagealloc debug causing a PF.

    This is caused by dirent->namelen being accessed before ensuring that
    there's enough bytes in the page for the dirent.  Fix it by pushing
    down reclen calculation.

    Signed-off-by: Tejun Heo <tj@kernel.org>
    Fixes: 5d7bc7e8680c ("fuse: allow using readdir cache")
    Cc: stable@vger.kernel.org # v4.20+
    Signed-off-by: Miklos Szeredi <mszeredi@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 2487800d0c02564ca6fdb77db48f0e9977b02ba7
Author: Vasily Averin <vvs@virtuozzo.com>
Date:   Fri Sep 13 18:17:11 2019 +0300

    fuse: fix missing unlock_page in fuse_writepage()

    commit d5880c7a8620290a6c90ced7a0e8bd0ad9419601 upstream.

    unlock_page() was missing in case of an already in-flight write against the
    same page.

    Signed-off-by: Vasily Averin <vvs@virtuozzo.com>
    Fixes: ff17be086477 ("fuse: writepage: skip already in flight")
    Cc: <stable@vger.kernel.org> # v3.13
    Signed-off-by: Miklos Szeredi <mszeredi@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 18df842f49c7df33b6fc926c47c7b6c5d4d226c2
Author: Eric Biggers <ebiggers@google.com>
Date:   Sun Sep 8 20:15:18 2019 -0700

    fuse: fix deadlock with aio poll and fuse_iqueue::waitq.lock

    commit 76e43c8ccaa35c30d5df853013561145a0f750a5 upstream.
```

```
    When IOCB_CMD_POLL is used on the FUSE device, aio_poll() disables IRQs
    and takes kioctx::ctx_lock, then fuse_iqueue::waitq.lock.

    This may have to wait for fuse_iqueue::waitq.lock to be released by one
    of many places that take it with IRQs enabled.  Since the IRQ handler
    may take kioctx::ctx_lock, lockdep reports that a deadlock is possible.

    Fix it by protecting the state of struct fuse_iqueue with a separate
    spinlock, and only accessing fuse_iqueue::waitq using the versions of
    the waitqueue functions which do IRQ-safe locking internally.

    Reproducer:

            #include <fcntl.h>
            #include <stdio.h>
            #include <sys/mount.h>
            #include <sys/stat.h>
            #include <sys/syscall.h>
            #include <unistd.h>
            #include <linux/aio_abi.h>

            int main()
            {
                    char opts[128];
                    int fd = open("/dev/fuse", O_RDWR);
                    aio_context_t ctx = 0;
                    struct iocb cb = { .aio_lio_opcode = IOCB_CMD_POLL, .aio_fildes = fd };
                    struct iocb *cbp = &cb;

                    sprintf(opts, "fd=%d,rootmode=040000,user_id=0,group_id=0", fd);
                    mkdir("mnt", 0700);
                    mount("foo",  "mnt", "fuse", 0, opts);
                    syscall(__NR_io_setup, 1, &ctx);
                    syscall(__NR_io_submit, ctx, 1, &cbp);
            }

    Beginning of lockdep output:

            ======================================================
            WARNING: SOFTIRQ-safe -> SOFTIRQ-unsafe lock order detected
            5.3.0-rc5 #9 Not tainted
            ------------------------------------------------------
            syz_fuse/135 [HC0[0]:SC0[0]:HE0:SE1] is trying to acquire:
            000000003590ceda (&fiq->waitq){+.+.}, at: spin_lock include/linux/spinlock.h:338
 [inline]
            000000003590ceda (&fiq->waitq){+.+.}, at: aio_poll fs/aio.c:1751 [inline]
            000000003590ceda (&fiq->waitq){+.+.}, at:
 __io_submit_one.constprop.0+0x203/0x5b0 fs/aio.c:1825

            and this task is already holding:
            0000000075037284 (&(&ctx->ctx_lock)->rlock){..-.}, at: spin_lock_irq
 include/linux/spinlock.h:363 [inline]
            0000000075037284 (&(&ctx->ctx_lock)->rlock){..-.}, at: aio_poll fs/aio.c:1749
 [inline]
            0000000075037284 (&(&ctx->ctx_lock)->rlock){..-.}, at:
 __io_submit_one.constprop.0+0x1f4/0x5b0 fs/aio.c:1825
            which would create a new lock dependency:
             (&(&ctx->ctx_lock)->rlock){..-.} -> (&fiq->waitq){+.+.}

            but this new dependency connects a SOFTIRQ-irq-safe lock:
             (&(&ctx->ctx_lock)->rlock){..-.}

            [...]

    Reported-by: syzbot+af05535bb79520f95431@syzkaller.appspotmail.com
```

```
    Reported-by: syzbot+d86c4426a01f60feddc7@syzkaller.appspotmail.com
    Fixes: bfe4037e722e ("aio: implement IOCB_CMD_POLL")
    Cc: <stable@vger.kernel.org> # v4.19+
    Cc: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Eric Biggers <ebiggers@google.com>
    Signed-off-by: Miklos Szeredi <mszeredi@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 458b88ad00e437c3b948b6689e1e8a3830bc06a4
Author: Jarkko Sakkinen <jarkko.sakkinen@linux.intel.com>
Date:   Mon Sep 16 11:38:34 2019 +0300

    tpm: Wrap the buffer from the caller to tpm_buf in tpm_send()

    commit e13cd21ffd50a07b55dcc4d8c38cedf27f28eaa1 upstream.

    tpm_send() does not give anymore the result back to the caller. This
    would require another memcpy(), which kind of tells that the whole
    approach is somewhat broken. Instead, as Mimi suggested, this commit
    just wraps the data to the tpm_buf, and thus the result will not go to
    the garbage.

    Obviously this assumes from the caller that it passes large enough
    buffer, which makes the whole API somewhat broken because it could be
    different size than @buflen but since trusted keys is the only module
    using this API right now I think that this fix is sufficient for the
    moment.

    In the near future the plan is to replace the parameters with a tpm_buf
    created by the caller.

    Reported-by: Mimi Zohar <zohar@linux.ibm.com>
    Suggested-by: Mimi Zohar <zohar@linux.ibm.com>
    Cc: stable@vger.kernel.org
    Fixes: 412eb585587a ("use tpm_buf in tpm_transmit_cmd() as the IO parameter")
    Signed-off-by: Jarkko Sakkinen <jarkko.sakkinen@linux.intel.com>
    Reviewed-by: Jerry Snitselaar <jsnitsel@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e4a8d7a6f95fbb19245f1443b4acb34020ea89b5
Author: Stefan Berger <stefanb@linux.ibm.com>
Date:   Thu Aug 29 20:09:06 2019 -0400

    tpm_tis_core: Set TPM_CHIP_FLAG_IRQ before probing for interrupts

    commit 1ea32c83c699df32689d329b2415796b7bfc2f6e upstream.

    The tpm_tis_core has to set the TPM_CHIP_FLAG_IRQ before probing for
    interrupts since there is no other place in the code that would set
    it.

    Cc: linux-stable@vger.kernel.org
    Fixes: 570a36097f30 ("tpm: drop 'irq' from struct tpm_vendor_specific")
    Signed-off-by: Stefan Berger <stefanb@linux.ibm.com>
    Signed-off-by: Jarkko Sakkinen <jarkko.sakkinen@linux.intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 7f064c378e2c8c848c7acc3ebba7ec45df1c5492
Author: Stefan Berger <stefanb@linux.ibm.com>
Date:   Tue Aug 20 08:25:17 2019 -0400

    tpm_tis_core: Turn on the TPM before probing IRQ's

    commit 5b359c7c43727e624eac3efc7ad21bd2defea161 upstream.
```

```
    The interrupt probing sequence in tpm_tis_core cannot obviously run with
    the TPM power gated. Power on the TPM with tpm_chip_start() before
    probing IRQ's. Turn it off once the probing is complete.

    Cc: linux-stable@vger.kernel.org
    Fixes: a3fbfae82b4c ("tpm: take TPM chip power gating out of tpm_transmit()")
    Signed-off-by: Stefan Berger <stefanb@linux.ibm.com>
    Reviewed-by: Jarkko Sakkinen <jarkko.sakkinen@linux.intel.com>
    Signed-off-by: Jarkko Sakkinen <jarkko.sakkinen@linux.intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 66252f7ac474e559a3020fb421be544adb056bbb
Author: Madhavan Srinivasan <maddy@linux.vnet.ibm.com>
Date:   Tue Aug 27 15:46:35 2019 +0530

    powerpc/imc: Dont create debugfs files for cpu-less nodes

    commit 41ba17f20ea835c489e77bd54e2da73184e22060 upstream.

    Commit <684d984038aa> ('powerpc/powernv: Add debugfs interface for
    imc-mode and imc') added debugfs interface for the nest imc pmu
    devices to support changing of different ucode modes. Primarily adding
    this capability for debug. But when doing so, the code did not
    consider the case of cpu-less nodes. So when reading the _cmd_ or
    _mode_ file of a cpu-less node will create this crash.

      Faulting instruction address: 0xc0000000000d0d58
      Oops: Kernel access of bad area, sig: 11 [#1]
      ...
      CPU: 67 PID: 5301 Comm: cat Not tainted 5.2.0-rc6-next-20190627+ #19
      NIP:  c0000000000d0d58 LR: c00000000049aa18 CTR:c0000000000d0d50
      REGS: c00020194548f9e0 TRAP: 0300   Not tainted  (5.2.0-rc6-next-20190627+)
      MSR:  9000000000009033 <SF,HV,EE,ME,IR,DR,RI,LE>  CR:28022822  XER: 00000000
      CFAR: c00000000049aa14 DAR: 000000000003fc08 DSISR:40000000 IRQMASK: 0
      ...
      NIP imc_mem_get+0x8/0x20
      LR  simple_attr_read+0x118/0x170
      Call Trace:
        simple_attr_read+0x70/0x170 (unreliable)
        debugfs_attr_read+0x6c/0xb0
        __vfs_read+0x3c/0x70
         vfs_read+0xbc/0x1a0
        ksys_read+0x7c/0x140
        system_call+0x5c/0x70

    Patch fixes the issue with a more robust check for vbase to NULL.

    Before patch, ls output for the debugfs imc directory

      # ls /sys/kernel/debug/powerpc/imc/
      imc_cmd_0    imc_cmd_251  imc_cmd_253  imc_cmd_255  imc_mode_0    imc_mode_251
imc_mode_253  imc_mode_255
      imc_cmd_250  imc_cmd_252  imc_cmd_254  imc_cmd_8    imc_mode_250  imc_mode_252
imc_mode_254  imc_mode_8

    After patch, ls output for the debugfs imc directory

      # ls /sys/kernel/debug/powerpc/imc/
      imc_cmd_0  imc_cmd_8  imc_mode_0  imc_mode_8

    Actual bug here is that, we have two loops with potentially different
    loop counts. That is, in imc_get_mem_addr_nest(), loop count is
    obtained from the dt entries. But in case of export_imc_mode_and_cmd(),
    loop was based on for_each_nid() count. Patch fixes the loop count in
    latter based on the struct mem_info. Ideally it would be better to
```

```
    have array size in struct imc_pmu.

    Fixes: 684d984038aa ('powerpc/powernv: Add debugfs interface for imc-mode and imc')
    Reported-by: Qian Cai <cai@lca.pw>
    Suggested-by: Michael Ellerman <mpe@ellerman.id.au>
    Signed-off-by: Madhavan Srinivasan <maddy@linux.vnet.ibm.com>
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Link: https://lore.kernel.org/r/20190827101635.6942-1-maddy@linux.vnet.ibm.com
    Cc: Jan Stancek <jstancek@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b24e3e392e16d8531ad10f2b2c722b2ff41ca342
Author: Ming Lei <ming.lei@redhat.com>
Date:   Thu Jul 25 10:05:00 2019 +0800

    scsi: implement .cleanup_rq callback

    [ Upstream commit b7e9e1fb7a9227be34ad4a5e778022c3164494cf ]

    Implement .cleanup_rq() callback for freeing driver private part
    of the request. Then we can avoid to leak this part if the request isn't
    completed by SCSI, and freed by blk-mq or upper layer(such as dm-rq) finally.

    Cc: Ewan D. Milne <emilne@redhat.com>
    Cc: Bart Van Assche <bvanassche@acm.org>
    Cc: Hannes Reinecke <hare@suse.com>
    Cc: Christoph Hellwig <hch@lst.de>
    Cc: Mike Snitzer <snitzer@redhat.com>
    Cc: dm-devel@redhat.com
    Cc: <stable@vger.kernel.org>
    Fixes: 396eaf21ee17 ("blk-mq: improve DM's blk-mq IO merging via
blk_insert_cloned_request feedback")
    Signed-off-by: Ming Lei <ming.lei@redhat.com>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e7febd8844c0e1628e2b349271220c03fd7a9408
Author: Ming Lei <ming.lei@redhat.com>
Date:   Thu Jul 25 10:04:59 2019 +0800

    blk-mq: add callback of .cleanup_rq

    [ Upstream commit 226b4fc75c78f9c497c5182d939101b260cfb9f3 ]

    SCSI maintains its own driver private data hooked off of each SCSI
    request, and the pridate data won't be freed after scsi_queue_rq()
    returns BLK_STS_RESOURCE or BLK_STS_DEV_RESOURCE. An upper layer driver
    (e.g. dm-rq) may need to retry these SCSI requests, before SCSI has
    fully dispatched them, due to a lower level SCSI driver's resource
    limitation identified in scsi_queue_rq(). Currently SCSI's per-request
    private data is leaked when the upper layer driver (dm-rq) frees and
    then retries these requests in response to BLK_STS_RESOURCE or
    BLK_STS_DEV_RESOURCE returns from scsi_queue_rq().

    This usecase is so specialized that it doesn't warrant training an
    existing blk-mq interface (e.g. blk_mq_free_request) to allow SCSI to
    account for freeing its driver private data -- doing so would add an
    extra branch for handling a special case that all other consumers of
    SCSI (and blk-mq) won't ever need to worry about.

    So the most pragmatic way forward is to delegate freeing SCSI driver
    private data to the upper layer driver (dm-rq).  Do so by adding
    new .cleanup_rq callback and calling a new blk_mq_cleanup_rq() method
    from dm-rq.  A following commit will implement the .cleanup_rq() hook
    in scsi_mq_ops.
```

```
    Cc: Ewan D. Milne <emilne@redhat.com>
    Cc: Bart Van Assche <bvanassche@acm.org>
    Cc: Hannes Reinecke <hare@suse.com>
    Cc: Christoph Hellwig <hch@lst.de>
    Cc: Mike Snitzer <snitzer@redhat.com>
    Cc: dm-devel@redhat.com
    Cc: <stable@vger.kernel.org>
    Fixes: 396eaf21ee17 ("blk-mq: improve DM's blk-mq IO merging via
blk_insert_cloned_request feedback")
    Signed-off-by: Ming Lei <ming.lei@redhat.com>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d24bb3c17c39b7be0abf764eb52adfca90539596
Author: Jan-Marek Glogowski <glogow@fbihome.de>
Date:   Sun Sep 15 16:57:28 2019 +0200

    ALSA: hda/realtek - PCI quirk for Medion E4254

    [ Upstream commit bd9c10bc663dd2eaac8fe39dad0f18cd21527446 ]

    The laptop has a combined jack to attach headsets on the right.
    The BIOS encodes them as two different colored jacks at the front,
    but otherwise it seems to be configured ok. But any adaption of
    the pins config on its own doesn't fix the jack detection to work
    in Linux. Still Windows works correct.

    This is somehow fixed by chaining ALC256_FIXUP_ASUS_HEADSET_MODE,
    which seems to register the microphone jack as a headset part and
    also results in fixing jack sensing, visible in dmesg as:

    -snd_hda_codec_realtek hdaudioC0D0:      Mic=0x19
    +snd_hda_codec_realtek hdaudioC0D0:      Headset Mic=0x19

    [ Actually the essential change is the location of the jack; the
      driver created "Front Mic Jack" without the matching volume / mute
      control element due to its jack location, which confused PA.
      -- tiwai ]

    Signed-off-by: Jan-Marek Glogowski <glogow@fbihome.de>
    Cc: <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/8f4f9b20-0aeb-f8f1-c02f-fd53c09679f1@fbihome.de
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1df66fe4e9eabdb61ae5df3ab9706108c3ca7cb6
Author: Peter Zijlstra <peterz@infradead.org>
Date:   Thu Aug 1 12:42:06 2019 +0200

    rcu/tree: Fix SCHED_FIFO params

    [ Upstream commit 130d9c331bc59a8733b47c58ef197a2b1fa3ed43 ]

    A rather embarrasing mistake had us call sched_setscheduler() before
    initializing the parameters passed to it.

    Fixes: 1a763fd7c633 ("rcu/tree: Call setschedule() gp ktread to SCHED_FIFO outside of
atomic region")
    Signed-off-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Reviewed-by: Paul E. McKenney <paulmck@linux.ibm.com>
    Cc: Juri Lelli <juri.lelli@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a942ec89ebb43766b4610ea819828e82a8d42172
```

```
Author: Adam Ford <aford173@gmail.com>
Date:   Wed Aug 28 13:33:51 2019 -0500

    ARM: dts: am3517-evm: Fix missing video

    commit 24cf23276a54dd2825d3e3965c1b1b453e2a113d upstream.

    A previous commit removed the panel-dpi driver, which made the
    video on the AM3517-evm stop working because it relied on the dpi
    driver for setting video timings.  Now that the simple-panel driver
    is available in omap2plus, this patch migrates the am3517-evm
    to use a similar panel and remove the manual timing requirements.

    Fixes: 8bf4b1621178 ("drm/omap: Remove panel-dpi driver")

    Signed-off-by: Adam Ford <aford173@gmail.com>
    Signed-off-by: Tony Lindgren <tony@atomide.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit cb526d1c5326fa0683d5ef1ea4db5ae12377fcab
Author: Joonwon Kang <kjw1627@gmail.com>
Date:   Sun Jul 28 00:58:41 2019 +0900

    randstruct: Check member structs in is_pure_ops_struct()

    commit 60f2c82ed20bde57c362e66f796cf9e0e38a6dbb upstream.

    While no uses in the kernel triggered this case, it was possible to have
    a false negative where a struct contains other structs which contain only
    function pointers because of unreachable code in is_pure_ops_struct().

    Signed-off-by: Joonwon Kang <kjw1627@gmail.com>
    Link: https://lore.kernel.org/r/20190727155841.GA13586@host
    Fixes: 313dd1b62921 ("gcc-plugins: Add the randstruct plugin")
    Cc: stable@vger.kernel.org
    Signed-off-by: Kees Cook <keescook@chromium.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 102d735feeb198466f5f86670bc77c58bfc524a0
Author: Jack Morgenstein <jackm@dev.mellanox.co.il>
Date:   Mon Sep 16 10:11:54 2019 +0300

    RDMA: Fix double-free in srq creation error flow

    commit 3eca7fc2d8d1275d9cf0c709f0937becbfcf6d96 upstream.

    The cited commit introduced a double-free of the srq buffer in the error
    flow of procedure __uverbs_create_xsrq().

    The problem is that ib_destroy_srq_user() called in the error flow also
    frees the srq buffer.

    Thus, if uverbs_response() fails in __uverbs_create_srq(), the srq buffer
    will be freed twice.

    Cc: <stable@vger.kernel.org>
    Fixes: 68e326dea1db ("RDMA: Handle SRQ allocations by IB/core")
    Link: https://lore.kernel.org/r/20190916071154.20383-5-leon@kernel.org
    Signed-off-by: Jack Morgenstein <jackm@dev.mellanox.co.il>
    Signed-off-by: Leon Romanovsky <leonro@mellanox.com>
    Reviewed-by: Jason Gunthorpe <jgg@mellanox.com>
    Signed-off-by: Jason Gunthorpe <jgg@mellanox.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 5b4651af6a079032c60c532e3b3c0b3a9de95bb1
```

```
Author: Kaike Wan <kaike.wan@intel.com>
Date:   Mon Jul 15 12:45:46 2019 -0400

    IB/hfi1: Do not update hcrc for a KDETH packet during fault injection

    commit b2590bdd0b1dfb91737e6cb07ebb47bd74957f7e upstream.

    When a KDETH packet is subject to fault injection during transmission,
    HCRC is supposed to be omitted from the packet so that the hardware on the
    receiver side would drop the packet. When creating pbc, the PbcInsertHcrc
    field is set to be PBC_IHCRC_NONE if the KDETH packet is subject to fault
    injection, but overwritten with PBC_IHCRC_LKDETH when update_hcrc() is
    called later.

    This problem is fixed by not calling update_hcrc() when the packet is
    subject to fault injection.

    Fixes: 6b6cf9357f78 ("IB/hfi1: Set PbcInsertHcrc for TID RDMA packets")
    Cc: <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20190715164546.74174.99296.stgit@awfm-01.aw.intel.com
    Reviewed-by: Mike Marciniszyn <mike.marciniszyn@intel.com>
    Signed-off-by: Kaike Wan <kaike.wan@intel.com>
    Signed-off-by: Mike Marciniszyn <mike.marciniszyn@intel.com>
    Signed-off-by: Jason Gunthorpe <jgg@mellanox.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 41e839b7b3f65effea06e5d5e809b888ea8e446d
Author: Ira Weiny <ira.weiny@intel.com>
Date:   Wed Sep 11 07:30:53 2019 -0400

    IB/hfi1: Define variables as unsigned long to fix KASAN warning

    commit f8659d68e2bee5b86a1beaf7be42d942e1fc81f4 upstream.

    Define the working variables to be unsigned long to be compatible with
    for_each_set_bit and change types as needed.

    While we are at it remove unused variables from a couple of functions.

    This was found because of the following KASAN warning:
     ==================================================================
       BUG: KASAN: stack-out-of-bounds in find_first_bit+0x19/0x70
       Read of size 8 at addr ffff888362d778d0 by task kworker/u308:2/1889

       CPU: 21 PID: 1889 Comm: kworker/u308:2 Tainted: G W         5.3.0-rc2-mm1+ #2
       Hardware name: Intel Corporation W2600CR/W2600CR, BIOS
SE5C600.86B.02.04.0003.102320141138 10/23/2014
       Workqueue: ib-comp-unb-wq ib_cq_poll_work [ib_core]
       Call Trace:
        dump_stack+0x9a/0xf0
        ? find_first_bit+0x19/0x70
        print_address_description+0x6c/0x332
        ? find_first_bit+0x19/0x70
        ? find_first_bit+0x19/0x70
        __kasan_report.cold.6+0x1a/0x3b
        ? find_first_bit+0x19/0x70
        kasan_report+0xe/0x12
        find_first_bit+0x19/0x70
        pma_get_opa_portstatus+0x5cc/0xa80 [hfi1]
        ? ret_from_fork+0x3a/0x50
        ? pma_get_opa_port_ectrs+0x200/0x200 [hfi1]
        ? stack_trace_consume_entry+0x80/0x80
        hfi1_process_mad+0x39b/0x26c0 [hfi1]
        ? __lock_acquire+0x65e/0x21b0
        ? clear_linkup_counters+0xb0/0xb0 [hfi1]
```

```
         ? check_chain_key+0x1d7/0x2e0
         ? lock_downgrade+0x3a0/0x3a0
         ? match_held_lock+0x2e/0x250
         ib_mad_recv_done+0x698/0x15e0 [ib_core]
         ? clear_linkup_counters+0xb0/0xb0 [hfi1]
         ? ib_mad_send_done+0xc80/0xc80 [ib_core]
         ? mark_held_locks+0x79/0xa0
         ? _raw_spin_unlock_irqrestore+0x44/0x60
         ? rvt_poll_cq+0x1e1/0x340 [rdmavt]
         __ib_process_cq+0x97/0x100 [ib_core]
         ib_cq_poll_work+0x31/0xb0 [ib_core]
         process_one_work+0x4ee/0xa00
         ? pwq_dec_nr_in_flight+0x110/0x110
         ? do_raw_spin_lock+0x113/0x1d0
         worker_thread+0x57/0x5a0
         ? process_one_work+0xa00/0xa00
         kthread+0x1bb/0x1e0
         ? kthread_create_on_node+0xc0/0xc0
         ret_from_fork+0x3a/0x50

        The buggy address belongs to the page:
        page:ffffea000d8b5dc0 refcount:0 mapcount:0 mapping:0000000000000000 index:0x0
        flags: 0x17ffffc0000000()
        raw: 0017ffffc0000000 0000000000000000 ffffea000d8b5dc8 0000000000000000
        raw: 0000000000000000 0000000000000000 00000000ffffffff 0000000000000000
        page dumped because: kasan: bad access detected

        addr ffff888362d778d0 is located in stack of task kworker/u308:2/1889 at offset 32 in
 frame:
         pma_get_opa_portstatus+0x0/0xa80 [hfi1]

        this frame has 1 object:
         [32, 36) 'vl_select_mask'

        Memory state around the buggy address:
         ffff888362d77780: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
         ffff888362d77800: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
        >ffff888362d77880: 00 00 00 00 00 00 f1 f1 f1 f1 04 f2 f2 f2 00 00
                                                    ^
         ffff888362d77900: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
         ffff888362d77980: 00 00 00 00 00 00 00 00 f1 f1 f1 f1 04 f2 f2 f2

      ==================================================================

    Cc: <stable@vger.kernel.org>
    Fixes: 7724105686e7 ("IB/hfi1: add driver files")
    Link: https://lore.kernel.org/r/20190911113053.126040.47327.stgit@awfm-01.aw.intel.com
    Reviewed-by: Mike Marciniszyn <mike.marciniszyn@intel.com>
    Signed-off-by: Ira Weiny <ira.weiny@intel.com>
    Signed-off-by: Kaike Wan <kaike.wan@intel.com>
    Signed-off-by: Dennis Dalessandro <dennis.dalessandro@intel.com>
    Signed-off-by: Jason Gunthorpe <jgg@mellanox.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

 commit 43229d88e68b330b56316e677fcec1edc97582ff
 Author: Danit Goldberg <danitg@mellanox.com>
 Date:   Mon Sep 16 09:48:18 2019 +0300

    IB/mlx5: Free mpi in mp_slave mode

    commit 5d44adebbb7e785939df3db36ac360f5e8b73e44 upstream.

    ib_add_slave_port() allocates a multiport struct but never frees it.
    Don't leak memory, free the allocated mpi struct during driver unload.
```

```
    Cc: <stable@vger.kernel.org>
    Fixes: 32f69e4be269 ("{net, IB}/mlx5: Manage port association for multiport RoCE")
    Link: https://lore.kernel.org/r/20190916064818.19823-3-leon@kernel.org
    Signed-off-by: Danit Goldberg <danitg@mellanox.com>
    Reviewed-by: Jason Gunthorpe <jgg@mellanox.com>
    Signed-off-by: Jason Gunthorpe <jgg@mellanox.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3800ee3bfa530b9400dac629620838058fd63af0
Author: Vincent Whitchurch <vincent.whitchurch@axis.com>
Date:   Thu Jul 11 16:29:37 2019 +0200

    printk: Do not lose last line in kmsg buffer dump

    commit c9dccacfccc72c32692eedff4a27a4b0833a2afd upstream.

    kmsg_dump_get_buffer() is supposed to select all the youngest log
    messages which fit into the provided buffer.  It determines the correct
    start index by using msg_print_text() with a NULL buffer to calculate
    the size of each entry.  However, when performing the actual writes,
    msg_print_text() only writes the entry to the buffer if the written len
    is lesser than the size of the buffer.  So if the lengths of the
    selected youngest log messages happen to precisely fill up the provided
    buffer, the last log message is not included.

    We don't want to modify msg_print_text() to fill up the buffer and start
    returning a length which is equal to the size of the buffer, since
    callers of its other users, such as kmsg_dump_get_line(), depend upon
    the current behaviour.

    Instead, fix kmsg_dump_get_buffer() to compensate for this.

    For example, with the following two final prints:

    [    6.427502] AAAAAAAAAAAAA
    [    6.427769] BBBBBBBB12345

    A dump of a 64-byte buffer filled by kmsg_dump_get_buffer(), before this
    patch:

     00000000: 3c 30 3e 5b 20 20 20 20 36 2e 35 32 32 31 39 37  <0>[    6.522197
     00000010: 5d 20 41 41 41 41 41 41 41 41 41 41 41 41 41 0a  ] AAAAAAAAAAAAA.
     00000020: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
     00000030: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................

    After this patch:

     00000000: 3c 30 3e 5b 20 20 20 20 36 2e 34 35 36 36 37 38  <0>[    6.456678
     00000010: 5d 20 42 42 42 42 42 42 42 42 31 32 33 34 35 0a  ] BBBBBBBB12345.
     00000020: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
     00000030: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................

    Link: http://lkml.kernel.org/r/20190711142937.4083-1-vincent.whitchurch@axis.com
    Fixes: e2ae715d66bf4bec ("kmsg - kmsg_dump() use iterator to receive log buffer
content")
    To: rostedt@goodmis.org
    Cc: linux-kernel@vger.kernel.org
    Cc: <stable@vger.kernel.org> # v3.5+
    Signed-off-by: Vincent Whitchurch <vincent.whitchurch@axis.com>
    Reviewed-by: Sergey Senozhatsky <sergey.senozhatsky@gmail.com>
    Signed-off-by: Petr Mladek <pmladek@suse.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 04c4056733795b9ce828674366f2560388dc2b0a
Author: Quinn Tran <qutran@marvell.com>
```

```
Date:   Fri Jul 26 09:07:32 2019 -0700

    scsi: qla2xxx: Fix Relogin to prevent modifying scan_state flag

    commit 8b5292bcfcacf15182a77a973a98d310e76fd58b upstream.

    Relogin fails to move forward due to scan_state flag indicating device is
    not there. Before relogin process, Session delete process accidently
    modified the scan_state flag.

    [mkp: typos plus corrected Fixes: sha as reported by sfr]

    Fixes: 2dee5521028c ("scsi: qla2xxx: Fix login state machine freeze")
    Cc: stable@vger.kernel.org
    Signed-off-by: Quinn Tran <qutran@marvell.com>
    Signed-off-by: Himanshu Madhani <hmadhani@marvell.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 66b9b89d009e2bcfdce0049daa5ff910cb9dd9b5
Author: Martin Wilck <Martin.Wilck@suse.com>
Date:   Wed Sep 4 15:52:29 2019 +0000

    scsi: scsi_dh_rdac: zero cdb in send_mode_select()

    commit 57adf5d4cfd3198aa480e7c94a101fc8c4e6109d upstream.

    cdb in send_mode_select() is not zeroed and is only partially filled in
    rdac_failover_get(), which leads to some random data getting to the
    device. Users have reported storage responding to such commands with
    INVALID FIELD IN CDB. Code before commit 327825574132 was not affected, as
    it called blk_rq_set_block_pc().

    Fix this by zeroing out the cdb first.

    Identified & fix proposed by HPE.

    Fixes: 327825574132 ("scsi_dh_rdac: switch to scsi_execute_req_flags()")
    Cc: stable@vger.kernel.org
    Link: https://lore.kernel.org/r/20190904155205.1666-1-martin.wilck@suse.com
    Signed-off-by: Martin Wilck <mwilck@suse.com>
    Acked-by: Ales Novak <alnovak@suse.cz>
    Reviewed-by: Shane Seymour <shane.seymour@hpe.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 4d97ddde96f63ac32403f3b23db0acb06f577496
Author: Takashi Sakamoto <o-takashi@sakamocchi.jp>
Date:   Tue Sep 10 22:51:52 2019 +0900

    ALSA: firewire-tascam: check intermediate state of clock status and retry

    commit e1a00b5b253a4f97216b9a33199a863987075162 upstream.

    2 bytes in MSB of register for clock status is zero during intermediate
    state after changing status of sampling clock in models of TASCAM FireWire
    series. The duration of this state differs depending on cases. During the
    state, it's better to retry reading the register for current status of
    the clock.

    In current implementation, the intermediate state is checked only when
    getting current sampling transmission frequency, then retry reading.
    This care is required for the other operations to read the register.

    This commit moves the codes of check and retry into helper function
```

```
    commonly used for operations to read the register.

    Fixes: e453df44f0d6 ("ALSA: firewire-tascam: add PCM functionality")
    Cc: <stable@vger.kernel.org> # v4.4+
    Signed-off-by: Takashi Sakamoto <o-takashi@sakamocchi.jp>
    Link: https://lore.kernel.org/r/20190910135152.29800-3-o-takashi@sakamocchi.jp
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit d714bc1461c674983d666dbdbfd02b53e5f582ed
Author: Takashi Sakamoto <o-takashi@sakamocchi.jp>
Date:   Tue Sep 10 22:51:51 2019 +0900

    ALSA: firewire-tascam: handle error code when getting current source of clock

    commit 2617120f4de6d0423384e0e86b14c78b9de84d5a upstream.

    The return value of snd_tscm_stream_get_clock() is ignored. This commit
    checks the value and handle error.

    Fixes: e453df44f0d6 ("ALSA: firewire-tascam: add PCM functionality")
    Cc: <stable@vger.kernel.org> # v4.4+
    Signed-off-by: Takashi Sakamoto <o-takashi@sakamocchi.jp>
    Link: https://lore.kernel.org/r/20190910135152.29800-2-o-takashi@sakamocchi.jp
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3a0b7157d6a9833cefd40b5f0fa1cc90285bb4b5
Author: Luca Coelho <luciano.coelho@intel.com>
Date:   Tue Sep 24 13:30:57 2019 +0300

    iwlwifi: fw: don't send GEO_TX_POWER_LIMIT command to FW version 36

    commit fddbfeece9c7882cc47754c7da460fe427e3e85b upstream.

    The intention was to have the GEO_TX_POWER_LIMIT command in FW version
    36 as well, but not all 8000 family got this feature enabled.  The
    8000 family is the only one using version 36, so skip this version
    entirely.  If we try to send this command to the firmwares that do not
    support it, we get a BAD_COMMAND response from the firmware.

    This fixes https://bugzilla.kernel.org/show_bug.cgi?id=204151.

    Cc: stable@vger.kernel.org # 4.19+
    Signed-off-by: Luca Coelho <luciano.coelho@intel.com>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e05708ab85bfb50226cc8e04cd8023ff23b8e55c
Author: Adam Ford <aford173@gmail.com>
Date:   Wed Aug 28 13:33:49 2019 -0500

    ARM: omap2plus_defconfig: Fix missing video

    commit 4957eccf979b025286b39388fd1a60cde601a10a upstream.

    When the panel-dpi driver was removed, the simple-panels driver
    was never enabled, so anyone who used the panel-dpi driver lost
    video, and those who used it inconjunction with simple-panels
    would have to manually enable CONFIG_DRM_PANEL_SIMPLE.

    This patch makes CONFIG_DRM_PANEL_SIMPLE a module in the same
    way the deprecated panel-dpi was.

    Fixes: 8bf4b1621178 ("drm/omap: Remove panel-dpi driver")
```

```
    Signed-off-by: Adam Ford <aford173@gmail.com>
    Signed-off-by: Tony Lindgren <tony@atomide.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e0986a44738ed0ebffed294def1233ff104e1f90
Author: Adam Ford <aford173@gmail.com>
Date:   Wed Aug 28 13:33:50 2019 -0500

    ARM: dts: logicpd-torpedo-baseboard: Fix missing video

    commit f9f5518a38684e031d913f40482721ff553f5ba2 upstream.

    A previous commit removed the panel-dpi driver, which made the
    Torpedo video stop working because it relied on the dpi driver
    for setting video timings.  Now that the simple-panel driver is
    available in omap2plus, this patch migrates the Torpedo dev kits
    to use a similar panel and remove the manual timing requirements.

    Fixes: 8bf4b1621178 ("drm/omap: Remove panel-dpi driver")

    Signed-off-by: Adam Ford <aford173@gmail.com>
    Signed-off-by: Tony Lindgren <tony@atomide.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b07c147e33938a298093279bd7c941917794f87a
Author: MyungJoo Ham <myungjoo.ham@samsung.com>
Date:   Mon Aug 26 21:37:37 2019 +0900

    PM / devfreq: passive: fix compiler warning

    [ Upstream commit 0465814831a926ce2f83e8f606d067d86745234e ]

    The recent commit of
    PM / devfreq: passive: Use non-devm notifiers
    had incurred compiler warning, "unused variable 'dev'".

    Reported-by: Stephen Rothwell <sfr@canb.auug.org.au>
    Signed-off-by: MyungJoo Ham <myungjoo.ham@samsung.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f21b06577c00f47755d863f80a6aae2adc1b779d
Author: Sakari Ailus <sakari.ailus@linux.intel.com>
Date:   Wed Aug 7 11:19:00 2019 -0300

    media: omap3isp: Set device on omap3isp subdevs

    [ Upstream commit e9eb103f027725053a4b02f93d7f2858b56747ce ]

    The omap3isp driver registered subdevs without the dev field being set. Do
    that now.

    Signed-off-by: Sakari Ailus <sakari.ailus@linux.intel.com>
    Reviewed-by: Laurent Pinchart <laurent.pinchart@ideasonboard.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dfe00eddab37232b614fc4bf790125f3551867de
Author: Jiří Paleček <jpalecek@web.de>
Date:   Sat Jun 22 19:42:04 2019 +0200

    kvm: Nested KVM MMUs need PAE root too

    [ Upstream commit 1cfff4d9a5d01fa61e5768a6afffc81ae1c8ecb9 ]
```

```
    On AMD processors, in PAE 32bit mode, nested KVM instances don't
    work. The L0 host get a kernel OOPS, which is related to
    arch.mmu->pae_root being NULL.

    The reason for this is that when setting up nested KVM instance,
    arch.mmu is set to &arch.guest_mmu (while normally, it would be
    &arch.root_mmu). However, the initialization and allocation of
    pae_root only creates it in root_mmu. KVM code (ie. in
    mmu_alloc_shadow_roots) then accesses arch.mmu->pae_root, which is the
    unallocated arch.guest_mmu->pae_root.

    This fix just allocates (and frees) pae_root in both guest_mmu and
    root_mmu (and also lm_root if it was allocated). The allocation is
    subject to previous restrictions ie. it won't allocate anything on
    64-bit and AFAIK not on Intel.

    Fixes: https://bugzilla.kernel.org/show_bug.cgi?id=203923
    Fixes: 14c07ad89f4d ("x86/kvm/mmu: introduce guest_mmu")
    Signed-off-by: Jiri Palecek <jpalecek@web.de>
    Tested-by: Jiri Palecek <jpalecek@web.de>
    Signed-off-by: Paolo Bonzini <pbonzini@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b01d7910727edb3a116d182151a319e453082a11
Author: Qu Wenruo <wqu@suse.com>
Date:   Thu Aug 22 10:14:15 2019 +0800

    btrfs: Detect unbalanced tree with empty leaf before crashing btree operations

    [ Upstream commit 62fdaa52a3d00a875da771719b6dc537ca79fce1 ]

    [BUG]
    With crafted image, btrfs will panic at btree operations:

      kernel BUG at fs/btrfs/ctree.c:3894!
      invalid opcode: 0000 [#1] SMP PTI
      CPU: 0 PID: 1138 Comm: btrfs-transacti Not tainted 5.0.0-rc8+ #9
      RIP: 0010:__push_leaf_left+0x6b6/0x6e0
      RSP: 0018:ffffc0bd4128b990 EFLAGS: 00010246
      RAX: 0000000000000000 RBX: ffffa0a4ab8f0e38 RCX: 0000000000000000
      RDX: ffffa0a280000000 RSI: 0000000000000000 RDI: ffffa0a4b3814000
      RBP: ffffc0bd4128ba38 R08: 0000000000001000 R09: ffffc0bd4128b948
      R10: 0000000000000000 R11: 0000000000000000 R12: 0000000000000240
      R13: ffffa0a4b556fb60 R14: ffffa0a4ab8f0af0 R15: ffffa0a4ab8f0af0
      FS: 0000000000000000(0000) GS:ffffa0a4b7a00000(0000) knlGS:0000000000000000
      CS: 0010 DS: 0000 ES: 0000 CR0: 0000000080050033
      CR2: 00007f2461c80020 CR3: 000000022b32a006 CR4: 00000000000206f0
      Call Trace:
      ? _cond_resched+0x1a/0x50
      push_leaf_left+0x179/0x190
      btrfs_del_items+0x316/0x470
      btrfs_del_csums+0x215/0x3a0
      __btrfs_free_extent.isra.72+0x5a7/0xbe0
      __btrfs_run_delayed_refs+0x539/0x1120
      btrfs_run_delayed_refs+0xdb/0x1b0
      btrfs_commit_transaction+0x52/0x950
      ? start_transaction+0x94/0x450
      transaction_kthread+0x163/0x190
      kthread+0x105/0x140
      ? btrfs_cleanup_transaction+0x560/0x560
      ? kthread_destroy_worker+0x50/0x50
      ret_from_fork+0x35/0x40
      Modules linked in:
      ---[ end trace c2425e6e89b5558f ]---
```

```
    [CAUSE]
    The offending csum tree looks like this:

      checksum tree key (CSUM_TREE ROOT_ITEM 0)
      node 29741056 level 1 items 14 free 107 generation 19 owner CSUM_TREE
              ...
              key (EXTENT_CSUM EXTENT_CSUM 85975040) block 29630464 gen 17
              key (EXTENT_CSUM EXTENT_CSUM 89911296) block 29642752 gen 17 <<<
              key (EXTENT_CSUM EXTENT_CSUM 92274688) block 29646848 gen 17
              ...

      leaf 29630464 items 6 free space 1 generation 17 owner CSUM_TREE
              item 0 key (EXTENT_CSUM EXTENT_CSUM 85975040) itemoff 3987 itemsize 8
                      range start 85975040 end 85983232 length 8192
              ...
      leaf 29642752 items 0 free space 3995 generation 17 owner 0
                           ^ empty leaf            invalid owner ^

      leaf 29646848 items 1 free space 602 generation 17 owner CSUM_TREE
              item 0 key (EXTENT_CSUM EXTENT_CSUM 92274688) itemoff 627 itemsize 3368
                      range start 92274688 end 95723520 length 3448832

    So we have a corrupted csum tree where one tree leaf is completely
    empty, causing unbalanced btree, thus leading to unexpected btree
    balance error.

    [FIX]
    For this particular case, we handle it in two directions to catch it:
    - Check if the tree block is empty through btrfs_verify_level_key()
      So that invalid tree blocks won't be read out through
      btrfs_search_slot() and its variants.

    - Check 0 tree owner in tree checker
      NO tree is using 0 as its tree owner, detect it and reject at tree
      block read time.

    Bugzilla: https://bugzilla.kernel.org/show_bug.cgi?id=202821
    Reviewed-by: Nikolay Borisov <nborisov@suse.com>
    Signed-off-by: Qu Wenruo <wqu@suse.com>
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit fba904d68c8bbfcc89c2210cfcb2351be90bc3e5
Author: Qu Wenruo <wqu@suse.com>
Date:   Tue Jul 16 17:00:34 2019 +0800

    btrfs: tree-checker: Add ROOT_ITEM check

    [ Upstream commit 259ee7754b6793af8bdd77f9ca818bc41cfe9541 ]

    This patch will introduce ROOT_ITEM check, which includes:
    - Key->objectid and key->offset check
      Currently only some easy check, e.g. 0 as rootid is invalid.

    - Item size check
      Root item size is fixed.

    - Generation checks
      Generation, generation_v2 and last_snapshot should not be greater than
      super generation + 1

    - Level and alignment check
      Level should be in [0, 7], and bytenr must be aligned to sector size.

    - Flags check
```

```
    Bugzilla: https://bugzilla.kernel.org/show_bug.cgi?id=203261
    Reported-by: Jungyeon Yoon <jungyeon.yoon@gmail.com>
    Signed-off-by: Qu Wenruo <wqu@suse.com>
    Reviewed-by: David Sterba <dsterba@suse.com>
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c8af37dc7509c9ca673620be9b52fa68ebbfd6da
Author: Qu Wenruo <wqu@suse.com>
Date:   Tue Jul 16 17:00:33 2019 +0800

    btrfs: extent-tree: Make sure we only allocate extents from block groups with the same
type

    [ Upstream commit 2a28468e525f3924efed7f29f2bc5a2926e7e19a ]

    [BUG]
    With fuzzed image and MIXED_GROUPS super flag, we can hit the following
    BUG_ON():

      kernel BUG at fs/btrfs/delayed-ref.c:491!
      invalid opcode: 0000 [#1] PREEMPT SMP NOPTI
      CPU: 0 PID: 1849 Comm: sync Tainted: G           O      5.2.0-custom #27
      RIP: 0010:update_existing_head_ref.cold+0x44/0x46 [btrfs]
      Call Trace:
       add_delayed_ref_head+0x20c/0x2d0 [btrfs]
       btrfs_add_delayed_tree_ref+0x1fc/0x490 [btrfs]
       btrfs_free_tree_block+0x123/0x380 [btrfs]
       __btrfs_cow_block+0x435/0x500 [btrfs]
       btrfs_cow_block+0x110/0x240 [btrfs]
       btrfs_search_slot+0x230/0xa00 [btrfs]
       ? __lock_acquire+0x105e/0x1e20
       btrfs_insert_empty_items+0x67/0xc0 [btrfs]
       alloc_reserved_file_extent+0x9e/0x340 [btrfs]
       __btrfs_run_delayed_refs+0x78e/0x1240 [btrfs]
       ? kvm_clock_read+0x18/0x30
       ? __sched_clock_gtod_offset+0x21/0x50
       btrfs_run_delayed_refs.part.0+0x4e/0x180 [btrfs]
       btrfs_run_delayed_refs+0x23/0x30 [btrfs]
       btrfs_commit_transaction+0x53/0x9f0 [btrfs]
       btrfs_sync_fs+0x7c/0x1c0 [btrfs]
       ? __ia32_sys_fdatasync+0x20/0x20
       sync_fs_one_sb+0x23/0x30
       iterate_supers+0x95/0x100
       ksys_sync+0x62/0xb0
       __ia32_sys_sync+0xe/0x20
       do_syscall_64+0x65/0x240
       entry_SYSCALL_64_after_hwframe+0x49/0xbe

    [CAUSE]
    This situation is caused by several factors:
    - Fuzzed image
      The extent tree of this fs missed one backref for extent tree root.
      So we can allocated space from that slot.

    - MIXED_BG feature
      Super block has MIXED_BG flag.

    - No mixed block groups exists
      All block groups are just regular ones.

    This makes data space_info->block_groups[] contains metadata block
    groups.  And when we reserve space for data, we can use space in
    metadata block group.
```

```
    Then we hit the following file operations:

    - fallocate
      We need to allocate data extents.
      find_free_extent() choose to use the metadata block to allocate space
      from, and choose the space of extent tree root, since its backref is
      missing.

      This generate one delayed ref head with is_data = 1.

    - extent tree update
      We need to update extent tree at run_delayed_ref time.

      This generate one delayed ref head with is_data = 0, for the same
      bytenr of old extent tree root.

    Then we trigger the BUG_ON().

    [FIX]
    The quick fix here is to check block_group->flags before using it.

    The problem can only happen for MIXED_GROUPS fs. Regular filesystems
    won't have space_info with DATA|METADATA flag, and no way to hit the
    bug.

    Bugzilla: https://bugzilla.kernel.org/show_bug.cgi?id=203255
    Reported-by: Jungyeon Yoon <jungyeon.yoon@gmail.com>
    Signed-off-by: Qu Wenruo <wqu@suse.com>
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ca6224f72793f674b13b672e277cd3c5eee68d09
Author: Qu Wenruo <wqu@suse.com>
Date:   Tue Jul 16 17:00:32 2019 +0800

    btrfs: delayed-inode: Kill the BUG_ON() in btrfs_delete_delayed_dir_index()

    [ Upstream commit 933c22a7512c5c09b1fdc46b557384efe8d03233 ]

    There is one report of fuzzed image which leads to BUG_ON() in
    btrfs_delete_delayed_dir_index().

    Although that fuzzed image can already be addressed by enhanced
    extent-tree error handler, it's still better to hunt down more BUG_ON().

    This patch will hunt down two BUG_ON()s in
    btrfs_delete_delayed_dir_index():
    - One for error from btrfs_delayed_item_reserve_metadata()
      Instead of BUG_ON(), we output an error message and free the item.
      And return the error.
      All callers of this function handles the error by aborting current
      trasaction.

    - One for possible EEXIST from __btrfs_add_delayed_deletion_item()
      That function can return -EEXIST.
      We already have a good enough error message for that, only need to
      clean up the reserved metadata space and allocated item.

    To help above cleanup, also modifiy __btrfs_remove_delayed_item() called
    in btrfs_release_delayed_item(), to skip unassociated item.

    Bugzilla: https://bugzilla.kernel.org/show_bug.cgi?id=203253
    Signed-off-by: Qu Wenruo <wqu@suse.com>
    Reviewed-by: David Sterba <dsterba@suse.com>
```

```
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2506669d27b007dad309b308cb42c02234644d70
Author: Oliver Neukum <oneukum@suse.com>
Date:   Tue Aug 13 14:04:11 2019 +0200

    zd1211rw: remove false assertion from zd_mac_clear()

    [ Upstream commit 7a2eb7367fdea72e448d1a847aa857f6caf8ea2f ]

    The function is called before the lock which is asserted was ever used.
    Just remove it.

    Reported-by: syzbot+74c65761783d66a9c97c@syzkaller.appspotmail.com
    Signed-off-by: Oliver Neukum <oneukum@suse.com>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d3500d29908c78a2c6a86b0a45c8a2bb2e424d5e
Author: Kai-Heng Feng <kai.heng.feng@canonical.com>
Date:   Wed Aug 21 13:10:04 2019 +0800

    iommu/amd: Override wrong IVRS IOAPIC on Raven Ridge systems

    [ Upstream commit 93d051550ee02eaff9a2541d825605a7bd778027 ]

    Raven Ridge systems may have malfunction touchpad or hang at boot if
    incorrect IVRS IOAPIC is provided by BIOS.

    Users already found correct "ivrs_ioapic=" values, let's put them inside
    kernel to workaround buggy BIOS.

    BugLink: https://bugs.launchpad.net/bugs/1795292
    BugLink: https://bugs.launchpad.net/bugs/1837688
    Reported-by: kbuild test robot <lkp@intel.com>
    Signed-off-by: Kai-Heng Feng <kai.heng.feng@canonical.com>
    Signed-off-by: Joerg Roedel <jroedel@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit bd25afb31eda00e0426cf003e226b4b4b18a1e58
Author: Takashi Iwai <tiwai@suse.de>
Date:   Thu Aug 22 09:58:07 2019 +0200

    ALSA: hda/realtek - Blacklist PC beep for Lenovo ThinkCentre M73/93

    [ Upstream commit 051c78af14fcd74a22b5af45548ad9d588247cc7 ]

    Lenovo ThinkCentre M73 and M93 don't seem to have a proper beep
    although the driver tries to probe and set up blindly.
    Blacklist these machines for suppressing the beep creation.

    BugLink: https://bugzilla.kernel.org/show_bug.cgi?id=204635
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 97f86e3de74686fe5cc5ccd6f46fbd5b953373e9
Author: Jani Nikula <jani.nikula@intel.com>
Date:   Wed Aug 21 12:43:12 2019 +0300

    drm: fix module name in edid_firmware log message

    [ Upstream commit ade925995b172f1d7410d1c665b2f47c5e99bef0 ]

    The module is drm_kms_helper, not drm_kms_firmware.
```

```
    Bugzilla: https://bugzilla.kernel.org/show_bug.cgi?id=204549
    Reported-by: Göran Uddeborg <goeran@uddeborg.se>
    Fixes: ac6c35a4d8c7 ("drm: add backwards compatibility support for
drm_kms_helper.edid_firmware")
    Reviewed-by: Daniel Vetter <daniel.vetter@ffwll.ch>
    Signed-off-by: Jani Nikula <jani.nikula@intel.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20190821094312.5514-1-
jani.nikula@intel.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2ecab41cf6e9774be08242f808ec3f5d07372ead
Author: Tomas Bortoli <tomasbortoli@gmail.com>
Date:   Wed Jul 31 12:19:05 2019 -0300

    media: ttusb-dec: Fix info-leak in ttusb_dec_send_command()

    [ Upstream commit a10feaf8c464c3f9cfdd3a8a7ce17e1c0d498da1 ]

    The function at issue does not always initialize each byte allocated
    for 'b' and can therefore leak uninitialized memory to a USB device in
    the call to usb_bulk_msg()

    Use kzalloc() instead of kmalloc()

    Signed-off-by: Tomas Bortoli <tomasbortoli@gmail.com>
    Reported-by: syzbot+0522702e9d67142379f1@syzkaller.appspotmail.com
    Signed-off-by: Sean Young <sean@mess.org>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c748297e8576cce33a9ddb9e3b9ccf4b2f11e290
Author: Ahzo <Ahzo@tutanota.com>
Date:   Mon Aug 5 21:14:18 2019 +0200

    drm/amd/powerplay/smu7: enforce minimal VBITimeout (v2)

    [ Upstream commit f659bb6dae58c113805f92822e4c16ddd3156b79 ]

    This fixes screen corruption/flickering on 75 Hz displays.

    v2: make print statement debug only (Alex)

    Bugzilla: https://bugs.freedesktop.org/show_bug.cgi?id=102646
    Reviewed-by: Evan Quan <evan.quan@amd.com>
    Signed-off-by: Ahzo <Ahzo@tutanota.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 911bc1bf59351caf8371782a86b2258c81288d88
Author: Takashi Iwai <tiwai@suse.de>
Date:   Tue Aug 13 17:11:28 2019 +0200

    ALSA: hda - Drop unsol event handler for Intel HDMI codecs

    [ Upstream commit f2dbe87c5ac1f88e6007ba1f1374f4bd8a197fb6 ]

    We don't need to deal with the unsol events for Intel chips that are
    tied with the graphics via audio component notifier.  Although the
    presence of the audio component is checked at the beginning of
    hdmi_unsol_event(), better to short cut by dropping unsol_event ops.

    BugLink: https://bugzilla.kernel.org/show_bug.cgi?id=204565
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit c620c6d85168220922d44046164c4d0baa0bfc19
Author: Tomas Espeleta <tomas.espeleta@gmail.com>
Date:   Fri Aug 9 16:37:54 2019 +0200

    ALSA: hda - Add a quirk model for fixing Huawei Matebook X right speaker

    [ Upstream commit a2ef03fe617a8365fb7794531b11ba587509a9b9 ]

    [ This is rather a revival of the patch Tomas sent in months ago, but
      applying only with the quirk model option -- tiwai ]

    Hard coded coefficients to make Huawuei Matebook X right speaker
    work. The Matebook X has a ALC298, please refer to bug 197801 on
    how these numbers were reverse engineered from the Windows driver

    The reversed engineered sequence represents a repeating pattern
    of verbs, and the only values that are changing periodically are
    written on indexes 0x23 and 0x25:

    0x500, 0x23
    0x400, VALUE1
    0x500, 0x25
    0x400, VALUE2

    * skipped reading sequences (0x500 - 0xc00 sequences are ignored)
    * static values from reverse engineering are used

    NOTE: since a significant risk is still considered, this is provided
    as an experimental fix that isn't applied as default for now.  For
    enabling the fix, you'll have to choose huawei-mbx-stereo via model
    option of snd-hda-intel module.

    If we get feedback from users that this works stably, we may apply it
    per default.

    [ Some coding style fixes and replacement with AC_VERB_* by tiwai ]

    BugLink: https://bugzilla.kernel.org/show_bug.cgi?id=197801
    Signed-off-by: Tomas Espeleta <tomas.espeleta@gmail.com>
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 10964e0af3ca7b7302456dc61d49c969a5c93cab
Author: Kai-Heng Feng <kai.heng.feng@canonical.com>
Date:   Mon Jul 8 12:55:45 2019 +0800

    e1000e: add workaround for possible stalled packet

    [ Upstream commit e5e9a2ecfe780975820e157b922edee715710b66 ]

    This works around a possible stalled packet issue, which may occur due to
    clock recovery from the PCH being too slow, when the LAN is transitioning
    from K1 at 1G link speed.

    Bugzilla: https://bugzilla.kernel.org/show_bug.cgi?id=204057

    Signed-off-by: Kai-Heng Feng <kai.heng.feng@canonical.com>
    Tested-by: Aaron Brown <aaron.f.brown@intel.com>
    Signed-off-by: Jeff Kirsher <jeffrey.t.kirsher@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7bb111f9e7cf968beb200f317890da6facd7c0a0
Author: Kevin Easton <kevin@guarana.org>
Date:   Wed Jul 10 13:31:38 2019 +0000
```

```
    libertas: Add missing sentinel at end of if_usb.c fw_table

    [ Upstream commit 764f3f1ecffc434096e0a2b02f1a6cc964a89df6 ]

    This sentinel tells the firmware loading process when to stop.

    Reported-and-tested-by: syzbot+98156c174c5a2cad9f8f@syzkaller.appspotmail.com
    Signed-off-by: Kevin Easton <kevin@guarana.org>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit aaa9678faffab2db6d33654852862f72924d667c
Author: Ulf Hansson <ulf.hansson@linaro.org>
Date:   Wed Sep 11 14:09:20 2019 +0200

    mmc: mtk-sd: Re-store SDIO IRQs mask at system resume

    [ Upstream commit 1c81d69d4c98aab56c5a7d5a810f84aefdb37e9b ]

    In cases when SDIO IRQs have been enabled, runtime suspend is prevented by
    the driver. However, this still means msdc_runtime_suspend|resume() gets
    called during system suspend/resume, via pm_runtime_force_suspend|resume().

    This means during system suspend/resume, the register context of the mtk-sd
    device most likely loses its register context, even in cases when SDIO IRQs
    have been enabled.

    To re-enable the SDIO IRQs during system resume, the mtk-sd driver
    currently relies on the mmc core to re-enable the SDIO IRQs when it resumes
    the SDIO card, but this isn't the recommended solution. Instead, it's
    better to deal with this locally in the mtk-sd driver, so let's do that.

    Signed-off-by: Ulf Hansson <ulf.hansson@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 10800152ee496823174211ff13da7721aadf2895
Author: Nigel Croxon <ncroxon@redhat.com>
Date:   Fri Sep 6 09:21:33 2019 -0400

    raid5: don't increment read_errors on EILSEQ return

    [ Upstream commit b76b4715eba0d0ed574f58918b29c1b2f0fa37a8 ]

    While MD continues to count read errors returned by the lower layer.
    If those errors are -EILSEQ, instead of -EIO, it should NOT increase
    the read_errors count.

    When RAID6 is set up on dm-integrity target that detects massive
    corruption, the leg will be ejected from the array.  Even if the
    issue is correctable with a sector re-write and the array has
    necessary redundancy to correct it.

    The leg is ejected because it runs up the rdev->read_errors beyond
    conf->max_nr_stripes.  The return status in dm-drypt when there is
    a data integrity error is -EILSEQ (BLK_STS_PROTECTION).

    Signed-off-by: Nigel Croxon <ncroxon@redhat.com>
    Signed-off-by: Song Liu <songliubraving@fb.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2ba27b2481614df1275def84574442436329aaf2
Author: Ulf Hansson <ulf.hansson@linaro.org>
Date:   Sun Sep 8 12:12:27 2019 +0200
```

```
    mmc: dw_mmc: Re-store SDIO IRQs mask at system resume

    [ Upstream commit 7c526608d5afb62cbc967225e2ccaacfdd142e9d ]

    In cases when SDIO IRQs have been enabled, runtime suspend is prevented by
    the driver. However, this still means dw_mci_runtime_suspend|resume() gets
    called during system suspend/resume, via pm_runtime_force_suspend|resume().
    This means during system suspend/resume, the register context of the dw_mmc
    device most likely loses its register context, even in cases when SDIO IRQs
    have been enabled.

    To re-enable the SDIO IRQs during system resume, the dw_mmc driver
    currently relies on the mmc core to re-enable the SDIO IRQs when it resumes
    the SDIO card, but this isn't the recommended solution. Instead, it's
    better to deal with this locally in the dw_mmc driver, so let's do that.

    Tested-by: Matthias Kaehlcke <mka@chromium.org>
    Signed-off-by: Ulf Hansson <ulf.hansson@linaro.org>
    Reviewed-by: Douglas Anderson <dianders@chromium.org>
    Signed-off-by: Ulf Hansson <ulf.hansson@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 06eb2cf6d5a02b1a277ff4ea3ebff5ee7664c549
Author: Ulf Hansson <ulf.hansson@linaro.org>
Date:   Sun Sep 8 12:12:26 2019 +0200

    mmc: core: Add helper function to indicate if SDIO IRQs is enabled

    [ Upstream commit bd880b00697befb73eff7220ee20bdae4fdd487b ]

    To avoid each host driver supporting SDIO IRQs, from keeping track
    internally about if SDIO IRQs has been claimed, let's introduce a common
    helper function, sdio_irq_claimed().

    The function returns true if SDIO IRQs are claimed, via using the
    information about the number of claimed irqs. This is safe, even without
    any locks, as long as the helper function is called only from
    runtime/system suspend callbacks of the host driver.

    Tested-by: Matthias Kaehlcke <mka@chromium.org>
    Signed-off-by: Ulf Hansson <ulf.hansson@linaro.org>
    Reviewed-by: Douglas Anderson <dianders@chromium.org>
    Signed-off-by: Ulf Hansson <ulf.hansson@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5dba7dcf90c702d5d19a0c53181ca285b14185cf
Author: Al Cooper <alcooperx@gmail.com>
Date:   Tue Sep 3 07:51:14 2019 -0400

    mmc: sdhci: Fix incorrect switch to HS mode

    [ Upstream commit c894e33ddc1910e14d6f2a2016f60ab613fd8b37 ]

    When switching from any MMC speed mode that requires 1.8v
    (HS200, HS400 and HS400ES) to High Speed (HS) mode, the system
    ends up configured for SDR12 with a 50MHz clock which is an illegal
    mode.

    This happens because the SDHCI_CTRL_VDD_180 bit in the
    SDHCI_HOST_CONTROL2 register is left set and when this bit is
    set, the speed mode is controlled by the SDHCI_CTRL_UHS field
    in the SDHCI_HOST_CONTROL2 register. The SDHCI_CTRL_UHS field
    will end up being set to 0 (SDR12) by sdhci_set_uhs_signaling()
    because there is no UHS mode being set.
```

```
    The fix is to change sdhci_set_uhs_signaling() to set the
    SDHCI_CTRL_UHS field to SDR25 (which is the same as HS) for
    any switch to HS mode.

    This was found on a new eMMC controller that does strict checking
    of the speed mode and the corresponding clock rate. It caused the
    switch to HS400 mode to fail because part of the sequence to switch
    to HS400 requires a switch from HS200 to HS before going to HS400.

    Suggested-by: Adrian Hunter <adrian.hunter@intel.com>
    Signed-off-by: Al Cooper <alcooperx@gmail.com>
    Signed-off-by: Ulf Hansson <ulf.hansson@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3d457d428aa97559ed756d88ef716e189537bfac
Author: Miles Chen <miles.chen@mediatek.com>
Date:   Thu Sep 12 18:34:52 2019 +0800

    sched/psi: Correct overly pessimistic size calculation

    [ Upstream commit 4adcdcea717cb2d8436bef00dd689aa5bc76f11b ]

    When passing a equal or more then 32 bytes long string to psi_write(),
    psi_write() copies 31 bytes to its buf and overwrites buf[30]
    with '\0'. Which makes the input string 1 byte shorter than
    it should be.

    Fix it by copying sizeof(buf) bytes when nbytes >= sizeof(buf).

    This does not cause problems in normal use case like:
    "some 500000 10000000" or "full 500000 10000000" because they
    are less than 32 bytes in length.

            /* assuming nbytes == 35 */
            char buf[32];

            buf_size = min(nbytes, (sizeof(buf) - 1)); /* buf_size = 31 */
            if (copy_from_user(buf, user_buf, buf_size))
                    return -EFAULT;

            buf[buf_size - 1] = '\0'; /* buf[30] = '\0' */

    Before:

     %cd /proc/pressure/
     %echo "123456789|123456789|123456789|1234" > memory
     [   22.473497] nbytes=35,buf_size=31
     [   22.473775] 123456789|123456789|123456789| (print 30 chars)
     %sh: write error: Invalid argument

     %echo "123456789|123456789|123456789|1" > memory
     [   64.916162] nbytes=32,buf_size=31
     [   64.916331] 123456789|123456789|123456789| (print 30 chars)
     %sh: write error: Invalid argument

    After:

     %cd /proc/pressure/
     %echo "123456789|123456789|123456789|1234" > memory
     [  254.837863] nbytes=35,buf_size=32
     [  254.838541] 123456789|123456789|123456789|1 (print 31 chars)
     %sh: write error: Invalid argument

     %echo "123456789|123456789|123456789|1" > memory
     [ 9965.714935] nbytes=32,buf_size=32
```

```
      [ 9965.715096] 123456789|123456789|123456789|1 (print 31 chars)
      %sh: write error: Invalid argument

    Also remove the superfluous parentheses.

    Signed-off-by: Miles Chen <miles.chen@mediatek.com>
    Cc: <linux-mediatek@lists.infradead.org>
    Cc: <wsd_upstream@mediatek.com>
    Cc: Linus Torvalds <torvalds@linux-foundation.org>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Cc: Thomas Gleixner <tglx@linutronix.de>
    Link: https://lkml.kernel.org/r/20190912103452.13281-1-miles.chen@mediatek.com
    Signed-off-by: Ingo Molnar <mingo@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6b8339401fc1a8840afe9b5f11c3bfb29e546958
Author: Ulf Hansson <ulf.hansson@linaro.org>
Date:   Sun Sep 8 12:12:30 2019 +0200

    mmc: core: Clarify sdio_irq_pending flag for MMC_CAP2_SDIO_IRQ_NOTHREAD

    [ Upstream commit 36d57efb4af534dd6b442ea0b9a04aa6dfa37abe ]

    The sdio_irq_pending flag is used to let host drivers indicate that it has
    signaled an IRQ. If that is the case and we only have a single SDIO func
    that have claimed an SDIO IRQ, our assumption is that we can avoid reading
    the SDIO_CCCR_INTx register and just call the SDIO func irq handler
    immediately. This makes sense, but the flag is set/cleared in a somewhat
    messy order, let's fix that up according to below.

    First, the flag is currently set in sdio_run_irqs(), which is executed as a
    work that was scheduled from sdio_signal_irq(). To make it more implicit
    that the host have signaled an IRQ, let's instead immediately set the flag
    in sdio_signal_irq(). This also makes the behavior consistent with host
    drivers that uses the legacy, mmc_signal_sdio_irq() API. This have no
    functional impact, because we don't expect host drivers to call
    sdio_signal_irq() until after the work (sdio_run_irqs()) have been executed
    anyways.

    Second, currently we never clears the flag when using the sdio_run_irqs()
    work, but only when using the sdio_irq_thread(). Let make the behavior
    consistent, by moving the flag to be cleared inside the common
    process_sdio_pending_irqs() function. Additionally, tweak the behavior of
    the flag slightly, by avoiding to clear it unless we processed the SDIO
    IRQ. The purpose with this at this point, is to keep the information about
    whether there have been an SDIO IRQ signaled by the host, so at system
    resume we can decide to process it without reading the SDIO_CCCR_INTx
    register.

    Tested-by: Matthias Kaehlcke <mka@chromium.org>
    Reviewed-by: Matthias Kaehlcke <mka@chromium.org>
    Signed-off-by: Ulf Hansson <ulf.hansson@linaro.org>
    Reviewed-by: Douglas Anderson <dianders@chromium.org>
    Signed-off-by: Ulf Hansson <ulf.hansson@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8033213bae2d924ea6cc09641d23b8820daa3f69
Author: Guoqing Jiang <guoqing.jiang@cloud.ionos.com>
Date:   Wed Sep 11 10:06:29 2019 +0200

    raid5: don't set STRIPE_HANDLE to stripe which is in batch list

    [ Upstream commit 6ce220dd2f8ea71d6afc29b9a7524c12e39f374a ]

    If stripe in batch list is set with STRIPE_HANDLE flag, then the stripe
```

```
    could be set with STRIPE_ACTIVE by the handle_stripe function. And if
    error happens to the batch_head at the same time, break_stripe_batch_list
    is called, then below warning could happen (the same report in [1]), it
    means a member of batch list was set with STRIPE_ACTIVE.

    [7028915.431770] stripe state: 2001
    [7028915.431815] ------------[ cut here ]------------
    [7028915.431828] WARNING: CPU: 18 PID: 29089 at drivers/md/raid5.c:4614
 break_stripe_batch_list+0x203/0x240 [raid456]
    [...]
    [7028915.431879] CPU: 18 PID: 29089 Comm: kworker/u82:5 Tainted: G          O
 4.14.86-1-storage #4.14.86-1.2~deb9
    [7028915.431881] Hardware name: Supermicro SSG-2028R-ACR24L/X10DRH-iT, BIOS 3.1
 06/18/2018
    [7028915.431888] Workqueue: raid5wq raid5_do_work [raid456]
    [7028915.431890] task: ffff9ab0ef36d7c0 task.stack: ffffb72926f84000
    [7028915.431896] RIP: 0010:break_stripe_batch_list+0x203/0x240 [raid456]
    [7028915.431898] RSP: 0018:ffffb72926f87ba8 EFLAGS: 00010286
    [7028915.431900] RAX: 0000000000000012 RBX: ffff9aaa84a98000 RCX: 0000000000000000
    [7028915.431901] RDX: 0000000000000000 RSI: ffff9ab2bfa15458 RDI: ffff9ab2bfa15458
    [7028915.431902] RBP: ffff9aaa8fb4e900 R08: 0000000000000001 R09: 0000000000002eb4
    [7028915.431903] R10: 00000000ffffffff R11: 0000000000000000 R12: ffff9ab1736f1b00
    [7028915.431904] R13: 0000000000000000 R14: ffff9aaa8fb4e900 R15: 0000000000000001
    [7028915.431906] FS:  0000000000000000(0000) GS:ffff9ab2bfa00000(0000)
 knlGS:0000000000000000
    [7028915.431907] CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    [7028915.431908] CR2: 00007ff953b9f5d8 CR3: 0000000bf4009002 CR4: 00000000003606e0
    [7028915.431909] DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000
    [7028915.431910] DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400
    [7028915.431910] Call Trace:
    [7028915.431923]  handle_stripe+0x8e7/0x2020 [raid456]
    [7028915.431930]  ? __wake_up_common_lock+0x89/0xc0
    [7028915.431935]  handle_active_stripes.isra.58+0x35f/0x560 [raid456]
    [7028915.431939]  raid5_do_work+0xc6/0x1f0 [raid456]

    Also commit 59fc630b8b5f9f ("RAID5: batch adjacent full stripe write")
    said "If a stripe is added to batch list, then only the first stripe
    of the list should be put to handle_list and run handle_stripe."

    So don't set STRIPE_HANDLE to stripe which is already in batch list,
    otherwise the stripe could be put to handle_list and run handle_stripe,
    then the above warning could be triggered.

    [1]. https://www.spinics.net/lists/raid/msg62552.html

    Signed-off-by: Guoqing Jiang <guoqing.jiang@cloud.ionos.com>
    Signed-off-by: Song Liu <songliubraving@fb.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit 209c1af1ea3ed83d63f6b7ad8ff34e278248de48
 Author: Hou Tao <houtao1@huawei.com>
 Date:   Tue May 21 15:59:03 2019 +0800

    block: make rq sector size accessible for block stats

    [ Upstream commit 3d24430694077313c75c6b89f618db09943621e4 ]

    Currently rq->data_len will be decreased by partial completion or
    zeroed by completion, so when blk_stat_add() is invoked, data_len
    will be zero and there will never be samples in poll_cb because
    blk_mq_poll_stats_bkt() will return -1 if data_len is zero.

    We could move blk_stat_add() back to __blk_mq_complete_request(),
    but that would make the effort of trying to call ktime_get_ns()
    once in vain. Instead we can reuse throtl_size field, and use
```

```
    it for both block stats and block throttle, and adjust the
    logic in blk_mq_poll_stats_bkt() accordingly.

    Fixes: 4bc6339a583c ("block: move blk_stat_add() to __blk_mq_end_request()")
    Tested-by: Pavel Begunkov <asml.silence@gmail.com>
    Signed-off-by: Hou Tao <houtao1@huawei.com>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3f52b1251503c92d7f382d350202d7dcad86fc6c
Author: Jackie Liu <liuyun01@kylinos.cn>
Date:   Mon Sep 9 20:50:39 2019 +0800

    io_uring: fix wrong sequence setting logic

    [ Upstream commit 8776f3fa15a5cd213c4dfab7ddaf557983374ea6 ]

    Sqo_thread will get sqring in batches, which will cause
    ctx->cached_sq_head to be added in batches. if one of these
    sqes is set with the DRAIN flag, then he will never get a
    chance to process, and finally sqo_thread will not exit.

    Fixes: de0617e4671 ("io_uring: add support for marking commands as draining")
    Signed-off-by: Jackie Liu <liuyun01@kylinos.cn>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 664acfa7199a4625627faee9aa6ddde20a23ab35
Author: Lukas Wunner <lukas@wunner.de>
Date:   Sat Aug 3 12:10:00 2019 +0200

    spi: bcm2835: Work around DONE bit erratum

    [ Upstream commit 4c524191c0a21d758b519087c64f84348095e940 ]

    Commit 3bd7f6589f67 ("spi: bcm2835: Overcome sglist entry length
    limitation") amended the BCM2835 SPI driver with support for DMA
    transfers whose buffers are not aligned to 4 bytes and require more than
    one sglist entry.

    When testing this feature with upcoming commits to speed up TX-only and
    RX-only transfers, I noticed that SPI transmission sometimes breaks.
    A function introduced by the commit, bcm2835_spi_transfer_prologue(),
    performs one or two PIO transmissions as a prologue to the actual DMA
    transmission.  It turns out that the breakage goes away if the DONE bit
    in the CS register is set when ending such a PIO transmission.

    The DONE bit signifies emptiness of the TX FIFO.  According to the spec,
    the bit is of type RO, so writing it should never have any effect.
    Perhaps the spec is wrong and the bit is actually of type RW1C.
    E.g. the I2C controller on the BCM2835 does have an RW1C DONE bit which
    needs to be cleared by the driver.  Another, possibly more likely
    explanation is that it's a hardware erratum since the issue does not
    occur consistently.

    Either way, amend bcm2835_spi_transfer_prologue() to always write the
    DONE bit.

    Usually a transmission is ended by bcm2835_spi_reset_hw().  If the
    transmission was successful, the TX FIFO is empty and thus the DONE bit
    is set when bcm2835_spi_reset_hw() reads the CS register.  The bit is
    then written back to the register, so we happen to do the right thing.

    However if DONE is not set, e.g. because transmission is aborted with
    a non-empty TX FIFO, the bit won't be written by bcm2835_spi_reset_hw()
```

```
    and it seems possible that transmission might subsequently break.  To be
    on the safe side, likewise amend bcm2835_spi_reset_hw() to always write
    the bit.

    Tested-by: Nuno Sá <nuno.sa@analog.com>
    Signed-off-by: Lukas Wunner <lukas@wunner.de>
    Acked-by: Stefan Wahren <wahrenst@gmx.net>
    Acked-by: Martin Sperl <kernel@martin.sperl.org>
    Link:
https://lore.kernel.org/r/edb004dff4af6106f6bfcb89e1a96391e96eb857.1564825752.git.lukas@wunn
er.de
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cd7a1295ab69e3ed3f7b858f8a12f218106e1e31
Author: Prarit Bhargava <prarit@redhat.com>
Date:   Thu Sep 5 08:03:11 2019 -0400

    tools/power/x86/intel-speed-select: Fix memory leak

    [ Upstream commit 3bc3d30ca324bfc3045a1a7fe1f5fe5ad5d92fd9 ]

    cpumasks are allocated by calling the alloc_cpu_mask() function and are
    never free'd.  They should be free'd after the commands have run.

    Fix the memory leaks by calling free_cpu_set().

    Signed-off-by: Prarit Bhargava <prarit@redhat.com>
    Acked-by: Srinivas Pandruvada <srinivas.pandruvada@linux.intel.com>
    Cc: Srinivas Pandruvada <srinivas.pandruvada@linux.intel.com>
    Cc: David Arcari <darcari@redhat.com>
    Cc: linux-kernel@vger.kernel.org
    Signed-off-by: Andy Shevchenko <andriy.shevchenko@linux.intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7407d0a392072fa915f46d313a0a7c7f362a6e1d
Author: Peter Ujfalusi <peter.ujfalusi@ti.com>
Date:   Fri Sep 6 08:55:24 2019 +0300

    ASoC: dmaengine: Make the pcm->name equal to pcm->id if the name is not set

    [ Upstream commit 2ec42f3147e1610716f184b02e65d7f493eed925 ]

    Some tools use the snd_pcm_info_get_name() to try to identify PCMs or for
    other purposes.

    Currently it is left empty with the dmaengine-pcm, in this case copy the
    pcm->id string as pcm->name.

    For example IGT is using this to find the HDMI PCM for testing audio on it.

    Signed-off-by: Peter Ujfalusi <peter.ujfalusi@ti.com>
    Reported-by: Arthur She <arthur.she@linaro.org>
    Link: https://lore.kernel.org/r/20190906055524.7393-1-peter.ujfalusi@ti.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 005d9ec550d3bb284eda8b00111b7adc60dc5c77
Author: M. Vefa Bicakci <m.v.b@runbox.com>
Date:   Thu Aug 15 21:41:40 2019 -0400

    platform/x86: intel_pmc_core_pltdrv: Module removal warning fix

    [ Upstream commit 0b43e41e93815ecd9616759cf5d64d3a7be8e6fb ]
```

```
    Prior to this commit, removing the intel_pmc_core_pltdrv module
    would cause the following warning:

      Device 'intel_pmc_core.0' does not have a release() function, it is broken and must be
fixed. See Documentation/kobject.txt.
      WARNING: CPU: 0 PID: 2202 at drivers/base/core.c:1238 device_release+0x6f/0x80

    This commit hence adds an empty release function for the driver.

    Signed-off-by: M. Vefa Bicakci <m.v.b@runbox.com>
    Signed-off-by: Andy Shevchenko <andriy.shevchenko@linux.intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dba6520552f4c57c2aa272dd653074b4b50be8a4
Author: M. Vefa Bicakci <m.v.b@runbox.com>
Date:   Thu Aug 15 21:41:39 2019 -0400

    platform/x86: intel_pmc_core: Do not ioremap RAM

    [ Upstream commit 7d505758b1e556cdf65a5e451744fe0ae8063d17 ]

    On a Xen-based PVH virtual machine with more than 4 GiB of RAM,
    intel_pmc_core fails initialization with the following warning message
    from the kernel, indicating that the driver is attempting to ioremap
    RAM:

      ioremap on RAM at 0x00000000fe000000 - 0x00000000fe001fff
      WARNING: CPU: 1 PID: 434 at arch/x86/mm/ioremap.c:186
__ioremap_caller.constprop.0+0x2aa/0x2c0
    ...
      Call Trace:
       ? pmc_core_probe+0x87/0x2d0 [intel_pmc_core]
       pmc_core_probe+0x87/0x2d0 [intel_pmc_core]

    This issue appears to manifest itself because of the following fallback
    mechanism in the driver:

            if (lpit_read_residency_count_address(&slp_s0_addr))
                    pmcdev->base_addr = PMC_BASE_ADDR_DEFAULT;

    The validity of address PMC_BASE_ADDR_DEFAULT (i.e., 0xFE000000) is not
    verified by the driver, which is what this patch introduces. With this
    patch, if address PMC_BASE_ADDR_DEFAULT is in RAM, then the driver will
    not attempt to ioremap the aforementioned address.

    Signed-off-by: M. Vefa Bicakci <m.v.b@runbox.com>
    Signed-off-by: Andy Shevchenko <andriy.shevchenko@linux.intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cbcee8af9d9b0172be86be296aa99c0247a04e3f
Author: Gayatri Kammela <gayatri.kammela@intel.com>
Date:   Thu Sep 5 12:30:17 2019 -0700

    x86/cpu: Add Tiger Lake to Intel family

    [ Upstream commit 6e1c32c5dbb4b90eea8f964c2869d0bde050dbe0 ]

    Add the model numbers/CPUIDs of Tiger Lake mobile and desktop to the
    Intel family.

    Suggested-by: Tony Luck <tony.luck@intel.com>
    Signed-off-by: Gayatri Kammela <gayatri.kammela@intel.com>
    Signed-off-by: Tony Luck <tony.luck@intel.com>
    Reviewed-by: Tony Luck <tony.luck@intel.com>
    Cc: Linus Torvalds <torvalds@linux-foundation.org>
```

```
    Cc: Peter Zijlstra <peterz@infradead.org>
    Cc: Rahul Tanwar <rahul.tanwar@linux.intel.com>
    Cc: Thomas Gleixner <tglx@linutronix.de>
    Link: https://lkml.kernel.org/r/20190905193020.14707-2-tony.luck@intel.com
    Signed-off-by: Ingo Molnar <mingo@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f2c56eadc277d5094401fef16bfd292727d7e7f5
Author: Marc Zyngier <maz@kernel.org>
Date:   Thu Sep 5 14:56:47 2019 +0100

    irqchip/gic-v3-its: Fix LPI release for Multi-MSI devices

    [ Upstream commit c9c96e30ecaa0aafa225aa1a5392cb7db17c7a82 ]

    When allocating a range of LPIs for a Multi-MSI capable device,
    this allocation extended to the closest power of 2.

    But on the release path, the interrupts are released one by
    one. This results in not releasing the "extra" range, leaking
    the its_device. Trying to reprobe the device will then fail.

    Fix it by releasing the LPIs the same way we allocate them.

    Fixes: 8208d1708b88 ("irqchip/gic-v3-its: Align PCI Multi-MSI allocation on their size")
    Reported-by: Jiaxing Luo <luojiaxing@huawei.com>
    Tested-by: John Garry <john.garry@huawei.com>
    Signed-off-by: Marc Zyngier <maz@kernel.org>
    Link: https://lore.kernel.org/r/f5e948aa-e32f-3f74-ae30-31fee06c2a74@huawei.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2db74f73d6235e0da99cdb9cc6126255b9591256
Author: Harald Freudenberger <freude@linux.ibm.com>
Date:   Thu Sep 5 09:38:17 2019 +0200

    s390/crypto: xts-aes-s390 fix extra run-time crypto self tests finding

    [ Upstream commit 9e323d45ba94262620a073a3f9945ca927c07c71 ]

    With 'extra run-time crypto self tests' enabled, the selftest
    for s390-xts fails with

      alg: skcipher: xts-aes-s390 encryption unexpectedly succeeded on
      test vector "random: len=0 klen=64"; expected_error=-22,
      cfg="random: inplace use_digest nosimd src_divs=[2.61%@+4006,
      84.44%@+21, 1.55%@+13, 4.50%@+344, 4.26%@+21, 2.64%@+27]"

    This special case with nbytes=0 is not handled correctly and this
    fix now makes sure that -EINVAL is returned when there is en/decrypt
    called with 0 bytes to en/decrypt.

    Signed-off-by: Harald Freudenberger <freude@linux.ibm.com>
    Signed-off-by: Vasily Gorbik <gor@linux.ibm.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0310d84339a7e88d6445da668778f968255bb654
Author: Christoph Hellwig <hch@lst.de>
Date:   Tue Sep 3 11:32:20 2019 +0200

    irqchip/sifive-plic: set max threshold for ignored handlers

    [ Upstream commit 9ce06497c2722a0f9109e4cc3ce35b7a69617886 ]

    When running in M-mode, the S-mode plic handlers are still listed in the
    device tree.  Ignore them by setting the maximum threshold.
```

```
    Signed-off-by: Christoph Hellwig <hch@lst.de>
    Acked-by: Marc Zyngier <maz@kernel.org>
    Signed-off-by: Paul Walmsley <paul.walmsley@sifive.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b4b15348642c8da219d6ec8ee1b48554574954ed
Author: Peter Zijlstra <peterz@infradead.org>
Date:   Tue Sep 3 09:53:52 2019 +0200

    x86/mm: Fix cpumask_of_node() error condition

    [ Upstream commit bc04a049f058a472695aa22905d57e2b1f4c77d9 ]

    When CONFIG_DEBUG_PER_CPU_MAPS=y we validate that the @node argument of
    cpumask_of_node() is a valid node_id. It however forgets to check for
    negative numbers. Fix this by explicitly casting to unsigned int.

      (unsigned)node >= nr_node_ids

    verifies: 0 <= node < nr_node_ids

    Also ammend the error message to match the condition.

    Signed-off-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Cc: Thomas Gleixner <tglx@linutronix.de>
    Cc: Borislav Petkov <bp@alien8.de>
    Cc: Yunsheng Lin <linyunsheng@huawei.com>
    Link: https://lkml.kernel.org/r/20190903075352.GY2369@hirez.programming.kicks-ass.net
    Signed-off-by: Ingo Molnar <mingo@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c49d6e417d12ba995564953be7e085f084ed80fc
Author: Masami Hiramatsu <mhiramat@kernel.org>
Date:   Tue Sep 3 20:08:21 2019 +0900

    kprobes: Prohibit probing on BUG() and WARN() address

    [ Upstream commit e336b4027775cb458dc713745e526fa1a1996b2a ]

    Since BUG() and WARN() may use a trap (e.g. UD2 on x86) to
    get the address where the BUG() has occurred, kprobes can not
    do single-step out-of-line that instruction. So prohibit
    probing on such address.

    Without this fix, if someone put a kprobe on WARN(), the
    kernel will crash with invalid opcode error instead of
    outputing warning message, because kernel can not find
    correct bug address.

    Signed-off-by: Masami Hiramatsu <mhiramat@kernel.org>
    Acked-by: Steven Rostedt (VMware) <rostedt@goodmis.org>
    Acked-by: Naveen N. Rao <naveen.n.rao@linux.vnet.ibm.com>
    Cc: Anil S Keshavamurthy <anil.s.keshavamurthy@intel.com>
    Cc: David S . Miller <davem@davemloft.net>
    Cc: Linus Torvalds <torvalds@linux-foundation.org>
    Cc: Naveen N . Rao <naveen.n.rao@linux.ibm.com>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Cc: Steven Rostedt <rostedt@goodmis.org>
    Cc: Thomas Gleixner <tglx@linutronix.de>
    Link: https://lkml.kernel.org/r/156750890133.19112.3393666300746167111.stgit@devnote2
    Signed-off-by: Ingo Molnar <mingo@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit fc3fb87ef9e45f6ccbe7a79a5d0be4e19e34fe92
Author: Peter Ujfalusi <peter.ujfalusi@ti.com>
Date:   Fri Aug 23 15:56:14 2019 +0300

    dmaengine: ti: edma: Do not reset reserved paRAM slots

    [ Upstream commit c5dbe60664b3660f5ac5854e21273ea2e7ff698f ]

    Skip resetting paRAM slots marked as reserved as they might be used by
    other cores.

    Signed-off-by: Peter Ujfalusi <peter.ujfalusi@ti.com>
    Link: https://lore.kernel.org/r/20190823125618.8133-2-peter.ujfalusi@ti.com
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d6092a9624ce32491e298f6b248b6ab31b2bbc5a
Author: Yufen Yu <yuyufen@huawei.com>
Date:   Tue Sep 3 21:12:41 2019 +0800

    md/raid1: fail run raid1 array when active disk less than one

    [ Upstream commit 07f1a6850c5d5a65c917c3165692b5179ac4cb6b ]

    When run test case:
      mdadm -CR /dev/md1 -l 1 -n 4 /dev/sd[a-d] --assume-clean --bitmap=internal
      mdadm -S /dev/md1
      mdadm -A /dev/md1 /dev/sd[b-c] --run --force

      mdadm --zero /dev/sda
      mdadm /dev/md1 -a /dev/sda

      echo offline > /sys/block/sdc/device/state
      echo offline > /sys/block/sdb/device/state
      sleep 5
      mdadm -S /dev/md1

      echo running > /sys/block/sdb/device/state
      echo running > /sys/block/sdc/device/state
      mdadm -A /dev/md1 /dev/sd[a-c] --run --force

    mdadm run fail with kernel message as follow:
    [  172.986064] md: kicking non-fresh sdb from array!
    [  173.004210] md: kicking non-fresh sdc from array!
    [  173.022383] md/raid1:md1: active with 0 out of 4 mirrors
    [  173.022406] md1: failed to create bitmap (-5)

    In fact, when active disk in raid1 array less than one, we
    need to return fail in raid1_run().

    Reviewed-by: NeilBrown <neilb@suse.de>
    Signed-off-by: Yufen Yu <yuyufen@huawei.com>
    Signed-off-by: Song Liu <songliubraving@fb.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit fd38427ca2084c839b8064f0474cf9dfd65eb912
Author: Wang Shenran <shenran268@gmail.com>
Date:   Wed Jul 24 11:01:10 2019 +0300

    hwmon: (acpi_power_meter) Change log level for 'unsafe software power cap'

    [ Upstream commit 6e4d91aa071810deac2cd052161aefb376ecf04e ]

    At boot time, the acpi_power_meter driver logs the following error level
    message: "Ignoring unsafe software power cap". Having read about it from
```

```
    a few sources, it seems that the error message can be quite misleading.

    While the message can imply that Linux is ignoring the fact that the
    system is operating in potentially dangerous conditions, the truth is
    the driver found an ACPI_PMC object that supports software power
    capping. The driver simply decides not to use it, perhaps because it
    doesn't support the object.

    The best solution is probably changing the log level from error to warning.
    All sources I have found, regarding the error, have downplayed its
    significance. There is not much of a reason for it to be on error level,
    while causing potential confusions or misinterpretations.

    Signed-off-by: Wang Shenran <shenran268@gmail.com>
    Link: https://lore.kernel.org/r/20190724080110.6952-1-shenran268@gmail.com
    Signed-off-by: Guenter Roeck <linux@roeck-us.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b5d174bdd5679e01e0fe6b16438c84f1946de155
Author: Marcel Bocu <marcel.p.bocu@gmail.com>
Date:   Mon Jul 22 20:46:53 2019 +0300

    hwmon: (k10temp) Add support for AMD family 17h, model 70h CPUs

    [ Upstream commit 12163cfbfc0f804cc7d27bc20e8d266ce7459260 ]

    It would seem like model 70h is behaving in the same way as model 30h,
    so let's just add the new F3 PCI ID to the list of compatible devices.

    Unlike previous Ryzen/Threadripper, Ryzen gen 3 processors do not need
    temperature offsets anymore. This has been reported in the press and
    verified on my Ryzen 3700X by checking that the idle temperature
    reported by k10temp is matching the temperature reported by the
    firmware.

    Vicki Pfau sent an identical patch after I checked that no-one had
    written this patch. I would have been happy about dropping my patch but
    unlike for his patch series, I had already Cc:ed the x86 people and
    they already reviewed the changes. Since Vicki has not answered to
    any email after his initial series, let's assume she is on vacation
    and let's avoid duplication of reviews from the maintainers and merge
    my series. To acknowledge Vicki's anteriority, I added her S-o-b to
    the patch.

    v2, suggested by Guenter Roeck and Brian Woods:
      - rename from 71h to 70h

    Signed-off-by: Vicki Pfau <vi@endrift.com>
    Signed-off-by: Marcel Bocu <marcel.p.bocu@gmail.com>
    Tested-by: Marcel Bocu <marcel.p.bocu@gmail.com>

    Cc: Thomas Gleixner <tglx@linutronix.de>
    Cc: Ingo Molnar <mingo@redhat.com>
    Cc: Borislav Petkov <bp@alien8.de>
    Cc: "H. Peter Anvin" <hpa@zytor.com>
    Cc: x86@kernel.org
    Cc: "Woods, Brian" <Brian.Woods@amd.com>
    Cc: Clemens Ladisch <clemens@ladisch.de>
    Cc: Jean Delvare <jdelvare@suse.com>
    Cc: Guenter Roeck <linux@roeck-us.net>
    Cc: linux-hwmon@vger.kernel.org
    Link: https://lore.kernel.org/r/20190722174653.2391-1-marcel.p.bocu@gmail.com
    Signed-off-by: Guenter Roeck <linux@roeck-us.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit eb35e355da6868522d4b0c17c2cba6e4c3b6f7c6
Author: Kent Overstreet <kent.overstreet@gmail.com>
Date:   Tue Sep 3 21:25:45 2019 +0800

    closures: fix a race on wakeup from closure_sync

    [ Upstream commit a22a9602b88fabf10847f238ff81fde5f906fef7 ]

    The race was when a thread using closure_sync() notices cl->s->done == 1
    before the thread calling closure_put() calls wake_up_process(). Then,
    it's possible for that thread to return and exit just before
    wake_up_process() is called - so we're trying to wake up a process that
    no longer exists.

    rcu_read_lock() is sufficient to protect against this, as there's an rcu
    barrier somewhere in the process teardown path.

    Signed-off-by: Kent Overstreet <kent.overstreet@gmail.com>
    Acked-by: Coly Li <colyli@suse.de>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7f8a4608585def33ece1c5ad3952f93ca6348717
Author: Wenwen Wang <wenwen@cs.uga.edu>
Date:   Tue Aug 20 22:44:19 2019 -0500

    ACPI / PCI: fix acpi_pci_irq_enable() memory leak

    [ Upstream commit 29b49958cf73b439b17fa29e9a25210809a6c01c ]

    In acpi_pci_irq_enable(), 'entry' is allocated by kzalloc() in
    acpi_pci_irq_check_entry() (invoked from acpi_pci_irq_lookup()). However,
    it is not deallocated if acpi_pci_irq_valid() returns false, leading to a
    memory leak. To fix this issue, free 'entry' before returning 0.

    Fixes: e237a5518425 ("x86/ACPI/PCI: Recognize that Interrupt Line 255 means "not
connected"")
    Signed-off-by: Wenwen Wang <wenwen@cs.uga.edu>
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 06cd4a06eb596a888239fb8ceb6ea15677cab396
Author: Wenwen Wang <wenwen@cs.uga.edu>
Date:   Fri Aug 16 00:08:27 2019 -0500

    ACPI: custom_method: fix memory leaks

    [ Upstream commit 03d1571d9513369c17e6848476763ebbd10ec2cb ]

    In cm_write(), 'buf' is allocated through kzalloc(). In the following
    execution, if an error occurs, 'buf' is not deallocated, leading to memory
    leaks. To fix this issue, free 'buf' before returning the error.

    Fixes: 526b4af47f44 ("ACPI: Split out custom_method functionality into an own driver")
    Signed-off-by: Wenwen Wang <wenwen@cs.uga.edu>
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a16f9916aab00f27c8dd8a84a392ac25748e56f7
Author: Marcel Bocu <marcel.p.bocu@gmail.com>
Date:   Mon Jul 22 20:45:10 2019 +0300

    x86/amd_nb: Add PCI device IDs for family 17h, model 70h

    [ Upstream commit af4e1c5eca95bed1192d8dc45c8ed63aea2209e8 ]
```

```
    The AMD Ryzen gen 3 processors came with a different PCI IDs for the
    function 3 & 4 which are used to access the SMN interface. The root
    PCI address however remained at the same address as the model 30h.

    Adding the F3/F4 PCI IDs respectively to the misc and link ids appear
    to be sufficient for k10temp, so let's add them and follow up on the
    patch if other functions need more tweaking.

    Vicki Pfau sent an identical patch after I checked that no-one had
    written this patch. I would have been happy about dropping my patch but
    unlike for his patch series, I had already Cc:ed the x86 people and
    they already reviewed the changes. Since Vicki has not answered to
    any email after his initial series, let's assume she is on vacation
    and let's avoid duplication of reviews from the maintainers and merge
    my series. To acknowledge Vicki's anteriority, I added her S-o-b to
    the patch.

    v2, suggested by Guenter Roeck and Brian Woods:
     - rename from 71h to 70h

    Signed-off-by: Vicki Pfau <vi@endrift.com>
    Signed-off-by: Marcel Bocu <marcel.p.bocu@gmail.com>
    Tested-by: Marcel Bocu <marcel.p.bocu@gmail.com>
    Acked-by: Thomas Gleixner <tglx@linutronix.de>
    Acked-by: Brian Woods <brian.woods@amd.com>
    Acked-by: Bjorn Helgaas <bhelgaas@google.com>   # pci_ids.h

    Cc: Thomas Gleixner <tglx@linutronix.de>
    Cc: Ingo Molnar <mingo@redhat.com>
    Cc: Borislav Petkov <bp@alien8.de>
    Cc: "H. Peter Anvin" <hpa@zytor.com>
    Cc: x86@kernel.org
    Cc: "Woods, Brian" <Brian.Woods@amd.com>
    Cc: Clemens Ladisch <clemens@ladisch.de>
    Cc: Jean Delvare <jdelvare@suse.com>
    Cc: Guenter Roeck <linux@roeck-us.net>
    Cc: linux-hwmon@vger.kernel.org
    Link: https://lore.kernel.org/r/20190722174510.2179-1-marcel.p.bocu@gmail.com
    Signed-off-by: Guenter Roeck <linux@roeck-us.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d77a5fec4cf96ec1ac1e7aba82b4e43c9e77a051
Author: Marek Szyprowski <m.szyprowski@samsung.com>
Date:   Fri Aug 30 14:52:42 2019 +0200

    ARM: dts: exynos: Mark LDO10 as always-on on Peach Pit/Pi Chromebooks

    [ Upstream commit 5b0eeeaa37615df37a9a30929b73e9defe61ca84 ]

    Commit aff138bf8e37 ("ARM: dts: exynos: Add TMU nodes regulator supply
    for Peach boards") assigned LDO10 to Exynos Thermal Measurement Unit,
    but it turned out that it supplies also some other critical parts and
    board freezes/crashes when it is turned off.

    The mentioned commit made Exynos TMU a consumer of that regulator and in
    typical case Exynos TMU driver keeps it enabled from early boot. However
    there are such configurations (example is multi_v7_defconfig), in which
    some of the regulators are compiled as modules and are not available
    from early boot. In such case it may happen that LDO10 is turned off by
    regulator core, because it has no consumers yet (in this case consumer
    drivers cannot get it, because the supply regulators for it are not yet
    available). This in turn causes the board to crash. This patch restores
    'always-on' property for the LDO10 regulator.
```

```
    Fixes: aff138bf8e37 ("ARM: dts: exynos: Add TMU nodes regulator supply for Peach
boards")
    Signed-off-by: Marek Szyprowski <m.szyprowski@samsung.com>
    Signed-off-by: Krzysztof Kozlowski <krzk@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8fc765faef7073b86b8f412a5da4a44991f33f2c
Author: Maxime Ripard <maxime.ripard@bootlin.com>
Date:   Wed Aug 28 14:52:05 2019 +0200

    ASoC: dt-bindings: sun4i-spdif: Fix dma-names warning

    [ Upstream commit 1a8e7cdfa4f5872bf0c202d09bff6628aba6b9f6 ]

    Even though the H6 compatible has been properly added, the exeption for the
    number of DMA channels hasn't been updated, leading in a validation
    warning.

    Fix this.

    Fixes: b20453031472 ("dt-bindings: sound: sun4i-spdif: Add Allwinner H6 compatible")
    Signed-off-by: Maxime Ripard <maxime.ripard@bootlin.com>
    Link: https://lore.kernel.org/r/20190828125209.28173-1-mripard@kernel.org
    Reviewed-by: Rob Herring <robh@kernel.org>
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4b444f3c0ee9ab8bd4524d3450ae9337eb546a86
Author: Tzvetomir Stoyanov <tstoyanov@vmware.com>
Date:   Mon Aug 5 16:43:15 2019 -0400

    libtraceevent: Change users plugin directory

    [ Upstream commit e97fd1383cd77c467d2aed7fa4e596789df83977 ]

    To be compliant with XDG user directory layout, the user's plugin
    directory is changed from ~/.traceevent/plugins to
    ~/.local/lib/traceevent/plugins/

    Suggested-by: Patrick McLean <chutzpah@gentoo.org>
    Signed-off-by: Tzvetomir Stoyanov <tstoyanov@vmware.com>
    Cc: Andrew Morton <akpm@linux-foundation.org>
    Cc: Jiri Olsa <jolsa@redhat.com>
    Cc: Namhyung Kim <namhyung@kernel.org>
    Cc: Patrick McLean <chutzpah@gentoo.org>
    Cc: linux-trace-devel@vger.kernel.org
    Link: https://lore.kernel.org/linux-trace-devel/20190313144206.41e75cf8@patrickm/
    Link: http://lore.kernel.org/linux-trace-devel/20190801074959.22023-4-
tz.stoyanov@gmail.com
    Link: http://lore.kernel.org/lkml/20190805204355.344622683@goodmis.org
    Signed-off-by: Steven Rostedt (VMware) <rostedt@goodmis.org>
    Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 10fa4d07f6550bd1bca2937939c9926cd8116f28
Author: Eric Dumazet <edumazet@google.com>
Date:   Wed Aug 28 06:13:38 2019 -0700

    iommu/iova: Avoid false sharing on fq_timer_on

    [ Upstream commit 0d87308cca2c124f9bce02383f1d9632c9be89c4 ]

    In commit 14bd9a607f90 ("iommu/iova: Separate atomic variables
    to improve performance") Jinyu Qi identified that the atomic_cmpxchg()
    in queue_iova() was causing a performance loss and moved critical fields
```

```
    so that the false sharing would not impact them.

    However, avoiding the false sharing in the first place seems easy.
    We should attempt the atomic_cmpxchg() no more than 100 times
    per second. Adding an atomic_read() will keep the cache
    line mostly shared.

    This false sharing came with commit 9a005a800ae8
    ("iommu/iova: Add flush timer").

    Signed-off-by: Eric Dumazet <edumazet@google.com>
    Fixes: 9a005a800ae8 ('iommu/iova: Add flush timer')
    Cc: Jinyu Qi <jinyuqi@huawei.com>
    Cc: Joerg Roedel <jroedel@suse.de>
    Acked-by: Robin Murphy <robin.murphy@arm.com>
    Signed-off-by: Joerg Roedel <jroedel@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 61376531360f13bf0c4eeb64cd832f67b4c380f3
Author: Dan Williams <dan.j.williams@intel.com>
Date:   Thu Aug 29 16:30:34 2019 -0700

    libata/ahci: Drop PCS quirk for Denverton and beyond

    [ Upstream commit c312ef176399e04fc5f7f2809d9a589751fbf6d9 ]

    The Linux ahci driver has historically implemented a configuration fixup
    for platforms / platform-firmware that fails to enable the ports prior
    to OS hand-off at boot. The fixup was originally implemented way back
    before ahci moved from drivers/scsi/ to drivers/ata/, and was updated in
    2007 via commit 49f290903935 "ahci: update PCS programming". The quirk
    sets a port-enable bitmap in the PCS register at offset 0x92.

    This quirk could be applied generically up until the arrival of the
    Denverton (DNV) platform. The DNV AHCI controller architecture supports
    more than 6 ports and along with that the PCS register location and
    format were updated to allow for more possible ports in the bitmap. DNV
    AHCI expands the register to 32-bits and moves it to offset 0x94.

    As it stands there are no known problem reports with existing Linux
    trying to set bits at offset 0x92 which indicates that the quirk is not
    applicable. Likely it is not applicable on a wider range of platforms,
    but it is difficult to discern which platforms if any still depend on
    the quirk.

    Rather than try to fix the PCS quirk to consider the DNV register layout
    instead require explicit opt-in. The assumption is that the OS driver
    need not touch this register, and platforms can be added with a new
    boad_ahci_pcs7 board-id when / if problematic platforms are found in the
    future. The logic in ahci_intel_pcs_quirk() looks for all Intel AHCI
    instances with "legacy" board-ids and otherwise skips the quirk if the
    board was matched by class-code.

    Reported-by: Stephen Douthit <stephend@silicom-usa.com>
    Cc: Christoph Hellwig <hch@infradead.org>
    Reviewed-by: Stephen Douthit <stephend@silicom-usa.com>
    Signed-off-by: Dan Williams <dan.j.williams@intel.com>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0ee33b4afaac8225c13eae012e65e9502c074e76
Author: Cezary Rojewski <cezary.rojewski@intel.com>
Date:   Thu Aug 22 13:36:13 2019 +0200

    ASoC: Intel: Haswell: Adjust machine device private context
```

```
    [ Upstream commit ca964edf0ddbfec2cb10b3d251d09598e7ca9b13 ]

    Apart from Haswell machines, all other devices have their private data
    set to snd_soc_acpi_mach instance.

    Changes for HSW/ BDW boards introduced with series:
    https://patchwork.kernel.org/cover/10782035/

    added support for dai_link platform_name adjustments within card probe
    routines. These take for granted private_data points to
    snd_soc_acpi_mach whereas for Haswell, it's sst_pdata instead. Change
    private context of platform_device - representing machine board - to
    address this.

    Fixes: e87055d732e3 ("ASoC: Intel: haswell: platform name fixup support")
    Fixes: 7e40ddcf974a ("ASoC: Intel: bdw-rt5677: platform name fixup support")
    Fixes: 2d067b2807f9 ("ASoC: Intel: broadwell: platform name fixup support")
    Signed-off-by: Cezary Rojewski <cezary.rojewski@intel.com>
    Link: https://lore.kernel.org/r/20190822113616.22702-2-cezary.rojewski@intel.com
    Tested-by: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com>
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f807c17b5395ebf5aac071f50bc7c794895dd835
Author: Qian Cai <cai@lca.pw>
Date:   Wed Aug 28 17:39:43 2019 -0400

    iommu/amd: Silence warnings under memory pressure

    [ Upstream commit 3d708895325b78506e8daf00ef31549476e8586a ]

    When running heavy memory pressure workloads, the system is throwing
    endless warnings,

    smartpqi 0000:23:00.0: AMD-Vi: IOMMU mapping error in map_sg (io-pages:
    5 reason: -12)
    Hardware name: HPE ProLiant DL385 Gen10/ProLiant DL385 Gen10, BIOS A40
    07/10/2019
    swapper/10: page allocation failure: order:0, mode:0xa20(GFP_ATOMIC),
    nodemask=(null),cpuset=/,mems_allowed=0,4
    Call Trace:
     <IRQ>
     dump_stack+0x62/0x9a
     warn_alloc.cold.43+0x8a/0x148
     __alloc_pages_nodemask+0x1a5c/0x1bb0
     get_zeroed_page+0x16/0x20
     iommu_map_page+0x477/0x540
     map_sg+0x1ce/0x2f0
     scsi_dma_map+0xc6/0x160
     pqi_raid_submit_scsi_cmd_with_io_request+0x1c3/0x470 [smartpqi]
     do_IRQ+0x81/0x170
     common_interrupt+0xf/0xf
     </IRQ>

    because the allocation could fail from iommu_map_page(), and the volume
    of this call could be huge which may generate a lot of serial console
    output and cosumes all CPUs.

    Fix it by silencing the warning in this call site, and there is still a
    dev_err() later to notify the failure.

    Signed-off-by: Qian Cai <cai@lca.pw>
    Signed-off-by: Joerg Roedel <jroedel@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 53ff50aa79e2a4813b859065dfe60367fa9e80ac
Author: Takashi Sakamoto <o-takashi@sakamocchi.jp>
Date:   Fri Aug 30 10:14:56 2019 +0900

    ALSA: firewire-motu: add support for MOTU 4pre

    [ Upstream commit 6af86bdb8ad41f4cf1292d3b10857dc322758328 ]

    MOTU 4pre was launched in 2012 by MOTU, Inc. This commit allows userspace
    applications can transmit and receive PCM frames and MIDI messages for
    this model via ALSA PCM interface and RawMidi/Sequencer interfaces.

    The device supports MOTU protocol version 3. Unlike the other devices, the
    device is simply designed. The size of data block is fixed to 10 quadlets
    during available sampling rates (44.1 - 96.0 kHz). Each data block
    includes 1 source packet header, 2 data chunks for messages, 8 data chunks
    for PCM samples and 2 data chunks for padding to quadlet alignment. The
    device has no MIDI, optical, BNC and AES/EBU interfaces.

    Like support for the other MOTU devices, the quality of playback sound
    is not enough good with periodical noise yet.

    $ python2 crpp < ~/git/am-config-rom/motu/motu-4pre.img
                  ROM header and bus information block
                  ------------------------------------------------------------------
    400  041078cc  bus_info_length 4, crc_length 16, crc 30924
    404  31333934  bus_name "1394"
    408  20ff7000  irmc 0, cmc 0, isc 1, bmc 0, cyc_clk_acc 255, max_rec 7 (256)
    40c  0001f200  company_id 0001f2     |
    410  000a41c5  device_id 00000a41c5  | EUI-64 0001f200000a41c5

                  root directory
                  ------------------------------------------------------------------
    414  0004ef04  directory_length 4, crc 61188
    418  030001f2  vendor
    41c  0c0083c0  node capabilities per IEEE 1394
    420  d1000002  --> unit directory at 428
    424  8d000005  --> eui-64 leaf at 438

                  unit directory at 428
                  ------------------------------------------------------------------
    428  0003ceda  directory_length 3, crc 52954
    42c  120001f2  specifier id
    430  13000045  version
    434  17103800  model

                  eui-64 leaf at 438
                  ------------------------------------------------------------------
    438  0002d248  leaf_length 2, crc 53832
    43c  0001f200  company_id 0001f2     |
    440  000a41c5  device_id 00000a41c5  | EUI-64 0001f200000a41c5

    Signed-off-by: Takashi Sakamoto <o-takashi@sakamocchi.jp>
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9fc4bb602b859ddcb6448651ccab2ae5208181c4
Author: Anton Eidelman <anton@lightbitslabs.com>
Date:   Fri Aug 16 13:00:10 2019 -0700

    nvme-multipath: fix ana log nsid lookup when nsid is not found

    [ Upstream commit e01f91dff91c7b16a6e3faf2565017d497a73f83 ]
```

```
    ANA log parsing invokes nvme_update_ana_state() per ANA group desc.
    This updates the state of namespaces with nsids in desc->nsids[].

    Both ctrl->namespaces list and desc->nsids[] array are sorted by nsid.
    Hence nvme_update_ana_state() performs a single walk over ctrl->namespaces:
    - if current namespace matches the current desc->nsids[n],
      this namespace is updated, and n is incremented.
    - the process stops when it encounters the end of either
      ctrl->namespaces end or desc->nsids[]

    In case desc->nsids[n] does not match any of ctrl->namespaces,
    the remaining nsids following desc->nsids[n] will not be updated.
    Such situation was considered abnormal and generated WARN_ON_ONCE.

    However ANA log MAY contain nsids not (yet) found in ctrl->namespaces.
    For example, lets consider the following scenario:
    - nvme0 exposes namespaces with nsids = [2, 3] to the host
    - a new namespace nsid = 1 is added dynamically
    - also, a ANA topology change is triggered
    - NS_CHANGED aen is generated and triggers scan_work
    - before scan_work discovers nsid=1 and creates a namespace, a NOTICE_ANA
      aen was issues and ana_work receives ANA log with nsids=[1, 2, 3]

    Result: ana_work fails to update ANA state on existing namespaces [2, 3]

    Solution:
    Change the way nvme_update_ana_state() namespace list walk
    checks the current namespace against desc->nsids[n] as follows:
    a) ns->head->ns_id < desc->nsids[n]: keep walking ctrl->namespaces.
    b) ns->head->ns_id == desc->nsids[n]: match, update the namespace
    c) ns->head->ns_id >= desc->nsids[n]: skip to desc->nsids[n+1]

    This enables correct operation in the scenario described above.
    This also allows ANA log to contain nsids currently invisible
    to the host, i.e. inactive nsids.

    Signed-off-by: Anton Eidelman <anton@lightbitslabs.com>
    Reviewed-by:   James Smart <james.smart@broadcom.com>
    Reviewed-by: Hannes Reinecke <hare@suse.com>
    Reviewed-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2f2a9d2b0d1b79d7b1432705a5bdf4ed20075a84
Author: Tom Wu <tomwu@mellanox.com>
Date:   Thu Aug 8 02:22:36 2019 +0000

    nvmet: fix data units read and written counters in SMART log

    [ Upstream commit 3bec2e3754becebd4c452999adb49bc62c575ea4 ]

    In nvme spec 1.3 there is a definition for data write/read counters
    from SMART log, (See section 5.14.1.2):
            This value is reported in thousands (i.e., a value of 1
            corresponds to 1000 units of 512 bytes read) and is rounded up.

    However, in nvme target where value is reported with actual units,
    but not thousands of units as the spec requires.

    Signed-off-by: Tom Wu <tomwu@mellanox.com>
    Reviewed-by: Israel Rukshin <israelr@mellanox.com>
    Reviewed-by: Max Gurtovoy <maxg@mellanox.com>
    Reviewed-by: Chaitanya Kulkarni <chaitanya.kulkarni@wdc.com>
    Reviewed-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Sagi Grimberg <sagi@grimberg.me>
```

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

commit 37ec2ae0aaba5c90bce903655539fe7b9f72bb3d
Author: Song Liu <songliubraving@fb.com>
Date:   Wed Aug 28 23:54:55 2019 +0200

    x86/mm/pti: Handle unaligned address gracefully in pti_clone_pagetable()

    [ Upstream commit 825d0b73cd7526b0bb186798583fae810091cbac ]

    pti_clone_pmds() assumes that the supplied address is either:

     - properly PUD/PMD aligned
    or
     - the address is actually mapped which means that independently
       of the mapping level (PUD/PMD/PTE) the next higher mapping
       exists.

    If that's not the case the unaligned address can be incremented by PUD or
    PMD size incorrectly. All callers supply mapped and/or aligned addresses,
    but for the sake of robustness it's better to handle that case properly and
    to emit a warning.

    [ tglx: Rewrote changelog and added WARN_ON_ONCE() ]

    Signed-off-by: Song Liu <songliubraving@fb.com>
    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Reviewed-by: Ingo Molnar <mingo@kernel.org>
    Acked-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Link:
https://lkml.kernel.org/r/alpine.DEB.2.21.1908282352470.1938@nanos.tec.linutronix.de
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b2d26f8b9f568f7e719cee45105a0dc2b53a8c19
Author: Shengjiu Wang <shengjiu.wang@nxp.com>
Date:   Wed Aug 28 13:20:17 2019 -0400

    ASoC: fsl_ssi: Fix clock control issue in master mode

    [ Upstream commit 696d05225cebffd172008d212657be90e823eac0 ]

    The test case is
    arecord -Dhw:0 -d 10 -f S16_LE -r 48000 -c 2 temp.wav &
    aplay -Dhw:0 -d 30 -f S16_LE -r 48000 -c 2 test.wav

    There will be error after end of arecord:
    aplay: pcm_write:2051: write error: Input/output error

    Capture and Playback work in parallel in master mode, one
    substream stops, the other substream is impacted, the
    reason is that clock is disabled wrongly.

    The clock's reference count is not increased when second
    substream starts, the hw_param() function returns in the
    beginning because first substream is enabled, then in end
    of first substream, the hw_free() disables the clock.

    This patch is to move the clock enablement to the place
    before checking of the device enablement in hw_param().

    Signed-off-by: Shengjiu Wang <shengjiu.wang@nxp.com>
    Link: https://lore.kernel.org/r/1567012817-12625-1-git-send-email-shengjiu.wang@nxp.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 3a65a1a1d7ffeaadc7043080c700e182827c2017
Author: Thomas Gleixner <tglx@linutronix.de>
Date:   Wed Aug 28 16:24:47 2019 +0200

    x86/mm/pti: Do not invoke PTI functions when PTI is disabled

    [ Upstream commit 990784b57731192b7d90c8d4049e6318d81e887d ]

    When PTI is disabled at boot time either because the CPU is not affected or
    PTI has been disabled on the command line, the boot code still calls into
    pti_finalize() which then unconditionally invokes:

         pti_clone_entry_text()
         pti_clone_kernel_text()

    pti_clone_kernel_text() was called unconditionally before the 32bit support
    was added and 32bit added the call to pti_clone_entry_text().

    The call has no side effects as cloning the page tables into the available
    second one, which was allocated for PTI does not create damage. But it does
    not make sense either and in case that this functionality would be extended
    later this might actually lead to hard to diagnose issues.

    Neither function should be called when PTI is runtime disabled. Make the
    invocation conditional.

    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Reviewed-by: Dave Hansen <dave.hansen@linux.intel.com>
    Acked-by: Ingo Molnar <mingo@kernel.org>
    Acked-by: Song Liu <songliubraving@fb.com>
    Acked-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Link: https://lkml.kernel.org/r/20190828143124.063353972@linutronix.de
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 81c3a46e33930d612c8004b352ac44980f3b9d16
Author: Andrew Murray <andrew.murray@arm.com>
Date:   Wed Aug 28 18:50:05 2019 +0100

    jump_label: Don't warn on __exit jump entries

    [ Upstream commit 8f35eaa5f2de020073a48ad51112237c5932cfcc ]

    On architectures that discard .exit.* sections at runtime, a
    warning is printed for each jump label that is used within an
    in-kernel __exit annotated function:

    can't patch jump_label at ehci_hcd_cleanup+0x8/0x3c
    WARNING: CPU: 0 PID: 1 at kernel/jump_label.c:410 __jump_label_update+0x12c/0x138

    As these functions will never get executed (they are free'd along
    with the rest of initmem) - we do not need to patch them and should
    not display any warnings.

    The warning is displayed because the test required to satisfy
    jump_entry_is_init is based on init_section_contains (__init_begin to
    __init_end) whereas the test in __jump_label_update is based on
    init_kernel_text (_sinittext to _einittext) via kernel_text_address).

    Fixes: 19483677684b ("jump_label: Annotate entries that operate on __init code earlier")
    Signed-off-by: Andrew Murray <andrew.murray@arm.com>
    Acked-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Acked-by: Mark Rutland <mark.rutland@arm.com>
    Signed-off-by: Will Deacon <will@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 98d94b5c74ba3623d4a137d48a92878d7d3b4dd5
Author: Andrew Murray <andrew.murray@arm.com>
Date:   Wed Aug 28 18:50:06 2019 +0100

    arm64: Use correct ll/sc atomic constraints

    [ Upstream commit 580fa1b874711d633f9b145b7777b0e83ebf3787 ]

    The A64 ISA accepts distinct (but overlapping) ranges of immediates for:

     * add arithmetic instructions ('I' machine constraint)
     * sub arithmetic instructions ('J' machine constraint)
     * 32-bit logical instructions ('K' machine constraint)
     * 64-bit logical instructions ('L' machine constraint)

    ... but we currently use the 'I' constraint for many atomic operations
    using sub or logical instructions, which is not always valid.

    When CONFIG_ARM64_LSE_ATOMICS is not set, this allows invalid immediates
    to be passed to instructions, potentially resulting in a build failure.
    When CONFIG_ARM64_LSE_ATOMICS is selected the out-of-line ll/sc atomics
    always use a register as they have no visibility of the value passed by
    the caller.

    This patch adds a constraint parameter to the ATOMIC_xx and
    __CMPXCHG_CASE macros so that we can pass appropriate constraints for
    each case, with uses updated accordingly.

    Unfortunately prior to GCC 8.1.0 the 'K' constraint erroneously accepted
    '4294967295', so we must instead force the use of a register.

    Signed-off-by: Andrew Murray <andrew.murray@arm.com>
    Signed-off-by: Will Deacon <will@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3172fe44eb840662b63a6d62e24cdcebc938e0eb
Author: Arnaldo Carvalho de Melo <acme@redhat.com>
Date:   Wed Aug 28 16:48:50 2019 -0300

    perf evlist: Use unshare(CLONE_FS) in sb threads to let setns(CLONE_NEWNS) work

    [ Upstream commit b397f8468fa27f08b83b348ffa56a226f72453af ]

    When we started using a thread to catch the PERF_RECORD_BPF_EVENT meta
    data events to then ask the kernel for further info (BTF, etc) for BPF
    programs shortly after they get loaded, we forgot to use
    unshare(CLONE_FS) as was done in:

      868a832918f6 ("perf top: Support lookup of symbols in other mount namespaces.")

    Do it so that we can enter the namespaces to read the build-ids at the
    end of a 'perf record' session for the DSOs that had hits.

    Before:

    Starting a 'stress-ng --cpus 8' inside a container and then, outside the
    container running:

      # perf record -a --namespaces sleep 5
      # perf buildid-list | grep stress-ng
      #

    We would end up with a 'perf.data' file that had no entry in its
    build-id table for the /usr/bin/stress-ng binary inside the container
    that got tons of PERF_RECORD_SAMPLEs.
```

```
    After:

      # perf buildid-list | grep stress-ng
      f2ed02c68341183a124b9b0f6e2e6c493c465b29 /usr/bin/stress-ng
      #

    Then its just a matter of making sure that that binary debuginfo package
    gets available in a place that 'perf report' will look at build-id keyed
    ELF files, which, in my case, on a f30 notebook, was a matter of
    installing the debuginfo file for the distro used in the container,
    fedora 31:

      # rpm -ivh
http://fedora.c3sl.ufpr.br/linux/development/31/Everything/x86_64/debug/tree/Packages/s/stre
ss-ng-debuginfo-0.07.29-10.fc31.x86_64.rpm

    Then, because perf currently looks for those debuginfo files (richer ELF
    symtab) inside that namespace (look at the setns calls):

      openat(AT_FDCWD, "/proc/self/ns/mnt", O_RDONLY) = 137
      openat(AT_FDCWD, "/proc/13169/ns/mnt", O_RDONLY) = 139
      setns(139, CLONE_NEWNS)                 = 0
      stat("/usr/bin/stress-ng", {st_mode=S_IFREG|0755, st_size=3065416, ...}) = 0
      openat(AT_FDCWD, "/usr/bin/stress-ng", O_RDONLY) = 140
      fcntl(140, F_GETFD)                     = 0
      fstat(140, {st_mode=S_IFREG|0755, st_size=3065416, ...}) = 0
      mmap(NULL, 3065416, PROT_READ, MAP_PRIVATE, 140, 0) = 0x7ff2fdc5b000
      munmap(0x7ff2fdc5b000, 3065416)         = 0
      close(140)                              = 0
      stat("stress-ng-0.07.29-10.fc31.x86_64.debug", 0x7fff45d71260) = -1 ENOENT (No such
file or directory)
      stat("/usr/bin/stress-ng-0.07.29-10.fc31.x86_64.debug", 0x7fff45d71260) = -1 ENOENT
(No such file or directory)
      stat("/usr/bin/.debug/stress-ng-0.07.29-10.fc31.x86_64.debug", 0x7fff45d71260) = -1
ENOENT (No such file or directory)
      stat("/usr/lib/debug/usr/bin/stress-ng-0.07.29-10.fc31.x86_64.debug", 0x7fff45d71260)
= -1 ENOENT (No such file or directory)
      stat("/root/.debug/.build-id/f2/ed02c68341183a124b9b0f6e2e6c493c465b29",
0x7fff45d711e0) = -1 ENOENT (No such file or directory)

    To only then go back to the "host" namespace to look just in the users's
    ~/.debug cache:

      setns(137, CLONE_NEWNS)                 = 0
      chdir("/root")                          = 0
      close(137)                              = 0
      close(139)                              = 0
      stat("/root/.debug/.build-id/f2/ed02c68341183a124b9b0f6e2e6c493c465b29/elf",
0x7fff45d732e0) = -1 ENOENT (No such file or directory)

    It continues to fail to resolve symbols:

      # perf report | grep stress-ng | head -5
         9.50%  stress-ng-cpu    stress-ng    [.] 0x0000000000021ac1
         8.58%  stress-ng-cpu    stress-ng    [.] 0x0000000000021ab4
         8.51%  stress-ng-cpu    stress-ng    [.] 0x0000000000021489
         7.17%  stress-ng-cpu    stress-ng    [.] 0x00000000000219b6
         3.93%  stress-ng-cpu    stress-ng    [.] 0x0000000000021478
      #

    To overcome that we use:

      # perf buildid-cache -v --add /usr/lib/debug/usr/bin/stress-ng-0.07.29-
10.fc31.x86_64.debug
```

```
       Adding f2ed02c68341183a124b9b0f6e2e6c493c465b29 /usr/lib/debug/usr/bin/stress-ng-
  0.07.29-10.fc31.x86_64.debug: Ok
       #
       # ls -la /root/.debug/.build-id/f2/ed02c68341183a124b9b0f6e2e6c493c465b29/elf
       -rw-r--r--. 3 root root 2401184 Jul 27 07:03 /root/.debug/.build-
  id/f2/ed02c68341183a124b9b0f6e2e6c493c465b29/elf
       # file /root/.debug/.build-id/f2/ed02c68341183a124b9b0f6e2e6c493c465b29/elf
       /root/.debug/.build-id/f2/ed02c68341183a124b9b0f6e2e6c493c465b29/elf: ELF 64-bit LSB
  shared object, x86-64, version 1 (SYSV), dynamically linked, interpreter \004,
  BuildID[sha1]=f2ed02c68341183a124b9b0f6e2e6c493c465b29, for GNU/Linux 3.2.0, with
  debug_info, not stripped, too many notes (256)
       #

      Now it finally works:

        # perf report | grep stress-ng | head -5
          23.59%  stress-ng-cpu    stress-ng    [.] ackermann
          23.33%  stress-ng-cpu    stress-ng    [.] is_prime
          17.36%  stress-ng-cpu    stress-ng    [.] stress_cpu_sieve
           6.08%  stress-ng-cpu    stress-ng    [.] stress_cpu_correlate
           3.55%  stress-ng-cpu    stress-ng    [.] queens_try
        #

      I'll make sure that it looks for the build-id keyed files in both the
      "host" namespace (the namespace the user running 'perf record' was a the
      time of the recording) and in the container namespace, as it shouldn't
      matter where a content based key lookup finds the ELF file to use in
      resolving symbols, etc.

      Reported-by: Karl Rister <krister@redhat.com>
      Cc: Alexander Shishkin <alexander.shishkin@linux.intel.com>
      Cc: Alexei Starovoitov <ast@kernel.org>
      Cc: Brendan Gregg <brendan.d.gregg@gmail.com>
      Cc: Daniel Borkmann <daniel@iogearbox.net>
      Cc: Krister Johansen <kjlx@templeofstupid.com>
      Cc: Namhyung Kim <namhyung@kernel.org>
      Cc: Peter Zijlstra <peterz@infradead.org>
      Cc: Song Liu <songliubraving@fb.com>
      Cc: Stanislav Fomichev <sdf@google.com>
      Cc: Thomas-Mich Richter <tmricht@linux.vnet.ibm.com>
      Fixes: 657ee5531903 ("perf evlist: Introduce side band thread")
      Link: https://lkml.kernel.org/n/tip-g79k0jz41adiaeuqud742t2l@git.kernel.org
      Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 1b47d388c0f027502abd2ea3c8b90776ff4a6ab1
  Author: Mark Rutland <mark.rutland@arm.com>
  Date:   Tue Aug 27 18:12:57 2019 +0100

      arm64: kpti: ensure patched kernel text is fetched from PoU

      [ Upstream commit f32c7a8e45105bd0af76872bf6eef0438ff12fb2 ]

      While the MMUs is disabled, I-cache speculation can result in
      instructions being fetched from the PoC. During boot we may patch
      instructions (e.g. for alternatives and jump labels), and these may be
      dirty at the PoU (and stale at the PoC).

      Thus, while the MMU is disabled in the KPTI pagetable fixup code we may
      load stale instructions into the I-cache, potentially leading to
      subsequent crashes when executing regions of code which have been
      modified at runtime.

      Similarly to commit:
```

```
       8ec41987436d566f ("arm64: mm: ensure patched kernel text is fetched from PoU")

    ... we can invalidate the I-cache after enabling the MMU to prevent such
    issues.

    The KPTI pagetable fixup code itself should be clean to the PoC per the
    boot protocol, so no maintenance is required for this code.

    Signed-off-by: Mark Rutland <mark.rutland@arm.com>
    Cc: Catalin Marinas <catalin.marinas@arm.com>
    Reviewed-by: James Morse <james.morse@arm.com>
    Signed-off-by: Will Deacon <will@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5e32a9e94ef2f33bf04ba4eeee7f984922e88cf8
Author: Neil Horman <nhorman@tuxdriver.com>
Date:   Thu Aug 22 10:34:21 2019 -0400

    x86/apic/vector: Warn when vector space exhaustion breaks affinity

    [ Upstream commit 743dac494d61d991967ebcfab92e4f80dc7583b3 ]

    On x86, CPUs are limited in the number of interrupts they can have affined
    to them as they only support 256 interrupt vectors per CPU. 32 vectors are
    reserved for the CPU and the kernel reserves another 22 for internal
    purposes. That leaves 202 vectors for assignement to devices.

    When an interrupt is set up or the affinity is changed by the kernel or the
    administrator, the vector assignment code attempts to honor the requested
    affinity mask. If the vector space on the CPUs in that affinity mask is
    exhausted the code falls back to a wider set of CPUs and assigns a vector
    on a CPU outside of the requested affinity mask silently.

    While the effective affinity is reflected in the corresponding
    /proc/irq/$N/effective_affinity* files the silent breakage of the requested
    affinity can lead to unexpected behaviour for administrators.

    Add a pr_warn() when this happens so that adminstrators get at least
    informed about it in the syslog.

    [ tglx: Massaged changelog and made the pr_warn() more informative ]

    Reported-by: djuran@redhat.com
    Signed-off-by: Neil Horman <nhorman@tuxdriver.com>
    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Tested-by: djuran@redhat.com
    Link: https://lkml.kernel.org/r/20190822143421.9535-1-nhorman@tuxdriver.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 21ae0a4ada41fb2a71b434fe4708d983c8ba46e9
Author: Neil Armstrong <narmstrong@baylibre.com>
Date:   Fri Aug 23 09:02:48 2019 +0200

    arm64: dts: meson: fix boards regulators states format

    [ Upstream commit f9717178b9be9477877d4c3776c61ff56d854ddf ]

    This fixes the following DT schemas check errors:
    meson-gxbb-odroidc2.dt.yaml: gpio-regulator-tf_io: states:0: Additional items are not
allowed (1800000, 1 were unexpected)
    meson-gxbb-odroidc2.dt.yaml: gpio-regulator-tf_io: states:0: [3300000, 0, 1800000, 1] is
too long
    meson-gxbb-nexbox-a95x.dt.yaml: gpio-regulator: states:0: Additional items are not
allowed (3300000, 1 were unexpected)
    meson-gxbb-nexbox-a95x.dt.yaml: gpio-regulator: states:0: [1800000, 0, 3300000, 1] is
```

```
too long
    meson-gxbb-p200.dt.yaml: gpio-regulator: states:0: Additional items are not allowed
(3300000, 1 were unexpected)
    meson-gxbb-p200.dt.yaml: gpio-regulator: states:0: [1800000, 0, 3300000, 1] is too long
    meson-gxl-s905x-hwacom-amazetv.dt.yaml: gpio-regulator: states:0: Additional items are
not allowed (3300000, 1 were unexpected)
    meson-gxl-s905x-hwacom-amazetv.dt.yaml: gpio-regulator: states:0: [1800000, 0, 3300000,
1] is too long
    meson-gxbb-p201.dt.yaml: gpio-regulator: states:0: Additional items are not allowed
(3300000, 1 were unexpected)
    meson-gxbb-p201.dt.yaml: gpio-regulator: states:0: [1800000, 0, 3300000, 1] is too long
    meson-g12b-odroid-n2.dt.yaml: gpio-regulator-tf_io: states:0: Additional items are not
allowed (1800000, 1 were unexpected)
    meson-g12b-odroid-n2.dt.yaml: gpio-regulator-tf_io: states:0: [3300000, 0, 1800000, 1]
is too long
    meson-gxl-s905x-nexbox-a95x.dt.yaml: gpio-regulator: states:0: Additional items are not
allowed (3300000, 1 were unexpected)
    meson-gxl-s905x-nexbox-a95x.dt.yaml: gpio-regulator: states:0: [1800000, 0, 3300000, 1]
is too long

    Signed-off-by: Neil Armstrong <narmstrong@baylibre.com>
    Reviewed-by: Martin Blumenstingl <martin.blumenstingl@googlemail.com>
    Signed-off-by: Kevin Hilman <khilman@baylibre.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 16c9a783d6d440a32e41e70767f58cf91f62ad3a
Author: Douglas RAILLARD <douglas.raillard@arm.com>
Date:   Wed Aug 7 16:33:40 2019 +0100

    sched/cpufreq: Align trace event behavior of fast switching

    [ Upstream commit 77c84dd1881d0f0176cb678d770bfbda26c54390 ]

    Fast switching path only emits an event for the CPU of interest, whereas the
    regular path emits an event for all the CPUs that had their frequency changed,
    i.e. all the CPUs sharing the same policy.

    With the current behavior, looking at cpu_frequency event for a given CPU that
    is using the fast switching path will not give the correct frequency signal.

    Signed-off-by: Douglas RAILLARD <douglas.raillard@arm.com>
    Acked-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit fd942272673cb0df5154d35eade60407f77fabd4
Author: Al Stone <ahs3@redhat.com>
Date:   Tue Aug 27 18:21:20 2019 -0600

    ACPI / CPPC: do not require the _PSD method

    [ Upstream commit 4c4cdc4c63853fee48c02e25c8605fb65a6c9924 ]

    According to the ACPI 6.3 specification, the _PSD method is optional
    when using CPPC.  The underlying assumption is that each CPU can change
    frequency independently from all other CPUs; _PSD is provided to tell
    the OS that some processors can NOT do that.

    However, the acpi_get_psd() function returns ENODEV if there is no _PSD
    method present, or an ACPI error status if an error occurs when evaluating
    _PSD, if present.  This makes _PSD mandatory when using CPPC, in violation
    of the specification, and only on Linux.

    This has forced some firmware writers to provide a dummy _PSD, even though
    it is irrelevant, but only because Linux requires it; other OSPMs follow
```

```
    the spec.  We really do not want to have OS specific ACPI tables, though.

    So, correct acpi_get_psd() so that it does not return an error if there
    is no _PSD method present, but does return a failure when the method can
    not be executed properly.  This allows _PSD to be optional as it should
    be.

    Signed-off-by: Al Stone <ahs3@redhat.com>
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b60548f72c1c43d459468b3aa6c61f4a84dd8f91
Author: Katsuhiro Suzuki <katsuhiro@katsuster.net>
Date:   Tue Aug 27 00:38:59 2019 +0900

    ASoC: es8316: fix headphone mixer volume table

    [ Upstream commit f972d02fee2496024cfd6f59021c9d89d54922a6 ]

    This patch fix setting table of Headphone mixer volume.
    Current code uses 4 ... 7 values but these values are prohibited.

    Correct settings are the following:
      0000 -12dB
      0001 -10.5dB
      0010 -9dB
      0011 -7.5dB
      0100 -6dB
      1000 -4.5dB
      1001 -3dB
      1010 -1.5dB
      1011 0dB

    Signed-off-by: Katsuhiro Suzuki <katsuhiro@katsuster.net>
    Reviewed-by: Daniel Drake <drake@endlessm.com>
    Link: https://lore.kernel.org/r/20190826153900.25969-1-katsuhiro@katsuster.net
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0df50dee08829d3cc1599a0e2722fae01eaf2563
Author: Dan Murphy <dmurphy@ti.com>
Date:   Tue Aug 20 14:53:05 2019 -0500

    leds: lm3532: Fixes for the driver for stability

    [ Upstream commit 6559ac32998248182572e1ccae79dc2eb40ac7c6 ]

    Fixed misspelled words, added error check during probe
    on the init of the registers, and fixed ALS/I2C control
    mode.

    Fixes: bc1b8492c764 ("leds: lm3532: Introduce the lm3532 LED driver")
    Reported-by: Pavel Machek <pavel@ucw.cz>
    Signed-off-by: Dan Murphy <dmurphy@ti.com>
    Acked-by: Pavel Machek <pavel@ucw.cz>
    Signed-off-by: Jacek Anaszewski <jacek.anaszewski@gmail.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit adc9850a86636abcf043e7aa709779a53e9d30c0
Author: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
Date:   Thu Aug 22 11:55:07 2019 -0300

    media: ov9650: add a sanity check

    [ Upstream commit 093347abc7a4e0490e3c962ecbde2dc272a8f708 ]
```

```
    As pointed by cppcheck:

            [drivers/media/i2c/ov9650.c:706]: (error) Shifting by a negative value is
 undefined behaviour
            [drivers/media/i2c/ov9650.c:707]: (error) Shifting by a negative value is
 undefined behaviour
            [drivers/media/i2c/ov9650.c:721]: (error) Shifting by a negative value is
 undefined behaviour

    Prevent mangling with gains with invalid values.

    As pointed by Sylvester, this should never happen in practice,
    as min value of V4L2_CID_GAIN control is 16 (gain is always >= 16
    and m is always >= 0), but it is too hard for a static analyzer
    to get this, as the logic with validates control min/max is
    elsewhere inside V4L2 core.

    Reviewed-by: Sylwester Nawrocki <s.nawrocki@samsung.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit a1b7fee84d2f58fc35dc09aac246ebeaa38cd4c9
 Author: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
 Date:   Thu Aug 22 11:46:40 2019 -0300

    media: aspeed-video: address a protential usage of an unitialized var

    [ Upstream commit 31b8b0bd6e55c3ea5a08bb8141fa5d3c90600e3b ]

    While this might not occur in practice, if the device is doing
    the right thing, it would be teoretically be possible to have
    both hsync_counter and vsync_counter negatives.

    If this ever happen, ctrl will be undefined, but the driver
    will still call:

            aspeed_video_update(video, VE_CTRL, 0, ctrl);

    Change the code to prevent this to happen.

    This was warned by cppcheck:

            [drivers/media/platform/aspeed-video.c:653]: (error) Uninitialized variable:
 ctrl

    Reviewed-by: Eddie James <eajames@linux.ibm.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit f670350e2595e20cd30029e7be68182238161136
 Author: Gustavo A. R. Silva <gustavo@embeddedor.com>
 Date:   Mon Apr 8 11:27:48 2019 -0500

    perf script: Fix memory leaks in list_scripts()

    [ Upstream commit 3b4acbb92dbda4829e021e5c6d5410658849fa1c ]

    In case memory resources for *buf* and *paths* were allocated, jump to
    *out* and release them before return.

    Signed-off-by: Gustavo A. R. Silva <gustavo@embeddedor.com>
    Cc: Alexander Shishkin <alexander.shishkin@linux.intel.com>
    Cc: Andi Kleen <ak@linux.intel.com>
    Cc: Gustavo A. R. Silva <gustavo@embeddedor.com>
```

```
    Cc: Jiri Olsa <jolsa@redhat.com>
    Cc: Namhyung Kim <namhyung@kernel.org>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Addresses-Coverity-ID: 1444328 ("Resource leak")
    Fixes: 6f3da20e151f ("perf report: Support builtin perf script in scripts menu")
    Link: http://lkml.kernel.org/r/20190408162748.GA21008@embeddedor
    Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3319131b8980a1a54545bab0ffdd549c9c08add5
Author: Andi Kleen <ak@linux.intel.com>
Date:   Fri Aug 23 14:03:38 2019 -0700

    perf report: Fix --ns time sort key output

    [ Upstream commit 3dab6ac080dcd7f71cb9ceb84ad7dafecd6f7c07 ]

    If the user specified --ns, the column to print the sort time stamp
    wasn't wide enough to actually print the full nanoseconds.

    Widen the time key column width when --ns is specified.

    Before:

      % perf record -a sleep 1
      % perf report --sort time,overhead,symbol --stdio --ns
      ...
           2.39%  187851.10000  [k] smp_call_function_single   -      -
           1.53%  187851.10000  [k] intel_idle                 -      -
           0.59%  187851.10000  [.] __wcscmp_ifunc             -      -
           0.33%  187851.10000  [.] 0000000000000000           -      -
           0.28%  187851.10000  [k] cpuidle_enter_state        -      -

    After:

      % perf report --sort time,overhead,symbol --stdio --ns
      ...
           2.39%  187851.100000000  [k] smp_call_function_single   -      -
           1.53%  187851.100000000  [k] intel_idle                 -      -
           0.59%  187851.100000000  [.] __wcscmp_ifunc             -      -
           0.33%  187851.100000000  [.] 0000000000000000           -      -
           0.28%  187851.100000000  [k] cpuidle_enter_state        -      -

    Signed-off-by: Andi Kleen <ak@linux.intel.com>
    Tested-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Cc: Jiri Olsa <jolsa@kernel.org>
    Link: http://lkml.kernel.org/r/20190823210338.12360-2-andi@firstfloor.org
    Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 54b4531060ea9570bb530ffd21fc6491f94be321
Author: Benjamin Peterson <benjamin@python.org>
Date:   Thu Aug 22 20:36:25 2019 -0700

    perf trace beauty ioctl: Fix off-by-one error in cmd->string table

    [ Upstream commit b92675f4a9c02dd78052645597dac9e270679ddf ]

    While tracing a program that calls isatty(3), I noticed that strace
    reported TCGETS for the request argument of the underlying ioctl(2)
    syscall while perf trace reported TCSETS. strace is corrrect. The bug in
    perf was due to the tty ioctl beauty table starting at 0x5400 rather
    than 0x5401.

    Committer testing:
```

```
      Using augmented_raw_syscalls.o and settings to make 'perf trace'
      use strace formatting, i.e. with this in ~/.perfconfig

      # cat ~/.perfconfig
      [trace]
            add_events =
/home/acme/git/linux/tools/perf/examples/bpf/augmented_raw_syscalls.c
            show_zeros = yes
            show_duration = no
            no_inherit = yes
            show_timestamp = no
            show_arg_names = no
            args_alignment = 40
            show_prefix = yes

      # strace -e ioctl stty > /dev/null
      ioctl(0, TCGETS, {B38400 opost isig icanon echo ...}) = 0
      ioctl(1, TIOCGWINSZ, 0x7fff8a9b0860)    = -1 ENOTTY (Inappropriate ioctl for device)
      ioctl(1, TCGETS, 0x7fff8a9b0540)        = -1 ENOTTY (Inappropriate ioctl for device)
      +++ exited with 0 +++
      #

    Before:

      # perf trace -e ioctl stty > /dev/null
      ioctl(0, TCSETS, 0x7fff2cf79f20)        = 0
      ioctl(1, TIOCSWINSZ, 0x7fff2cf79f40)    = -1 ENOTTY (Inappropriate ioctl for device)
      ioctl(1, TCSETS, 0x7fff2cf79c20)        = -1 ENOTTY (Inappropriate ioctl for device)
      #

    After:

      # perf trace -e ioctl stty > /dev/null
      ioctl(0, TCGETS, 0x7ffed0763920)        = 0
      ioctl(1, TIOCGWINSZ, 0x7ffed0763940)    = -1 ENOTTY (Inappropriate ioctl for device)
      ioctl(1, TCGETS, 0x7ffed0763620)        = -1 ENOTTY (Inappropriate ioctl for device)
      #

    Signed-off-by: Benjamin Peterson <benjamin@python.org>
    Tested-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Cc: Alexander Shishkin <alexander.shishkin@linux.intel.com>
    Cc: Jiri Olsa <jolsa@redhat.com>
    Cc: Namhyung Kim <namhyung@kernel.org>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Fixes: 1cc47f2d46206d67285aea0ca7e8450af571da13 ("perf trace beauty ioctl: Improve 'cmd'
beautifier")
    Link: http://lkml.kernel.org/r/20190823033625.18814-1-benjamin@python.org
    Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5b3eea04789f3e2523dd8ec01f5326ee5e14a1ba
Author: Maciej S. Szmigiero <mail@maciej.szmigiero.name>
Date:   Tue Aug 20 19:05:55 2019 -0300

    media: saa7134: fix terminology around saa7134_i2c_eeprom_md7134_gate()

    [ Upstream commit 9d802222a3405599d6e1984d9324cddf592ea1f4 ]

    saa7134_i2c_eeprom_md7134_gate() function and the associated comment uses
    an inverted i2c gate open / closed terminology.
    Let's fix this.

    Signed-off-by: Maciej S. Szmigiero <mail@maciej.szmigiero.name>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
```

```
    [hverkuil-cisco@xs4all.nl: fix alignment checkpatch warning]
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5d4eec9b1029e306c93ef778fbc91b5a729745c5
Author: Wenwen Wang <wenwen@cs.uga.edu>
Date:   Sat Aug 17 02:27:46 2019 -0300

    media: cpia2_usb: fix memory leaks

    [ Upstream commit 1c770f0f52dca1a2323c594f01f5ec6f1dddc97f ]

    In submit_urbs(), 'cam->sbuf[i].data' is allocated through kmalloc_array().
    However, it is not deallocated if the following allocation for urbs fails.
    To fix this issue, free 'cam->sbuf[i].data' if usb_alloc_urb() fails.

    Signed-off-by: Wenwen Wang <wenwen@cs.uga.edu>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit be5ca71813bdfd4f7044dd9b51c9744e771843f2
Author: Wenwen Wang <wenwen@cs.uga.edu>
Date:   Sun Aug 18 02:40:14 2019 -0300

    media: saa7146: add cleanup in hexium_attach()

    [ Upstream commit 42e64117d3b4a759013f77bbcf25ab6700e55de7 ]

    If saa7146_register_device() fails, no cleanup is executed, leading to
    memory/resource leaks. To fix this issue, perform necessary cleanup work
    before returning the error.

    Signed-off-by: Wenwen Wang <wenwen@cs.uga.edu>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 21f618b96d31cc7f2aa2782048d4d150c8ac89ef
Author: Hans Verkuil <hverkuil-cisco@xs4all.nl>
Date:   Fri Aug 23 08:12:59 2019 -0300

    media: cec-notifier: clear cec_adap in cec_notifier_unregister

    [ Upstream commit 14d5511691e5290103bc480998bc322e68f139d4 ]

    If cec_notifier_cec_adap_unregister() is called before
    cec_unregister_adapter() then everything is OK (and this is the
    case today). But if it is the other way around, then
    cec_notifier_unregister() is called first, and that doesn't
    set n->cec_adap to NULL.

    So if e.g. cec_notifier_set_phys_addr() is called after
    cec_notifier_unregister() but before cec_unregister_adapter()
    then n->cec_adap points to an unregistered and likely deleted
    cec adapter. So just set n->cec_adap->notifier and n->cec_adap
    to NULL for rubustness.

    Eventually cec_notifier_unregister will disappear and this will
    be simplified substantially.

    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit d3d3281fcba97bfc0942451eca28b40ee90e58b3
Author: Kamil Konieczny <k.konieczny@partner.samsung.com>
Date:   Wed Aug 7 15:38:35 2019 +0200

    PM / devfreq: exynos-bus: Correct clock enable sequence

    [ Upstream commit 2c2b20e0da89c76759ee28c6824413ab2fa3bfc6 ]

    Regulators should be enabled before clocks to avoid h/w hang. This
    require change in exynos_bus_probe() to move exynos_bus_parse_of()
    after exynos_bus_parent_parse_of() and change in error handling.
    Similar change is needed in exynos_bus_exit() where clock should be
    disabled before regulators.

    Signed-off-by: Kamil Konieczny <k.konieczny@partner.samsung.com>
    Acked-by: Chanwoo Choi <cw00.choi@samsung.com>
    Signed-off-by: MyungJoo Ham <myungjoo.ham@samsung.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b7c47578136850aa5951654763ca977ae4c4e3e4
Author: Leonard Crestez <leonard.crestez@nxp.com>
Date:   Thu Aug 8 19:54:08 2019 +0300

    PM / devfreq: passive: Use non-devm notifiers

    [ Upstream commit 0ef7c7cce43f6ecc2b96d447e69b2900a9655f7c ]

    The devfreq passive governor registers and unregisters devfreq
    transition notifiers on DEVFREQ_GOV_START/GOV_STOP using devm wrappers.

    If devfreq itself is registered with devm then a warning is triggered on
    rmmod from devm_devfreq_unregister_notifier. Call stack looks like this:

            devm_devfreq_unregister_notifier+0x30/0x40
            devfreq_passive_event_handler+0x4c/0x88
            devfreq_remove_device.part.8+0x6c/0x9c
            devm_devfreq_dev_release+0x18/0x20
            release_nodes+0x1b0/0x220
            devres_release_all+0x78/0x84
            device_release_driver_internal+0x100/0x1c0
            driver_detach+0x4c/0x90
            bus_remove_driver+0x7c/0xd0
            driver_unregister+0x2c/0x58
            platform_driver_unregister+0x10/0x18
            imx_devfreq_platdrv_exit+0x14/0xd40 [imx_devfreq]

    This happens because devres_release_all will first remove all the nodes
    into a separate todo list so the nested devres_release from
    devm_devfreq_unregister_notifier won't find anything.

    Fix the warning by calling the non-devm APIS for frequency notification.
    Using devm wrappers is not actually useful for a governor anyway: it
    relies on the devfreq core to correctly match the GOV_START/GOV_STOP
    notifications.

    Fixes: 996133119f57 ("PM / devfreq: Add new passive governor")
    Signed-off-by: Leonard Crestez <leonard.crestez@nxp.com>
    Acked-by: Chanwoo Choi <cw00.choi@samsung.com>
    Signed-off-by: MyungJoo Ham <myungjoo.ham@samsung.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a9d93f21d4a8807648df6ac0d4281171d5d73c29
Author: Masahiro Yamada <yamada.masahiro@socionext.com>
Date:   Fri Aug 23 11:58:08 2019 +0900
```

```
    ARM: OMAP2+: move platform-specific asm-offset.h to arch/arm/mach-omap2

    [ Upstream commit ccf4975dca233b1d6a74752d6ab35c239edc0d58 ]

    <generated/ti-pm-asm-offsets.h> is only generated and included by
    arch/arm/mach-omap2/, so it does not need to reside in the globally
    visible include/generated/.

    I renamed it to arch/arm/mach-omap2/pm-asm-offsets.h since the prefix
    'ti-' is just redundant in mach-omap2/.

    My main motivation of this change is to avoid the race condition for
    the parallel build (-j) when CONFIG_IKHEADERS is enabled.

    When it is enabled, all the headers under include/ are archived into
    kernel/kheaders_data.tar.xz and exposed in the sysfs.

    In the parallel build, we have no idea in which order files are built.

     - If ti-pm-asm-offsets.h is built before kheaders_data.tar.xz,
       the header will be included in the archive. Probably nobody will
       use it, but it is harmless except that it will increase the archive
       size needlessly.

     - If kheaders_data.tar.xz is built before ti-pm-asm-offsets.h,
       the header will not be included in the archive. However, in the next
       build, the archive will be re-generated to include the newly-found
       ti-pm-asm-offsets.h. This is not nice from the build system point
       of view.

     - If ti-pm-asm-offsets.h and kheaders_data.tar.xz are built at the
       same time, the corrupted header might be included in the archive,
       which does not look nice either.

    This commit fixes the race.

    Signed-off-by: Masahiro Yamada <yamada.masahiro@socionext.com>
    Tested-by: Keerthy <j-keerthy@ti.com>
    Signed-off-by: Tony Lindgren <tony@atomide.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6351fe2eeb0b77def3074bf1ecf787912766638a
Author: Ezequiel Garcia <ezequiel@collabora.com>
Date:   Fri Jun 21 18:39:49 2019 -0300

    PM / devfreq: Fix kernel oops on governor module load

    [ Upstream commit 7544fd7f384591038646d3cd9efb311ab4509e24 ]

    A bit unexpectedly (but still documented), request_module may
    return a positive value, in case of a modprobe error.
    This is currently causing issues in the devfreq framework.

    When a request_module exits with a positive value, we currently
    return that via ERR_PTR. However, because the value is positive,
    it's not a ERR_VALUE proper, and is therefore treated as a
    valid struct devfreq_governor pointer, leading to a kernel oops.

    Fix this by returning -EINVAL if request_module returns a positive
    value.

    Fixes: b53b0128052ff ("PM / devfreq: Fix static checker warning in
try_then_request_governor")
    Signed-off-by: Ezequiel Garcia <ezequiel@collabora.com>
    Reviewed-by: Chanwoo Choi <cw00.choi@samsung.com>
```

```
    Signed-off-by: MyungJoo Ham <myungjoo.ham@samsung.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 877b352ca0f1a03c64dfa2ba80567eb775ad5203
Author: Geert Uytterhoeven <geert+renesas@glider.be>
Date:   Mon Aug 19 18:54:02 2019 +0200

    soc: renesas: Enable ARM_ERRATA_754322 for affected Cortex-A9

    [ Upstream commit 2eced4607a1e6f51f928ae3e521fe02be5cb7d23 ]

    ARM Erratum 754322 affects Cortex-A9 revisions r2p* and r3p*.

    Automatically enable support code to mitigate the erratum when compiling
    a kernel for any of the affected Renesas SoCs:
      - RZ/A1: r3p0,
      - R-Mobile A1: r2p4,
      - R-Car M1A: r2p2-00rel0,
      - R-Car H1: r3p0,
      - SH-Mobile AG5: r2p2.

    EMMA Mobile EV2 (r1p3) and RZ/A2 (r4p1) are not affected.

    Signed-off-by: Geert Uytterhoeven <geert+renesas@glider.be>
    Reviewed-by: Simon Horman <horms+renesas@verge.net.au>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8964a11cfd123215f46ed988788f4a50d58951cc
Author: Geert Uytterhoeven <geert+renesas@glider.be>
Date:   Fri Aug 9 15:43:07 2019 +0200

    soc: renesas: rmobile-sysc: Set GENPD_FLAG_ALWAYS_ON for always-on domain

    [ Upstream commit af0bc634728c0bc6a3f66f911f227d5c6396db88 ]

    Currently the R-Mobile "always-on" PM Domain is implemented by returning
    -EBUSY from the generic_pm_domain.power_off() callback, and doing
    nothing in the generic_pm_domain.power_on() callback.  However, this
    means the PM Domain core code is not aware of the semantics of this
    special domain, leading to boot warnings like the following on
    SH/R-Mobile SoCs:

        sh_cmt e6130000.timer: PM domain c5 will not be powered off

    Fix this by making the always-on nature of the domain explicit instead,
    by setting the GENPD_FLAG_ALWAYS_ON flag.  This removes the need for the
    domain to provide power control callbacks.

    Signed-off-by: Geert Uytterhoeven <geert+renesas@glider.be>
    Reviewed-by: Simon Horman <horms+renesas@verge.net.au>
    Reviewed-by: Ulf Hansson <ulf.hansson@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 773a6805bf48b6e461c2b720522fac39dcd57954
Author: Masahiro Yamada <yamada.masahiro@socionext.com>
Date:   Fri Aug 23 11:43:45 2019 +0900

    ARM: at91: move platform-specific asm-offset.h to arch/arm/mach-at91

    [ Upstream commit 9fac85a6db8999922f2cd92dfe2e83e063b31a94 ]

    <generated/at91_pm_data-offsets.h> is only generated and included by
    arch/arm/mach-at91/, so it does not need to reside in the globally
    visible include/generated/.
```

```
    I renamed it to arch/arm/mach-at91/pm_data-offsets.h since the prefix
    'at91_' is just redundant in mach-at91/.

    My main motivation of this change is to avoid the race condition for
    the parallel build (-j) when CONFIG_IKHEADERS is enabled.

    When it is enabled, all the headers under include/ are archived into
    kernel/kheaders_data.tar.xz and exposed in the sysfs.

    In the parallel build, we have no idea in which order files are built.

     - If at91_pm_data-offsets.h is built before kheaders_data.tar.xz,
       the header will be included in the archive. Probably nobody will
       use it, but it is harmless except that it will increase the archive
       size needlessly.

     - If kheaders_data.tar.xz is built before at91_pm_data-offsets.h,
       the header will not be included in the archive. However, in the next
       build, the archive will be re-generated to include the newly-found
       at91_pm_data-offsets.h. This is not nice from the build system point
       of view.

     - If at91_pm_data-offsets.h and kheaders_data.tar.xz are built at the
       same time, the corrupted header might be included in the archive,
       which does not look nice either.

    This commit fixes the race.

    Signed-off-by: Masahiro Yamada <yamada.masahiro@socionext.com>
    Link: https://lore.kernel.org/r/20190823024346.591-1-yamada.masahiro@socionext.com
    Signed-off-by: Alexandre Belloni <alexandre.belloni@bootlin.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 15cc220adfc9eaa1adc94a55207f20acb80b5f78
Author: Yazen Ghannam <yazen.ghannam@amd.com>
Date:   Thu Aug 22 00:00:00 2019 +0000

    EDAC/amd64: Decode syndrome before translating address

    [ Upstream commit 8a2eaab7daf03b23ac902481218034ae2fae5e16 ]

    AMD Family 17h systems currently require address translation in order to
    report the system address of a DRAM ECC error. This is currently done
    before decoding the syndrome information. The syndrome information does
    not depend on the address translation, so the proper EDAC csrow/channel
    reporting can function without the address. However, the syndrome
    information will not be decoded if the address translation fails.

    Decode the syndrome information before doing the address translation.
    The syndrome information is architecturally defined in MCA_SYND and can
    be considered robust. The address translation is system-specific and may
    fail on newer systems without proper updates to the translation
    algorithm.

    Fixes: 713ad54675fd ("EDAC, amd64: Define and register UMC error decode function")
    Signed-off-by: Yazen Ghannam <yazen.ghannam@amd.com>
    Signed-off-by: Borislav Petkov <bp@suse.de>
    Cc: "linux-edac@vger.kernel.org" <linux-edac@vger.kernel.org>
    Cc: James Morse <james.morse@arm.com>
    Cc: Mauro Carvalho Chehab <mchehab@kernel.org>
    Cc: Tony Luck <tony.luck@intel.com>
    Link: https://lkml.kernel.org/r/20190821235938.118710-6-Yazen.Ghannam@amd.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 66b1ee884488388b3e3a40ec4ebd88fa0af7c368
```

```
Author: Yazen Ghannam <yazen.ghannam@amd.com>
Date:   Wed Aug 21 23:59:56 2019 +0000

    EDAC/amd64: Recognize DRAM device type ECC capability

    [ Upstream commit f8be8e5680225ac9caf07d4545f8529b7395327f ]

    AMD Family 17h systems support x4 and x16 DRAM devices. However, the
    device type is not checked when setting mci.edac_ctl_cap.

    Set the appropriate capability flag based on the device type.

    Default to x8 DRAM device when neither the x4 or x16 bits are set.

     [ bp: reverse cpk_en check to save an indentation level. ]

    Fixes: 2d09d8f301f5 ("EDAC, amd64: Determine EDAC MC capabilities on Fam17h")
    Signed-off-by: Yazen Ghannam <yazen.ghannam@amd.com>
    Signed-off-by: Borislav Petkov <bp@suse.de>
    Cc: "linux-edac@vger.kernel.org" <linux-edac@vger.kernel.org>
    Cc: James Morse <james.morse@arm.com>
    Cc: Mauro Carvalho Chehab <mchehab@kernel.org>
    Cc: Tony Luck <tony.luck@intel.com>
    Link: https://lkml.kernel.org/r/20190821235938.118710-3-Yazen.Ghannam@amd.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 01625bde6f4e2e83c9a275447ee9aa5468a9dee5
Author: Gerald BAEZA <gerald.baeza@st.com>
Date:   Thu Aug 22 09:07:01 2019 +0000

    libperf: Fix alignment trap with xyarray contents in 'perf stat'

    [ Upstream commit d9c5c083416500e95da098c01be092b937def7fa ]

    Following the patch 'perf stat: Fix --no-scale', an alignment trap
    happens in process_counter_values() on ARMv7 platforms due to the
    attempt to copy non 64 bits aligned double words (pointed by 'count')
    via a NEON vectored instruction ('vld1' with 64 bits alignment
    constraint).

    This patch sets a 64 bits alignment constraint on 'contents[]' field in
    'struct xyarray' since the 'count' pointer used above points to such a
    structure.

    Signed-off-by: Gerald Baeza <gerald.baeza@st.com>
    Cc: Alexander Shishkin <alexander.shishkin@linux.intel.com>
    Cc: Alexandre Torgue <alexandre.torgue@st.com>
    Cc: Andi Kleen <ak@linux.intel.com>
    Cc: Jiri Olsa <jolsa@redhat.com>
    Cc: Mathieu Poirier <mathieu.poirier@linaro.org>
    Cc: Namhyung Kim <namhyung@kernel.org>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Link: http://lkml.kernel.org/r/1566464769-16374-1-git-send-email-gerald.baeza@st.com
    Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 57da4b1bbd689eb5534a8225395842cb7a559c60
Author: Anson Huang <Anson.Huang@nxp.com>
Date:   Sun Aug 18 02:32:22 2019 -0400

    cpufreq: imx-cpufreq-dt: Add i.MX8MN support

    [ Upstream commit 75c000c4bcbe2b0eb82baf90c7dd75c7380cc3fd ]

    i.MX8MN has different speed grading definition as below, it has 4 bits
```

```
    to define speed grading, add support for it.

     SPEED_GRADE[3:0]    MHz
        0000            2300
        0001            2200
        0010            2100
        0011            2000
        0100            1900
        0101            1800
        0110            1700
        0111            1600
        1000            1500
        1001            1400
        1010            1300
        1011            1200
        1100            1100
        1101            1000
        1110             900
        1111             800

    Signed-off-by: Anson Huang <Anson.Huang@nxp.com>
    Reviewed-by: Leonard Crestez <leonard.crestez@nxp.com>
    Signed-off-by: Viresh Kumar <viresh.kumar@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7fc00c591bef3b2bcccf9cf626756a5c2d36fe33
Author: Yazen Ghannam <yazen.ghannam@amd.com>
Date:   Wed Aug 21 23:59:55 2019 +0000

    EDAC/amd64: Support more than two controllers for chip selects handling

    [ Upstream commit d971e28e2ce4696fcc32998c8aced5e47701fffe ]

    The struct chip_select array that's used for saving chip select bases
    and masks is fixed at length of two. There should be one struct
    chip_select for each controller, so this array should be increased to
    support systems that may have more than two controllers.

    Increase the size of the struct chip_select array to eight, which is the
    largest number of controllers per die currently supported on AMD
    systems.

    Fix number of DIMMs and Chip Select bases/masks on Family17h, because
    AMD Family 17h systems support 2 DIMMs, 4 CS bases, and 2 CS masks per
    channel.

    Also, carve out the Family 17h+ reading of the bases/masks into a
    separate function. This effectively reverts the original bases/masks
    reading code to before Family 17h support was added.

    Signed-off-by: Yazen Ghannam <yazen.ghannam@amd.com>
    Signed-off-by: Borislav Petkov <bp@suse.de>
    Cc: "linux-edac@vger.kernel.org" <linux-edac@vger.kernel.org>
    Cc: James Morse <james.morse@arm.com>
    Cc: Mauro Carvalho Chehab <mchehab@kernel.org>
    Cc: Tony Luck <tony.luck@intel.com>
    Link: https://lkml.kernel.org/r/20190821235938.118710-2-Yazen.Ghannam@amd.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 202be5d6e46f682b9d1d79cd4dc6ab726e62ef1c
Author: Wenwen Wang <wenwen@cs.uga.edu>
Date:   Sun Aug 18 00:45:40 2019 -0300

    media: dvb-core: fix a memory leak bug
```

```
    [ Upstream commit fcd5ce4b3936242e6679875a4d3c3acfc8743e15 ]

    In dvb_create_media_entity(), 'dvbdev->entity' is allocated through
    kzalloc(). Then, 'dvbdev->pads' is allocated through kcalloc(). However, if
    kcalloc() fails, the allocated 'dvbdev->entity' is not deallocated, leading
    to a memory leak bug. To fix this issue, free 'dvbdev->entity' before
    returning -ENOMEM.

    Signed-off-by: Wenwen Wang <wenwen@cs.uga.edu>
    Signed-off-by: Sean Young <sean@mess.org>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 263321b012d27153f7923aa929822098d194b235
Author: Thomas Gleixner <tglx@linutronix.de>
Date:   Mon Aug 19 16:31:46 2019 +0200

    posix-cpu-timers: Sanitize bogus WARNONS

    [ Upstream commit 692117c1f7a6770ed41dd8f277cd9fed1dfb16f1 ]

    Warning when p == NULL and then proceeding and dereferencing p does not
    make any sense as the kernel will crash with a NULL pointer dereference
    right away.

    Bailing out when p == NULL and returning an error code does not cure the
    underlying problem which caused p to be NULL. Though it might allow to
    do proper debugging.

    Same applies to the clock id check in set_process_cpu_timer().

    Clean them up and make them return without trying to do further damage.

    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Reviewed-by: Frederic Weisbecker <frederic@kernel.org>
    Link: https://lkml.kernel.org/r/20190819143801.846497772@linutronix.de
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2e279d556e85801b1d7c5021429d93a282eef1f2
Author: Sean Young <sean@mess.org>
Date:   Sat Aug 3 00:12:03 2019 -0300

    media: dvb-frontends: use ida for pll number

    [ Upstream commit c268e7adea52be0093de1164c425f3c8d8927770 ]

    KASAN: global-out-of-bounds Read in dvb_pll_attach

    Syzbot reported global-out-of-bounds Read in dvb_pll_attach, while
    accessing id[dvb_pll_devcount], because dvb_pll_devcount was 65,
    that is more than size of 'id' which is DVB_PLL_MAX(64).

    Rather than increasing dvb_pll_devcount every time, use ida so that
    numbers are allocated correctly. This does mean that no more than
    64 devices can be attached at the same time, but this is more than
    sufficient.

    usb 1-1: dvb_usb_v2: will pass the complete MPEG2 transport stream to the
    software demuxer
    dvbdev: DVB: registering new adapter (774 Friio White ISDB-T USB2.0)
    usb 1-1: media controller created
    dvbdev: dvb_create_media_entity: media entity 'dvb-demux' registered.
    tc90522 0-0018: Toshiba TC90522 attached.
    usb 1-1: DVB: registering adapter 0 frontend 0 (Toshiba TC90522 ISDB-T
    module)...
```

```
dvbdev: dvb_create_media_entity: media entity 'Toshiba TC90522 ISDB-T
module' registered.
==================================================================
BUG: KASAN: global-out-of-bounds in dvb_pll_attach+0x6c5/0x830
drivers/media/dvb-frontends/dvb-pll.c:798
Read of size 4 at addr ffffffff89c9e5e0 by task kworker/0:1/12

CPU: 0 PID: 12 Comm: kworker/0:1 Not tainted 5.2.0-rc6+ #13
Hardware name: Google Google Compute Engine/Google Compute Engine, BIOS
Google 01/01/2011
Workqueue: usb_hub_wq hub_event
Call Trace:
  __dump_stack lib/dump_stack.c:77 [inline]
  dump_stack+0xca/0x13e lib/dump_stack.c:113
  print_address_description+0x67/0x231 mm/kasan/report.c:188
  __kasan_report.cold+0x1a/0x32 mm/kasan/report.c:317
  kasan_report+0xe/0x20 mm/kasan/common.c:614
  dvb_pll_attach+0x6c5/0x830 drivers/media/dvb-frontends/dvb-pll.c:798
  dvb_pll_probe+0xfe/0x174 drivers/media/dvb-frontends/dvb-pll.c:877
  i2c_device_probe+0x790/0xaa0 drivers/i2c/i2c-core-base.c:389
  really_probe+0x281/0x660 drivers/base/dd.c:509
  driver_probe_device+0x104/0x210 drivers/base/dd.c:670
  __device_attach_driver+0x1c2/0x220 drivers/base/dd.c:777
  bus_for_each_drv+0x15c/0x1e0 drivers/base/bus.c:454
  __device_attach+0x217/0x360 drivers/base/dd.c:843
  bus_probe_device+0x1e4/0x290 drivers/base/bus.c:514
  device_add+0xae6/0x16f0 drivers/base/core.c:2111
  i2c_new_client_device+0x5b3/0xc40 drivers/i2c/i2c-core-base.c:778
  i2c_new_device+0x19/0x50 drivers/i2c/i2c-core-base.c:821
  dvb_module_probe+0xf9/0x220 drivers/media/dvb-core/dvbdev.c:985
  friio_tuner_attach+0x125/0x1d0 drivers/media/usb/dvb-usb-v2/gl861.c:536
  dvb_usbv2_adapter_frontend_init
drivers/media/usb/dvb-usb-v2/dvb_usb_core.c:675 [inline]
  dvb_usbv2_adapter_init drivers/media/usb/dvb-usb-v2/dvb_usb_core.c:804
[inline]
  dvb_usbv2_init drivers/media/usb/dvb-usb-v2/dvb_usb_core.c:865 [inline]
  dvb_usbv2_probe.cold+0x24dc/0x255d
drivers/media/usb/dvb-usb-v2/dvb_usb_core.c:980
  usb_probe_interface+0x305/0x7a0 drivers/usb/core/driver.c:361
  really_probe+0x281/0x660 drivers/base/dd.c:509
  driver_probe_device+0x104/0x210 drivers/base/dd.c:670
  __device_attach_driver+0x1c2/0x220 drivers/base/dd.c:777
  bus_for_each_drv+0x15c/0x1e0 drivers/base/bus.c:454
  __device_attach+0x217/0x360 drivers/base/dd.c:843
  bus_probe_device+0x1e4/0x290 drivers/base/bus.c:514
  device_add+0xae6/0x16f0 drivers/base/core.c:2111
  usb_set_configuration+0xdf6/0x1670 drivers/usb/core/message.c:2023
  generic_probe+0x9d/0xd5 drivers/usb/core/generic.c:210
  usb_probe_device+0x99/0x100 drivers/usb/core/driver.c:266
  really_probe+0x281/0x660 drivers/base/dd.c:509
  driver_probe_device+0x104/0x210 drivers/base/dd.c:670
  __device_attach_driver+0x1c2/0x220 drivers/base/dd.c:777
  bus_for_each_drv+0x15c/0x1e0 drivers/base/bus.c:454
  __device_attach+0x217/0x360 drivers/base/dd.c:843
  bus_probe_device+0x1e4/0x290 drivers/base/bus.c:514
  device_add+0xae6/0x16f0 drivers/base/core.c:2111
  usb_new_device.cold+0x8c1/0x1016 drivers/usb/core/hub.c:2534
  hub_port_connect drivers/usb/core/hub.c:5089 [inline]
  hub_port_connect_change drivers/usb/core/hub.c:5204 [inline]
  port_event drivers/usb/core/hub.c:5350 [inline]
  hub_event+0x1ada/0x3590 drivers/usb/core/hub.c:5432
  process_one_work+0x905/0x1570 kernel/workqueue.c:2269
  process_scheduled_works kernel/workqueue.c:2331 [inline]
  worker_thread+0x7ab/0xe20 kernel/workqueue.c:2417
  kthread+0x30b/0x410 kernel/kthread.c:255
```

```
       ret_from_fork+0x24/0x30 arch/x86/entry/entry_64.S:352

     The buggy address belongs to the variable:
       id+0x100/0x120

     Memory state around the buggy address:
       ffffffff89c9e480: fa fa fa fa 00 00 fa fa fa fa fa fa 00 00 00 00
       ffffffff89c9e500: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
     > ffffffff89c9e580: 00 00 00 00 00 00 00 00 00 00 00 00 fa fa fa fa
                                                             ^
       ffffffff89c9e600: 04 fa fa fa fa fa fa fa 04 fa fa fa fa fa fa fa
       ffffffff89c9e680: 04 fa fa fa fa fa fa fa 04 fa fa fa fa fa fa fa
     ==================================================================

     Reported-by: syzbot+8a8f48672560c8ca59dd@syzkaller.appspotmail.com
     Signed-off-by: Sean Young <sean@mess.org>
     Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 254030e924bd5a8424f010983f5b092df3e20a56
  Author: A Sun <as1033x@comcast.net>
  Date:   Thu Aug 15 13:41:19 2019 -0300

     media: mceusb: fix (eliminate) TX IR signal length limit

     [ Upstream commit 9fc3ce31f5bde660197f35135e90a1cced58aa2c ]

     Fix and eliminate mceusb's IR length limit for IR signals transmitted to
     the MCE IR blaster ports.

     An IR signal TX exceeding 306 pulse/space samples presently causes -EINVAL
     return error. There's no such limitation nor error with the MCE device
     hardware. And valid IR signals exist with more than 400 pulse/space for the
     control of certain appliances (eg Panasonic ACXA75C00600 air conditioner).

     The scope of this patch is limited to the mceusb driver. There are still
     IR signal TX length and time constraints that related modules of rc core
     (eg LIRC) impose, further up the driver stack.

     Changes for mceusb_tx_ir():

     Converts and sends LIRC IR pulse/space sequence to MCE device IR
     pulse/space format.

     Break long length LIRC sequence into multiple (unlimited number of) parts
     for sending to the MCE device.
     Reduce kernel stack IR buffer size: 128 (was 384)
     Increase MCE IR data packet size: 31 (was 5)
     Zero time LIRC pulse/space no longer copied to MCE IR data.
     Eliminate overwriting the source/input LIRC IR data in txbuf[].
     Eliminate -EINVAL return; return number of IR samples sent (>0) or
     MCE write error code (<0).

     New mce_write() and mce_write_callback():

     Implements synchronous blocking I/O, with timeout, for writing/sending
     data to the MCE device.

     An unlimited multipart IR signal sent to the MCE device faster than real
     time requires flow control absent with the original mce_request_packet()
     and mce_async_callback() asynchronous I/O implementation. Also absent is
     TX error feedback.

     mce_write() combines and replaces mce_request_packet() and
     mce_async_callback() with conversion to synchronous I/O.
```

```
    mce_write() returns bytes sent (>0) or MCE device write error (<0).
    Debug hex dump TX data before processing.

    Rename mce_async_out() -> mce_command_out():

    The original name is misleading with underlying synchronous I/O
    implementation. Function renamed to mce_command_out().

    Changes in mceusb_handle_command():

    Add support for MCE device error case MCE_RSP_TX_TIMEOUT
    "IR TX timeout (TX buffer underrun)"

    Changes in mceusb_dev_printdata():

    Changes support test and debug of multipart TX IR.

    Add buffer boundary information (offset and buffer size) to TX hex dump.
    Correct TX trace bug "Raw IR data, 0 pulse/space samples"
    Add trace for MCE_RSP_TX_TIMEOUT "IR TX timeout (TX buffer underrun)"

    Other changes:

    The driver's write to USB device architecture change (async to sync I/O)
    is significant so we bump DRIVER_VERSION to "1.95" (from "1.94").

    Tests:

    $ cat -n irdata1 | head -3
         1  carrier 36000
         2  pulse 6350
         3  space 6350
    $ cat -n irdata1 | tail -3
        76  pulse 6350
        77  space 6350
        78  pulse 6350
    $ ir-ctl -s irdata1

    [1549021.073612] mceusb 1-1.3:1.0: requesting 36000 HZ carrier
    [1549021.073635] mceusb 1-1.3:1.0: tx data[0]: 9f 06 01 45 (len=4 sz=4)
    [1549021.073649] mceusb 1-1.3:1.0: Request carrier of 35714 Hz (period 28us)
    [1549021.073848] mceusb 1-1.3:1.0: tx done status = 4 (wait = 100, expire = 100
(1000ms), urb->actual_length = 4, urb->status = 0)
    [1549021.074689] mceusb 1-1.3:1.0: rx data[0]: 9f 06 01 45 (len=4 sz=4)
    [1549021.074701] mceusb 1-1.3:1.0: Got carrier of 35714 Hz (period 28us)
    [1549021.102023] mceusb 1-1.3:1.0: tx data[0]: 9f 08 03 (len=3 sz=3)
    [1549021.102036] mceusb 1-1.3:1.0: Request transmit blaster mask of 0x03
    [1549021.102219] mceusb 1-1.3:1.0: tx done status = 3 (wait = 100, expire = 100
(1000ms), urb->actual_length = 3, urb->status = 0)
    [1549021.131979] mceusb 1-1.3:1.0: tx data[0]: 9e ff 7f ff 7f ff 7f ff 7f ff 7f ff 7f ff
7f ff 7f ff 7f ff 7f ff 7f ff 7f ff 7f ff 7f ff 7f 9e ff 7f ff 7f ff 7f ff 7f ff 7f ff 7f ff
7f ff 7f ff 7f ff 7f ff 7f ff 7f ff 7f ff 7f ff 7f 91 ff (len=81 sz=81)
    [1549021.131992] mceusb 1-1.3:1.0: Raw IR data, 30 pulse/space samples
    [1549021.133592] mceusb 1-1.3:1.0: tx done status = 81 (wait = 100, expire = 100
(1000ms), urb->actual_length = 81, urb->status = 0)

    Hex dumps limited to 64 bytes.
    0xff is MCE maximum time pulse, 0x7f is MCE maximum time space.

    $ cat -n irdata2 | head -3
         1  carrier 36000
         2  pulse 50
         3  space 50
    $ cat -n irdata2 | tail -3
       254  pulse 50
```

```
        255  space 50
        256  pulse 50
    $ ir-ctl -s irdata2

    [1549306.586998] mceusb 1-1.3:1.0: tx data[0]: 9f 08 03 (len=3 sz=3)
    [1549306.587015] mceusb 1-1.3:1.0: Request transmit blaster mask of 0x03
    [1549306.587252] mceusb 1-1.3:1.0: tx done status = 3 (wait = 100, expire = 100
 (1000ms), urb->actual_length = 3, urb->status = 0)
    [1549306.613275] mceusb 1-1.3:1.0: tx data[0]: 9e 81 01 81 01 81 01 81 01 81 01 81 01 81
 01 81 01 81 01 81 01 81 01 81 01 81 01 81 01 81 01 81 01 81 01 81 01 81 01 81 01 81 01 81
 01 81 01 81 01 81 01 81 01 81 01 81 01 81 01 81 01 9e 81 (len=128 sz=128)
    [1549306.613291] mceusb 1-1.3:1.0: Raw IR data, 30 pulse/space samples
    [1549306.614837] mceusb 1-1.3:1.0: tx done status = 128 (wait = 100, expire = 100
 (1000ms), urb->actual_length = 128, urb->status = 0)
    [1549306.614861] mceusb 1-1.3:1.0: tx data[0]: 9e 01 81 01 81 01 81 01 81 01 81 01 81 01
 81 01 81 01 81 01 81 01 81 01 81 01 81 01 81 01 81 01 81 01 81 01 81 01 81 01 81 01 81 01
 81 01 81 01 81 01 81 01 81 01 81 01 81 01 81 01 81 9e 01 (len=128 sz=128)
    [1549306.614869] mceusb 1-1.3:1.0: Raw IR data, 30 pulse/space samples
    [1549306.620199] mceusb 1-1.3:1.0: tx done status = 128 (wait = 100, expire = 100
 (1000ms), urb->actual_length = 128, urb->status = 0)
    [1549306.620212] mceusb 1-1.3:1.0: tx data[0]: 89 81 01 81 01 81 01 81 01 81 80 (len=11
 sz=11)
    [1549306.620221] mceusb 1-1.3:1.0: Raw IR data, 9 pulse/space samples
    [1549306.633294] mceusb 1-1.3:1.0: tx done status = 11 (wait = 98, expire = 100
 (1000ms), urb->actual_length = 11, urb->status = 0)

    Hex dumps limited to 64 bytes.
    0x81 is MCE minimum time pulse, 0x01 is MCE minimum time space.
    TX IR part 3 sz=11 shows 20msec I/O blocking delay
    (100expire - 98wait = 2jiffies)

    Signed-off-by: A Sun <as1033x@comcast.net>
    Signed-off-by: Sean Young <sean@mess.org>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ea81e37da4f5f2768ead63197bd4435a68885f1a
Author: Vasily Gorbik <gor@linux.ibm.com>
Date:   Mon Aug 12 15:52:07 2019 +0200

    s390/kasan: provide uninstrumented __strlen

    [ Upstream commit f45f7b5bdaa4828ce871cf03f7c01599a0de57a5 ]

    s390 kasan code uses sclp_early_printk to report initialization
    failures. The code doing that should not be instrumented, because kasan
    shadow memory has not been set up yet. Even though sclp_early_core.c is
    compiled with instrumentation disabled it uses strlen function, which
    is instrumented and would produce shadow memory access if used. To
    avoid that, introduce uninstrumented __strlen function to be used
    instead.

    Before commit 7e0d92f00246 ("s390/kasan: improve string/memory functions
    checks") few string functions (including strlen) were escaping kasan
    instrumentation due to usage of platform specific versions which are
    implemented in inline assembly.

    Fixes: 7e0d92f00246 ("s390/kasan: improve string/memory functions checks")
    Acked-by: Ilya Leoshkevich <iii@linux.ibm.com>
    Signed-off-by: Vasily Gorbik <gor@linux.ibm.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2dc3162e5dee47c807493a3084313758aaa48a9b
Author: James Morse <james.morse@arm.com>
Date:   Tue Aug 20 18:45:57 2019 +0100
```

```
    arm64: entry: Move ct_user_exit before any other exception

    [ Upstream commit 2671828c3ff4ffadf777f793a1f3232d6e51394a ]

    When taking an SError or Debug exception from EL0, we run the C
    handler for these exceptions before updating the context tracking
    code and unmasking lower priority interrupts.

    When booting with nohz_full lockdep tells us we got this wrong:
    | ============================
    | WARNING: suspicious RCU usage
    | 5.3.0-rc2-00010-gb4b5e9dcb11b-dirty #11271 Not tainted
    | ----------------------------
    | include/linux/rcupdate.h:643 rcu_read_unlock() used illegally wh!
    |
    | other info that might help us debug this:
    |
    |
    | RCU used illegally from idle CPU!
    | rcu_scheduler_active = 2, debug_locks = 1
    | RCU used illegally from extended quiescent state!
    | 1 lock held by a.out/432:
    |  #0: 00000000c7a79515 (rcu_read_lock){....}, at: brk_handler+0x00
    |
    | stack backtrace:
    | CPU: 1 PID: 432 Comm: a.out Not tainted 5.3.0-rc2-00010-gb4b5e9d1
    | Hardware name: ARM LTD ARM Juno Development Platform/ARM Juno De8
    | Call trace:
    |  dump_backtrace+0x0/0x140
    |  show_stack+0x14/0x20
    |  dump_stack+0xbc/0x104
    |  lockdep_rcu_suspicious+0xf8/0x108
    |  brk_handler+0x164/0x1b0
    |  do_debug_exception+0x11c/0x278
    |  el0_dbg+0x14/0x20

    Moving the ct_user_exit calls to be before do_debug_exception() means
    they are also before trace_hardirqs_off() has been updated. Add a new
    ct_user_exit_irqoff macro to avoid the context-tracking code using
    irqsave/restore before we've updated trace_hardirqs_off(). To be
    consistent, do this everywhere.

    The C helper is called enter_from_user_mode() to match x86 in the hope
    we can merge them into kernel/context_tracking.c later.

    Cc: Masami Hiramatsu <mhiramat@kernel.org>
    Fixes: 6c81fe7925cc4c42 ("arm64: enable context tracking")
    Signed-off-by: James Morse <james.morse@arm.com>
    Signed-off-by: Will Deacon <will@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 23a0276a7688a4c96ecb5f8d187ffdd771aefcab
Author: Liguang Zhang <zhangliguang@linux.alibaba.com>
Date:   Mon Jul 15 14:58:44 2019 +0800

    ACPI / APEI: Release resources if gen_pool_add() fails

    [ Upstream commit 6abc7622271dc520f241462e2474c71723638851 ]

    Destroy ghes_estatus_pool and release memory allocated via vmalloc() on
    errors in ghes_estatus_pool_init() in order to avoid memory leaks.

     [ bp: do the labels properly and with descriptive names and massage. ]
```

```
    Signed-off-by: Liguang Zhang <zhangliguang@linux.alibaba.com>
    Signed-off-by: Borislav Petkov <bp@suse.de>
    Link: https://lkml.kernel.org/r/1563173924-47479-1-git-send-email-
 zhangliguang@linux.alibaba.com
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit cd456700100dbf685ac211996a536a7e06542d81
 Author: Mike Christie <mchristi@redhat.com>
 Date:   Tue Aug 13 11:39:51 2019 -0500

    nbd: add missing config put

    [ Upstream commit 887e975c4172d0d5670c39ead2f18ba1e4ec8133 ]

    Fix bug added with the patch:

    commit 8f3ea35929a0806ad1397db99a89ffee0140822a
    Author: Josef Bacik <josef@toxicpanda.com>
    Date:   Mon Jul 16 12:11:35 2018 -0400

        nbd: handle unexpected replies better

    where if the timeout handler runs when the completion path is and we fail
    to grab the mutex in the timeout handler we will leave a config reference
    and cannot free the config later.

    Reviewed-by: Josef Bacik <josef@toxicpanda.com>
    Signed-off-by: Mike Christie <mchristi@redhat.com>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit e14408917944e7b101551a35010d3d38042e636e
 Author: Codrin Ciubotariu <codrin.ciubotariu@microchip.com>
 Date:   Tue Aug 20 19:24:09 2019 +0300

    ASoC: mchp-i2s-mcc: Fix unprepare of GCLK

    [ Upstream commit 988b59467b2b14523a266957affbe9eca3e99fc9 ]

    If hw_free() gets called after hw_params(), GCLK remains prepared,
    preventing further use of it. This patch fixes this by unpreparing the
    clock in hw_free() or if hw_params() gets an error.

    Fixes: 7e0cdf545a55 ("ASoC: mchp-i2s-mcc: add driver for I2SC Multi-Channel Controller")
    Signed-off-by: Codrin Ciubotariu <codrin.ciubotariu@microchip.com>
    Link: https://lore.kernel.org/r/20190820162411.24836-2-codrin.ciubotariu@microchip.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit 6f843cb302cd7cf6c45dec1bb42d386d428789a1
 Author: Wenwen Wang <wenwen@cs.uga.edu>
 Date:   Mon Aug 19 15:41:42 2019 -0500

    led: triggers: Fix a memory leak bug

    [ Upstream commit 60e2dde1e91ae0addb21ac380cc36ebee7534e49 ]

    In led_trigger_set(), 'event' is allocated in kasprintf(). However, it is
    not deallocated in the following execution if the label 'err_activate' or
    'err_add_groups' is entered, leading to memory leaks. To fix this issue,
    free 'event' before returning the error.

    Fixes: 52c47742f79d ("leds: triggers: send uevent when changing triggers")
    Signed-off-by: Wenwen Wang <wenwen@cs.uga.edu>
```

```
    Acked-by: Pavel Machek <pavel@ucw.cz>
    Signed-off-by: Jacek Anaszewski <jacek.anaszewski@gmail.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7f2b11071ff67680110b10ca1c0abf4fa7be326a
Author: Codrin Ciubotariu <codrin.ciubotariu@microchip.com>
Date:   Tue Aug 20 19:24:10 2019 +0300

    ASoC: mchp-i2s-mcc: Wait for RX/TX RDY only if controller is running

    [ Upstream commit 0f6fc97501b790c971b11b52a654009d21c45238 ]

    Since hw_free() can be called multiple times and not just after a stop
    trigger command, we should check whether the RX or TX ready interrupt was
    truly enabled previously. For this, we assure that the condition of the
    wait event is always true, except when RX/TX interrupts are enabled.

    Fixes: 7e0cdf545a55 ("ASoC: mchp-i2s-mcc: add driver for I2SC Multi-Channel Controller")
    Signed-off-by: Codrin Ciubotariu <codrin.ciubotariu@microchip.com>
    Link: https://lore.kernel.org/r/20190820162411.24836-3-codrin.ciubotariu@microchip.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4d48669a016b2dc4340e00c962af22cdf385ce49
Author: Maxime Ripard <maxime.ripard@bootlin.com>
Date:   Mon Aug 19 21:25:14 2019 +0200

    ASoC: sun4i-i2s: Don't use the oversample to calculate BCLK

    [ Upstream commit 7df8f9a20196072162d9dc8fe99943f2d35f23d5 ]

    The BCLK divider should be calculated using the parameters that actually
    make the BCLK rate: the number of channels, the sampling rate and the
    sample width.

    We've been using the oversample_rate previously because in the former SoCs,
    the BCLK's parent is MCLK, which in turn is being used to generate the
    oversample rate, so we end up with something like this:

    oversample = mclk_rate / sampling_rate
    bclk_div = oversample / word_size / channels

    So, bclk_div = mclk_rate / sampling_rate / word_size / channels.

    And this is actually better, since the oversampling ratio only plays a role
    because the MCLK is its parent, not because of what BCLK is supposed to be.

    Furthermore, that assumption of MCLK being the parent has been broken on
    newer SoCs, so let's use the proper formula, and have the parent rate as an
    argument.

    Fixes: 7d2993811a1e ("ASoC: sun4i-i2s: Add support for H3")
    Fixes: 21faaea1343f ("ASoC: sun4i-i2s: Add support for A83T")
    Fixes: 66ecce332538 ("ASoC: sun4i-i2s: Add compatibility with A64 codec I2S")
    Signed-off-by: Maxime Ripard <maxime.ripard@bootlin.com>
    Link: https://lore.kernel.org/r/c3595e3a9788c2ef2dcc30aa3c8c4953bb5cc249.1566242458.git-
series.maxime.ripard@bootlin.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 35dd762128a356d4898c995e484f5455cc4967da
Author: Arnaldo Carvalho de Melo <acme@redhat.com>
Date:   Tue Aug 20 11:45:17 2019 -0300

    tools headers: Fixup bitsperlong per arch includes
```

```
    [ Upstream commit 42fc2e9ef9603a7948aaa4ffd8dfb94b30294ad8 ]

    We were getting the file by luck, from one of the paths in -I, fix it to
    get it from the proper place:

      $ cd tools/include/uapi/asm/
      [acme@quaco asm]$ grep include bitsperlong.h
      #include "../../arch/x86/include/uapi/asm/bitsperlong.h"
      #include "../../arch/arm64/include/uapi/asm/bitsperlong.h"
      #include "../../arch/powerpc/include/uapi/asm/bitsperlong.h"
      #include "../../arch/s390/include/uapi/asm/bitsperlong.h"
      #include "../../arch/sparc/include/uapi/asm/bitsperlong.h"
      #include "../../arch/mips/include/uapi/asm/bitsperlong.h"
      #include "../../arch/ia64/include/uapi/asm/bitsperlong.h"
      #include "../../arch/riscv/include/uapi/asm/bitsperlong.h"
      #include "../../arch/alpha/include/uapi/asm/bitsperlong.h"
      #include <asm-generic/bitsperlong.h>
      $ ls -la ../../arch/x86/include/uapi/asm/bitsperlong.h
      ls: cannot access '../../arch/x86/include/uapi/asm/bitsperlong.h': No such file or
directory
      $ ls -la ../../../arch/*/include/uapi/asm/bitsperlong.h
      -rw-rw-r--. 1 237 ../../../arch/alpha/include/uapi/asm/bitsperlong.h
      -rw-rw-r--. 1 841 ../../../arch/arm64/include/uapi/asm/bitsperlong.h
      -rw-rw-r--. 1 966 ../../../arch/hexagon/include/uapi/asm/bitsperlong.h
      -rw-rw-r--. 1 234 ../../../arch/ia64/include/uapi/asm/bitsperlong.h
      -rw-rw-r--. 1 100 ../../../arch/microblaze/include/uapi/asm/bitsperlong.h
      -rw-rw-r--. 1 244 ../../../arch/mips/include/uapi/asm/bitsperlong.h
      -rw-rw-r--. 1 352 ../../../arch/parisc/include/uapi/asm/bitsperlong.h
      -rw-rw-r--. 1 312 ../../../arch/powerpc/include/uapi/asm/bitsperlong.h
      -rw-rw-r--. 1 353 ../../../arch/riscv/include/uapi/asm/bitsperlong.h
      -rw-rw-r--. 1 292 ../../../arch/s390/include/uapi/asm/bitsperlong.h
      -rw-rw-r--. 1 323 ../../../arch/sparc/include/uapi/asm/bitsperlong.h
      -rw-rw-r--. 1 320 ../../../arch/x86/include/uapi/asm/bitsperlong.h
      $

    Found while fixing some other problem, before it was escaping the
    tools/ chroot and using stuff in the kernel sources:

        CC       /tmp/build/perf/util/find_bit.o
    In file included from
/git/linux/tools/include/../../arch/x86/include/uapi/asm/bitsperlong.h:11,
                     from /git/linux/tools/include/uapi/asm/bitsperlong.h:3,
                     from /git/linux/tools/include/linux/bits.h:6,
                     from /git/linux/tools/include/linux/bitops.h:13,
                     from ../lib/find_bit.c:17:

      # cd /git/linux/tools/include/../../arch/x86/include/uapi/asm/
      # pwd
      /git/linux/arch/x86/include/uapi/asm
      #

    Now it is getting the one we want it to, i.e. the one inside tools/:

        CC       /tmp/build/perf/util/find_bit.o
      In file included from /git/linux/tools/arch/x86/include/uapi/asm/bitsperlong.h:11,
                       from /git/linux/tools/include/linux/bits.h:6,
                       from /git/linux/tools/include/linux/bitops.h:13,

    Cc: Adrian Hunter <adrian.hunter@intel.com>
    Cc: Jiri Olsa <jolsa@kernel.org>
    Cc: Namhyung Kim <namhyung@kernel.org>
    Link: https://lkml.kernel.org/n/tip-8f8cfqywmf6jk8a3ucr0ixhu@git.kernel.org
    Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 4fd9e745e136f2641c776fd7bc432971251d5187
Author: Michael Ellerman <mpe@ellerman.id.au>
Date:   Tue Aug 20 17:23:19 2019 +1000

    powerpc/Makefile: Always pass --synthetic to nm if supported

    [ Upstream commit 117acf5c29dd89e4c86761c365b9724dba0d9763 ]

    Back in 2004 we added logic to arch/ppc64/Makefile to pass
    the --synthetic option to nm, if it was supported by nm.

    Then in 2005 when arch/ppc64 and arch/ppc were merged, the logic to
    add --synthetic was moved inside an #ifdef CONFIG_PPC64 block within
    arch/powerpc/Makefile, and has remained there since.

    That was fine, though crufty, until recently when a change to
    init/Kconfig added a config time check that uses $(NM). On powerpc
    that leads to an infinite loop because Kconfig uses $(NM) to calculate
    some values, then the powerpc Makefile changes $(NM), which Kconfig
    notices and restarts.

    The original commit that added --synthetic simply said:
      On new toolchains we need to use nm --synthetic or we miss code
      symbols.

    And the nm man page says that the --synthetic option causes nm to:
      Include synthetic symbols in the output. These are special symbols
      created by the linker for various purposes.

    So it seems safe to always pass --synthetic if nm supports it, ie. on
    32-bit and 64-bit, it just means 32-bit kernels might have more
    symbols reported (and in practice I see no extra symbols). Making it
    unconditional avoids the #ifdef CONFIG_PPC64, which in turn avoids the
    infinite loop.

    Debugged-by: Peter Collingbourne <pcc@google.com>
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Signed-off-by: Will Deacon <will@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 20f3c328bf2c3293a813292052ba16b0544debc6
Author: Kunihiko Hayashi <hayashi.kunihiko@socionext.com>
Date:   Tue Aug 20 15:16:04 2019 +0900

    ASoC: uniphier: Fix double reset assersion when transitioning to suspend state

    [ Upstream commit c372a35550c8d60f673b20210eea58a06d6d38cb ]

    When transitioning to supend state, uniphier_aio_dai_suspend() is called
    and asserts reset lines and disables clocks.

    However, if there are two or more DAIs, uniphier_aio_dai_suspend() are
    called multiple times, and double reset assersion will cause.

    This patch defines the counter that has the number of DAIs at first, and
    whenever uniphier_aio_dai_suspend() are called, it decrements the
    counter. And only if the counter is zero, it asserts reset lines and
    disables clocks.

    In the same way, uniphier_aio_dai_resume() are called, it increments the
    counter after deasserting reset lines and enabling clocks.

    Fixes: 139a34200233 ("ASoC: uniphier: add support for UniPhier AIO CPU DAI driver")
    Signed-off-by: Kunihiko Hayashi <hayashi.kunihiko@socionext.com>
```

```
    Link: https://lore.kernel.org/r/1566281764-14059-1-git-send-email-
hayashi.kunihiko@socionext.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 203b82f54de11b5b015f5cf3fe0ccdacd98f5173
Author: Hans Verkuil <hverkuil-cisco@xs4all.nl>
Date:   Thu Aug 15 10:00:33 2019 -0300

    media: hdpvr: add terminating 0 at end of string

    [ Upstream commit 8b8900b729e4f31f12ac1127bde137c775c327e6 ]

    dev->usbc_buf was passed as argument for %s, but it was not safeguarded
    by a terminating 0.

    This caused this syzbot issue:

    https://syzkaller.appspot.com/bug?extid=79d18aac4bf1770dd050

    Reported-and-tested-by: syzbot+79d18aac4bf1770dd050@syzkaller.appspotmail.com

    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f4b35748511aeb94418622ecfa3f40c90ca8c789
Author: Hans Verkuil <hverkuil-cisco@xs4all.nl>
Date:   Thu Aug 15 09:40:52 2019 -0300

    media: radio/si470x: kill urb on error

    [ Upstream commit 0d616f2a3fdbf1304db44d451d9f07008556923b ]

    In the probe() function radio->int_in_urb was not killed if an
    error occurred in the probe sequence. It was also missing in
    the disconnect.

    This caused this syzbot issue:

    https://syzkaller.appspot.com/bug?extid=2d4fc2a0c45ad8da7e99

    Reported-and-tested-by: syzbot+2d4fc2a0c45ad8da7e99@syzkaller.appspotmail.com

    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 266affcbf09f12be122fa7c9486387f830608fac
Author: Hans de Goede <hdegoede@redhat.com>
Date:   Mon Aug 12 12:21:13 2019 +0200

    x86/platform/intel/iosf_mbi Rewrite locking

    [ Upstream commit 00452ba9fdb5bf6fb5fea1dae5227b4bbed44fc4 ]

    There are 2 problems with the old iosf PMIC I2C bus arbritration code which
    need to be addressed:

    1. The lockdep code complains about a possible deadlock in the
    iosf_mbi_[un]block_punit_i2c_access code:

    [    6.712662] ======================================================
    [    6.712673] WARNING: possible circular locking dependency detected
    [    6.712685] 5.3.0-rc2+ #79 Not tainted
```

```
     [    6.712692] -------------------------------------------------------
     [    6.712702] kworker/0:1/7 is trying to acquire lock:
     [    6.712712] 00000000df1c5681 (iosf_mbi_block_punit_i2c_access_count_mutex){+.+.}, at:
  iosf_mbi_unblock_punit_i2c_access+0x13/0x90
     [    6.712739]
                    but task is already holding lock:
     [    6.712749] 0000000067cb23e7 (iosf_mbi_punit_mutex){+.+.}, at:
  iosf_mbi_block_punit_i2c_access+0x97/0x186
     [    6.712768]
                    which lock already depends on the new lock.

     [    6.712780]
                    the existing dependency chain (in reverse order) is:
     [    6.712792]
                    -> #1 (iosf_mbi_punit_mutex){+.+.}:
     [    6.712808]        __mutex_lock+0xa8/0x9a0
     [    6.712818]        iosf_mbi_block_punit_i2c_access+0x97/0x186
     [    6.712831]        i2c_dw_acquire_lock+0x20/0x30
     [    6.712841]        i2c_dw_set_reg_access+0x15/0xb0
     [    6.712851]        i2c_dw_probe+0x57/0x473
     [    6.712861]        dw_i2c_plat_probe+0x33e/0x640
     [    6.712874]        platform_drv_probe+0x38/0x80
     [    6.712884]        really_probe+0xf3/0x380
     [    6.712894]        driver_probe_device+0x59/0xd0
     [    6.712905]        bus_for_each_drv+0x84/0xd0
     [    6.712915]        __device_attach+0xe4/0x170
     [    6.712925]        bus_probe_device+0x9f/0xb0
     [    6.712935]        deferred_probe_work_func+0x79/0xd0
     [    6.712946]        process_one_work+0x234/0x560
     [    6.712957]        worker_thread+0x50/0x3b0
     [    6.712967]        kthread+0x10a/0x140
     [    6.712977]        ret_from_fork+0x3a/0x50
     [    6.712986]
                    -> #0 (iosf_mbi_block_punit_i2c_access_count_mutex){+.+.}:
     [    6.713004]        __lock_acquire+0xe07/0x1930
     [    6.713015]        lock_acquire+0x9d/0x1a0
     [    6.713025]        __mutex_lock+0xa8/0x9a0
     [    6.713035]        iosf_mbi_unblock_punit_i2c_access+0x13/0x90
     [    6.713047]        i2c_dw_set_reg_access+0x4d/0xb0
     [    6.713058]        i2c_dw_probe+0x57/0x473
     [    6.713068]        dw_i2c_plat_probe+0x33e/0x640
     [    6.713079]        platform_drv_probe+0x38/0x80
     [    6.713089]        really_probe+0xf3/0x380
     [    6.713099]        driver_probe_device+0x59/0xd0
     [    6.713109]        bus_for_each_drv+0x84/0xd0
     [    6.713119]        __device_attach+0xe4/0x170
     [    6.713129]        bus_probe_device+0x9f/0xb0
     [    6.713140]        deferred_probe_work_func+0x79/0xd0
     [    6.713150]        process_one_work+0x234/0x560
     [    6.713160]        worker_thread+0x50/0x3b0
     [    6.713170]        kthread+0x10a/0x140
     [    6.713180]        ret_from_fork+0x3a/0x50
     [    6.713189]
                    other info that might help us debug this:

     [    6.713202]  Possible unsafe locking scenario:

     [    6.713212]        CPU0                    CPU1
     [    6.713221]        ----                    ----
     [    6.713229]   lock(iosf_mbi_punit_mutex);
     [    6.713239]
  lock(iosf_mbi_block_punit_i2c_access_count_mutex);
     [    6.713253]                                lock(iosf_mbi_punit_mutex);
     [    6.713265]   lock(iosf_mbi_block_punit_i2c_access_count_mutex);
     [    6.713276]
```

```
                    *** DEADLOCK ***

    In practice can never happen because only the first caller which
    increments iosf_mbi_block_punit_i2c_access_count will also take
    iosf_mbi_punit_mutex, that is the whole purpose of the counter, which
    itself is protected by iosf_mbi_block_punit_i2c_access_count_mutex.

    But there is no way to tell the lockdep code about this and we really
    want to be able to run a kernel with lockdep enabled without these
    warnings being triggered.

    2. The lockdep warning also points out another real problem, if 2 threads
    both are in a block of code protected by iosf_mbi_block_punit_i2c_access
    and the first thread to acquire the block exits before the second thread
    then the second thread will call mutex_unlock on iosf_mbi_punit_mutex,
    but it is not the thread which took the mutex and unlocking by another
    thread is not allowed.

    Fix this by getting rid of the notion of holding a mutex for the entire
    duration of the PMIC accesses, be it either from the PUnit side, or from an
    in kernel I2C driver. In general holding a mutex after exiting a function
    is a bad idea and the above problems show this case is no different.

    Instead 2 counters are now used, one for PMIC accesses from the PUnit
    and one for accesses from in kernel I2C code. When access is requested
    now the code will wait (using a waitqueue) for the counter of the other
    type of access to reach 0 and on release, if the counter reaches 0 the
    wakequeue is woken.

    Note that the counter approach is necessary to allow nested calls.
    The main reason for this is so that a series of i2c transfers can be done
    with the punit blocked from accessing the bus the whole time. This is
    necessary to be able to safely read/modify/write a PMIC register without
    racing with the PUNIT doing the same thing.

    Allowing nested iosf_mbi_block_punit_i2c_access() calls also is desirable
    from a performance pov since the whole dance necessary to block the PUnit
    from accessing the PMIC I2C bus is somewhat expensive.

    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Reviewed-by: Andy Shevchenko <andy.shevchenko@gmail.com>
    Link: https://lkml.kernel.org/r/20190812102113.95794-1-hdegoede@redhat.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e64a3456bb8b3844d7723ec3e9c865a46c70e8a9
Author: Stefan Agner <stefan.agner@toradex.com>
Date:   Mon Aug 12 14:21:17 2019 +0000

    ARM: dts: imx7-colibri: disable HS400

    [ Upstream commit a95fbda08ee20cd063ce5826e0df95a2c22ea8c5 ]

    Force HS200 by masking bit 63 of the SDHCI capability register.
    The i.MX ESDHC driver uses SDHCI_QUIRK2_CAPS_BIT63_FOR_HS400. With
    that the stack checks bit 63 to descide whether HS400 is available.
    Using sdhci-caps-mask allows to mask bit 63. The stack then selects
    HS200 as operating mode.

    This prevents rare communication errors with minimal effect on
    performance:
            sdhci-esdhc-imx 30b60000.usdhc: warning! HS400 strobe DLL
                    status REF not lock!

    Signed-off-by: Stefan Agner <stefan.agner@toradex.com>
```

```
    Signed-off-by: Philippe Schenker <philippe.schenker@toradex.com>
    Reviewed-by: Oleksandr Suvorov <oleksandr.suvorov@toradex.com>
    Signed-off-by: Shawn Guo <shawnguo@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dba52f500f6ffeb71e0b2176363965d0c6339469
Author: Bjorn Andersson <bjorn.andersson@linaro.org>
Date:   Tue Aug 13 20:09:42 2019 -0700

    arm64: dts: qcom: qcs404-evb: Mark WCSS clocks protected

    [ Upstream commit 54d895bea43c94f31304d59f82d755b7f4b59e7c ]

    '7d0c76bdf227 ("clk: qcom: Add WCSS gcc clock control for QCS404")'
    introduces two new clocks to gcc. These are not used before
    clk_disable_unused() and as such the clock framework tries to disable
    them.

    But on the EVB these registers are only accessible through TrustZone, so
    these clocks must be marked as "protected" to prevent the clock code
    from touching them.

    Numerical values are used as the constants are not yet available in a
    common tree.

    Reviewed-by: Niklas Cassel <niklas.cassel@linaro.org>
    Reported-by: Mark Brown <broonie@kernel.org>
    Reported-by: Niklas Cassel <niklas.cassel@linaro.org>
    Signed-off-by: Bjorn Andersson <bjorn.andersson@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b753aac64f9eab6a68d4797b170d7b56c4f61769
Author: André Draszik <git@andred.net>
Date:   Fri Aug 9 04:12:27 2019 +0100

    ARM: dts: imx7d: cl-som-imx7: make ethernet work again

    [ Upstream commit 9846a4524ac90b63496580b7ad50674b40d92a8f ]

    Recent changes to the atheros at803x driver caused
    ethernet to stop working on this board.
    In particular commit 6d4cd041f0af
    ("net: phy: at803x: disable delay only for RGMII mode")
    and commit cd28d1d6e52e
    ("net: phy: at803x: Disable phy delay for RGMII mode")
    fix the AR8031 driver to configure the phy's (RX/TX)
    delays as per the 'phy-mode' in the device tree.

    This now prevents ethernet from working on this board.

    It used to work before those commits, because the
    AR8031 comes out of reset with RX delay enabled, and
    the at803x driver didn't touch the delay configuration
    at all when "rgmii" mode was selected, and because
    arch/arm/mach-imx/mach-imx7d.c:ar8031_phy_fixup()
    unconditionally enables TX delay.

    Since above commits ar8031_phy_fixup() also has no
    effect anymore, and the end-result is that all delays
    are disabled in the phy, no ethernet.

    Update the device tree to restore functionality.

    Signed-off-by: André Draszik <git@andred.net>
    CC: Ilya Ledvich <ilya@compulab.co.il>
```

```
     CC: Igor Grinberg <grinberg@compulab.co.il>
     CC: Rob Herring <robh+dt@kernel.org>
     CC: Mark Rutland <mark.rutland@arm.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: NXP Linux Team <linux-imx@nxp.com>
     CC: devicetree@vger.kernel.org
     CC: linux-arm-kernel@lists.infradead.org
     Signed-off-by: Shawn Guo <shawnguo@kernel.org>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit a9f6232e13dc3809b198918b698ced3a68fa125c
 Author: Finn Thain <fthain@telegraphics.com.au>
 Date:   Fri Aug 2 10:10:25 2019 +1000

     m68k: Prevent some compiler warnings in Coldfire builds

     [ Upstream commit 94c04390225bcd8283103fd0c04be20cc30cc979 ]

     Since commit d3b41b6bb49e ("m68k: Dispatch nvram_ops calls to Atari or
     Mac functions"), Coldfire builds generate compiler warnings due to the
     unconditional inclusion of asm/atarihw.h and asm/macintosh.h.

     The inclusion of asm/atarihw.h causes warnings like this:

     In file included from ./arch/m68k/include/asm/atarihw.h:25:0,
                      from arch/m68k/kernel/setup_mm.c:41,
                      from arch/m68k/kernel/setup.c:3:
     ./arch/m68k/include/asm/raw_io.h:39:0: warning: "__raw_readb" redefined
      #define __raw_readb in_8

     In file included from ./arch/m68k/include/asm/io.h:6:0,
                      from arch/m68k/kernel/setup_mm.c:36,
                      from arch/m68k/kernel/setup.c:3:
     ./arch/m68k/include/asm/io_no.h:16:0: note: this is the location of the previous
 definition
      #define __raw_readb(addr) \
     ...

     This issue is resolved by dropping the asm/raw_io.h include. It turns out
     that asm/io_mm.h already includes that header file.

     Moving the relevant macro definitions helps to clarify this dependency
     and make it safe to include asm/atarihw.h.

     The other warnings look like this:

     In file included from arch/m68k/kernel/setup_mm.c:48:0,
                      from arch/m68k/kernel/setup.c:3:
     ./arch/m68k/include/asm/macintosh.h:19:35: warning: 'struct irq_data' declared inside
 parameter list will not be visible outside of this definition or declaration
      extern void mac_irq_enable(struct irq_data *data);
                                         ^~~~~~~~
     ...

     This issue is resolved by adding the missing linux/irq.h include.

     Signed-off-by: Finn Thain <fthain@telegraphics.com.au>
     Acked-by: Greg Ungerer <gerg@linux-m68k.org>
     Signed-off-by: Geert Uytterhoeven <geert@linux-m68k.org>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit 02e558322c45caa16f4f7fa3d8d855b3808c8709
```

```
Author: Arnd Bergmann <arnd@arndb.de>
Date:   Fri Aug 9 16:40:35 2019 +0200

    net: lpc-enet: fix printk format strings

    [ Upstream commit de6f97b2bace0e2eb6c3a86e124d1e652a587b56 ]

    compile-testing this driver on other architectures showed
    multiple warnings:

      drivers/net/ethernet/nxp/lpc_eth.c: In function 'lpc_eth_drv_probe':
      drivers/net/ethernet/nxp/lpc_eth.c:1337:19: warning: format '%d' expects argument of
type 'int', but argument 4 has type 'resource_size_t {aka long long unsigned int}' [-
Wformat=]

      drivers/net/ethernet/nxp/lpc_eth.c:1342:19: warning: format '%x' expects argument of
type 'unsigned int', but argument 4 has type 'dma_addr_t {aka long long unsigned int}' [-
Wformat=]

    Use format strings that work on all architectures.

    Link: https://lore.kernel.org/r/20190809144043.476786-10-arnd@arndb.de
    Reported-by: kbuild test robot <lkp@intel.com>
    Signed-off-by: Arnd Bergmann <arnd@arndb.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c44f658d1731f0ebb48397cda27a36291959833c
Author: Mark Rutland <mark.rutland@arm.com>
Date:   Wed Aug 14 15:31:57 2019 +0100

    kasan/arm64: fix CONFIG_KASAN_SW_TAGS && KASAN_INLINE

    [ Upstream commit 34b5560db40d2941cfbe82eca1641353d5aed1a9 ]

    The generic Makefile.kasan propagates CONFIG_KASAN_SHADOW_OFFSET into
    KASAN_SHADOW_OFFSET, but only does so for CONFIG_KASAN_GENERIC.

    Since commit:

      6bd1d0be0e97936d ("arm64: kasan: Switch to using KASAN_SHADOW_OFFSET")

    ... arm64 defines CONFIG_KASAN_SHADOW_OFFSET in Kconfig rather than
    defining KASAN_SHADOW_OFFSET in a Makefile. Thus, if
    CONFIG_KASAN_SW_TAGS && KASAN_INLINE are selected, we get build time
    splats due to KASAN_SHADOW_OFFSET not being set:

    | [mark@lakrids:~/src/linux]% usellvm 8.0.1 usekorg 8.1.0  make ARCH=arm64
CROSS_COMPILE=aarch64-linux- CC=clang
    | scripts/kconfig/conf  --syncconfig Kconfig
    |   CC      scripts/mod/empty.o
    | clang (LLVM option parsing): for the -hwasan-mapping-offset option: '' value invalid
for uint argument!
    | scripts/Makefile.build:273: recipe for target 'scripts/mod/empty.o' failed
    | make[1]: *** [scripts/mod/empty.o] Error 1
    | Makefile:1123: recipe for target 'prepare0' failed
    | make: *** [prepare0] Error 2

    Let's fix this by always propagating CONFIG_KASAN_SHADOW_OFFSET into
    KASAN_SHADOW_OFFSET if CONFIG_KASAN is selected, moving the existing
    common definition of +CFLAGS_KASAN_NOSANITIZE to the top of
    Makefile.kasan.

    Cc: Catalin Marinas <catalin.marinas@arm.com>
    Signed-off-by: Mark Rutland <mark.rutland@arm.com>
    Acked-by: Andrey Ryabinin <aryabinin@virtuozzo.com>
```

```
    Tested-by Steve Capper <steve.capper@arm.com>
    Signed-off-by: Will Deacon <will@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e67ec23a77b45e51c6f7b27f259233e24ccd893d
Author: Ezequiel Garcia <ezequiel@collabora.com>
Date:   Thu Jun 27 19:29:12 2019 -0300

    media: imx: mipi csi-2: Don't fail if initial state times-out

    [ Upstream commit 0d5078c7172c46db6c58718d817b9fcf769554b4 ]

    Not all sensors will be able to guarantee a proper initial state.
    This may be either because the driver is not properly written,
    or (probably unlikely) because the hardware won't support it.

    While the right solution in the former case is to fix the sensor
    driver, the real world not always allows right solutions, due to lack
    of available documentation and support on these sensors.

    Let's relax this requirement, and allow the driver to support stream start,
    even if the sensor initial sequence wasn't the expected.

    Also improve the warning message to better explain the problem and provide
    a hint that the sensor driver needs to be fixed.

    Signed-off-by: Ezequiel Garcia <ezequiel@collabora.com>
    Signed-off-by: Fabio Estevam <festevam@gmail.com>
    Reviewed-by: Steve Longerbeam <slongerbeam@gmail.com>
    Reviewed-by: Philipp Zabel <p.zabel@pengutronix.de>
    Signed-off-by: Sakari Ailus <sakari.ailus@linux.intel.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 820e85a3821d1ec48d9a732e72d462e38b5488af
Author: Sakari Ailus <sakari.ailus@linux.intel.com>
Date:   Wed Aug 7 11:21:27 2019 -0300

    media: omap3isp: Don't set streaming state on random subdevs

    [ Upstream commit 7ef57be07ac146e70535747797ef4aee0f06e9f9 ]

    The streaming state should be set to the first upstream sub-device only,
    not everywhere, for a sub-device driver itself knows how to best control
    the streaming state of its own upstream sub-devices.

    Signed-off-by: Sakari Ailus <sakari.ailus@linux.intel.com>
    Reviewed-by: Laurent Pinchart <laurent.pinchart@ideasonboard.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2418d4b41db19ed6d888338100297344e00727e3
Author: Ezequiel Garcia <ezequiel@collabora.com>
Date:   Thu Aug 8 08:05:40 2019 -0300

    media: i2c: ov5645: Fix power sequence

    [ Upstream commit 092e8eb90a7dc7dd210cd4e2ea36075d0a7f96af ]

    This is mostly a port of Jacopo's fix:

      commit aa4bb8b8838ffcc776a79f49a4d7476b82405349
      Author: Jacopo Mondi <jacopo@jmondi.org>
      Date:   Fri Jul 6 05:51:52 2018 -0400
```

```
        media: ov5640: Re-work MIPI startup sequence

    In the OV5645 case, the changes are:

    - At set_power(1) time power up MIPI Tx/Rx and set data and clock lanes in
      LP11 during 'sleep' and 'idle' with MIPI clock in non-continuous mode.
    - At set_power(0) time power down MIPI Tx/Rx (in addition to the current
      power down of regulators and clock gating).
    - At s_stream time enable/disable the MIPI interface output.

    With this commit the sensor is able to enter LP-11 mode during power up,
    as expected by some CSI-2 controllers.

    Many thanks to Fabio Estevam for his help debugging this issue.

    Tested-by: Fabio Estevam <festevam@gmail.com>
    Signed-off-by: Ezequiel Garcia <ezequiel@collabora.com>
    Reviewed-by: Philipp Zabel <p.zabel@pengutronix.de>
    Reviewed-by: Jacopo Mondi <jacopo@jmondi.org>
    Signed-off-by: Sakari Ailus <sakari.ailus@linux.intel.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5a132f34ba5a4b593338c0928535d56945ffb0fc
Author: Colin Ian King <colin.king@canonical.com>
Date:   Sun Jul 28 14:11:24 2019 -0300

    media: vsp1: fix memory leak of dl on error return path

    [ Upstream commit 70c55c1ad1a76e804ee5330e134674f5d2741cb7 ]

    Currently when the call vsp1_dl_body_get fails and returns null the
    error return path leaks the allocation of dl. Fix this by kfree'ing
    dl before returning.

    Addresses-Coverity: ("Resource leak")

    Fixes: 5d7936b8e27d ("media: vsp1: Convert display lists to use new body pool")
    Signed-off-by: Colin Ian King <colin.king@canonical.com>
    Reviewed-by: Kieran Bingham <kieran.bingham+renesas@ideasonboard.com>
    Signed-off-by: Laurent Pinchart <laurent.pinchart+renesas@ideasonboard.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 329649129872495ad1488a79e5e74bd34b03a1d5
Author: Tan Xiaojun <tanxiaojun@huawei.com>
Date:   Fri Aug 2 11:48:57 2019 +0800

    perf record: Support aarch64 random socket_id assignment

    [ Upstream commit 0a4d8fb229dd78f9e0752817339e19e903b37a60 ]

    Same as in the commit 01766229533f ("perf record: Support s390 random
    socket_id assignment"), aarch64 also have this problem.

    Without this fix:

      [root@localhost perf]# ./perf report --header -I -v
      ...
      socket_id number is too big.You may need to upgrade the perf tool.

      # ========
      # captured on    : Thu Aug  1 22:58:38 2019
      # header version : 1
      ...
```

```
       # Core ID and Socket ID information is not available
       ...

     With this fix:
       [root@localhost perf]# ./perf report --header -I -v
       ...
       cpumask list: 0-31
       cpumask list: 32-63
       cpumask list: 64-95
       cpumask list: 96-127

       # ========
       # captured on    : Thu Aug  1 22:58:38 2019
       # header version : 1
       ...
       # CPU 0: Core ID 0, Socket ID 36
       # CPU 1: Core ID 1, Socket ID 36
       ...
       # CPU 126: Core ID 126, Socket ID 8442
       # CPU 127: Core ID 127, Socket ID 8442
       ...

     Signed-off-by: Tan Xiaojun <tanxiaojun@huawei.com>
     Acked-by: Jiri Olsa <jolsa@kernel.org>
     Cc: Alexander Shishkin <alexander.shishkin@linux.intel.com>
     Cc: Alexey Budankov <alexey.budankov@linux.intel.com>
     Cc: Kan Liang <kan.liang@linux.intel.com>
     Cc: Namhyung Kim <namhyung@kernel.org>
     Cc: Peter Zijlstra <peterz@infradead.org>
     Cc: Song Liu <songliubraving@fb.com>
     Cc: Steven Rostedt (VMware) <rostedt@goodmis.org>
     Cc: Tzvetomir Stoyanov (VMware) <tz.stoyanov@gmail.com>
     Link: http://lkml.kernel.org/r/1564717737-21602-1-git-send-email-tanxiaojun@huawei.com
     Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 1faf44895a71350aa9ce8e2d76fa67da429f22c2
  Author: Arnd Bergmann <arnd@arndb.de>
  Date:   Fri Aug 9 18:33:19 2019 +0200

     ARM: xscale: fix multi-cpu compilation

     [ Upstream commit c7b68049943079550d4e6af0f10aa3aabd64131a ]

     Building a combined ARMv4+XScale kernel produces these
     and other build failures:

     /tmp/copypage-xscale-3aa821.s: Assembler messages:
     /tmp/copypage-xscale-3aa821.s:167: Error: selected processor does not support `pld
  [r7,#0]' in ARM mode
     /tmp/copypage-xscale-3aa821.s:168: Error: selected processor does not support `pld
  [r7,#32]' in ARM mode
     /tmp/copypage-xscale-3aa821.s:169: Error: selected processor does not support `pld
  [r1,#0]' in ARM mode
     /tmp/copypage-xscale-3aa821.s:170: Error: selected processor does not support `pld
  [r1,#32]' in ARM mode
     /tmp/copypage-xscale-3aa821.s:171: Error: selected processor does not support `pld
  [r7,#64]' in ARM mode
     /tmp/copypage-xscale-3aa821.s:176: Error: selected processor does not support `ldrd
  r4,r5,[r7],#8' in ARM mode
     /tmp/copypage-xscale-3aa821.s:180: Error: selected processor does not support `strd
  r4,r5,[r1],#8' in ARM mode

     Add an explict .arch armv5 in the inline assembly to allow the ARMv5
     specific instructions regardless of the compiler -march= target.
```

```
    Link: https://lore.kernel.org/r/20190809163334.489360-5-arnd@arndb.de
    Signed-off-by: Arnd Bergmann <arnd@arndb.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7b5c2568470fb7117994193ac301213d1e16da81
Author: Arnd Bergmann <arnd@arndb.de>
Date:   Fri Aug 9 18:33:17 2019 +0200

    dmaengine: iop-adma: use correct printk format strings

    [ Upstream commit 00c9755524fbaa28117be774d7c92fddb5ca02f3 ]

    When compile-testing on other architectures, we get lots of warnings
    about incorrect format strings, like:

       drivers/dma/iop-adma.c: In function 'iop_adma_alloc_slots':
       drivers/dma/iop-adma.c:307:6: warning: format '%x' expects argument of type 'unsigned
int', but argument 6 has type 'dma_addr_t {aka long long unsigned int}' [-Wformat=]

       drivers/dma/iop-adma.c: In function 'iop_adma_prep_dma_memcpy':
    >> drivers/dma/iop-adma.c:518:40: warning: format '%u' expects argument of type
'unsigned int', but argument 5 has type 'size_t {aka long unsigned int}' [-Wformat=]

    Use %zu for printing size_t as required, and cast the dma_addr_t
    arguments to 'u64' for printing with %llx. Ideally this should use
    the %pad format string, but that requires an lvalue argument that
    doesn't work here.

    Link: https://lore.kernel.org/r/20190809163334.489360-3-arnd@arndb.de
    Signed-off-by: Arnd Bergmann <arnd@arndb.de>
    Acked-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Arnd Bergmann <arnd@arndb.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 151e5ef048db1723fdd7164a52d160735c96f8af
Author: Darius Rad <alpha@area49.net>
Date:   Tue Jul 23 13:37:46 2019 -0300

    media: rc: imon: Allow iMON RC protocol for ffdc 7e device

    [ Upstream commit b20a6e298bcb8cb8ae18de26baaf462a6418515b ]

    Allow selecting the IR protocol, MCE or iMON, for a device that
    identifies as follows (with config id 0x7e):

    15c2:ffdc SoundGraph Inc. iMON PAD Remote Controller

    As the driver is structured to default to iMON when both RC
    protocols are supported, existing users of this device (using MCE
    protocol) will need to manually switch to MCE (RC-6) protocol from
    userspace (with ir-keytable, sysfs).

    Signed-off-by: Darius Rad <alpha@area49.net>
    Signed-off-by: Sean Young <sean@mess.org>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5f318db1a360524fd2500a995c09322758a8e28b
Author: John Keeping <john@metanate.com>
Date:   Thu Aug 15 11:01:45 2019 +0100

    perf unwind: Fix libunwind when tid != pid

    [ Upstream commit e8ba2906f6b9054102ad035ac9cafad9d4168589 ]
```

```
    Commit e5adfc3e7e77 ("perf map: Synthesize maps only for thread group
    leader") changed the recording side so that we no longer get mmap events
    for threads other than the thread group leader (when synthesising these
    events for threads which exist before perf is started).

    When a file recorded after this change is loaded, the lack of mmap
    records mean that unwinding is not set up for any other threads.

    This can be seen in a simple record/report scenario:

            perf record --call-graph=dwarf -t $TID
            perf report

    If $TID is a process ID then the report will show call graphs, but if
    $TID is a secondary thread the output is as if --call-graph=none was
    specified.

    Following the rationale in that commit, move the libunwind fields into
    struct map_groups and update the libunwind functions to take this
    instead of the struct thread.  This is only required for
    unwind__finish_access which must now be called from map_groups__delete
    and the others are changed for symmetry.

    Note that unwind__get_entries keeps the thread argument since it is
    required for symbol lookup and the libdw unwind provider uses the thread
    ID.

    Signed-off-by: John Keeping <john@metanate.com>
    Reviewed-by: Jiri Olsa <jolsa@kernel.org>
    Cc: Alexander Shishkin <alexander.shishkin@linux.intel.com>
    Cc: Konstantin Khlebnikov <khlebnikov@yandex-team.ru>
    Cc: Namhyung Kim <namhyung@kernel.org>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Fixes: e5adfc3e7e77 ("perf map: Synthesize maps only for thread group leader")
    Link: http://lkml.kernel.org/r/20190815100146.28842-2-john@metanate.com
    Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit eed5e182543c7e34121049a0c29038373f0b0e95
Author: Kees Cook <keescook@chromium.org>
Date:   Tue Aug 13 16:04:50 2019 -0700

    arm64/efi: Move variable assignments after SECTIONS

    [ Upstream commit 90776dd1c427cbb4d381aa4b13338f1fb1d20f5e ]

    It seems that LLVM's linker does not correctly handle variable assignments
    involving section positions that are updated during the SECTIONS
    parsing. Commit aa69fb62bea1 ("arm64/efi: Mark __efistub_stext_offset as
    an absolute symbol explicitly") ran into this too, but found a different
    workaround.

    However, this was not enough, as other variables were also miscalculated
    which manifested as boot failures under UEFI where __efistub__end was
    not taking the correct _end value (they should be the same):

    $ ld.lld -EL -maarch64elf --no-undefined -X -shared \
            -Bsymbolic -z notext -z norelro --no-apply-dynamic-relocs \
            -o vmlinux.lld -T poc.lds --whole-archive vmlinux.o && \
      readelf -Ws vmlinux.lld | egrep '\b(__efistub_|)_end\b'
    368272: ffff000002218000     0 NOTYPE  LOCAL  HIDDEN    38 __efistub__end
    368322: ffff000012318000     0 NOTYPE  GLOBAL DEFAULT   38 _end

    $ aarch64-linux-gnu-ld.bfd -EL -maarch64elf --no-undefined -X -shared \
```

```
            -Bsymbolic -z notext -z norelro --no-apply-dynamic-relocs \
            -o vmlinux.bfd -T poc.lds --whole-archive vmlinux.o && \
      readelf -Ws vmlinux.bfd | egrep '\b(__efistub_|)_end\b'
     338124: ffff000012318000     0 NOTYPE  LOCAL  DEFAULT  ABS __efistub__end
     383812: ffff000012318000     0 NOTYPE  GLOBAL DEFAULT 15325 _end

     To work around this, all of the __efistub_-prefixed variable assignments
     need to be moved after the linker script's SECTIONS entry. As it turns
     out, this also solves the problem fixed in commit aa69fb62bea1, so those
     changes are reverted here.

     Link: https://github.com/ClangBuiltLinux/linux/issues/634
     Link: https://bugs.llvm.org/show_bug.cgi?id=42990
     Acked-by: Ard Biesheuvel <ard.biesheuvel@linaro.org>
     Signed-off-by: Kees Cook <keescook@chromium.org>
     Signed-off-by: Will Deacon <will@kernel.org>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 89fe850d63482449c8612175b1c95adca7c6b7b5
Author: Sean Young <sean@mess.org>
Date:   Sun Aug 11 02:05:51 2019 -0300

     media: em28xx: modules workqueue not inited for 2nd device

     [ Upstream commit 46e4a26615cc7854340e4b69ca59ee78d6f20c8b ]

     syzbot reports an error on flush_request_modules() for the second device.
     This workqueue was never initialised so simply remove the offending line.

     usb 1-1: USB disconnect, device number 2
     em28xx 1-1:1.153: Disconnecting em28xx #1
     ------------[ cut here ]------------
     WARNING: CPU: 0 PID: 12 at kernel/workqueue.c:3031
     __flush_work.cold+0x2c/0x36 kernel/workqueue.c:3031
     Kernel panic - not syncing: panic_on_warn set ...
     CPU: 0 PID: 12 Comm: kworker/0:1 Not tainted 5.3.0-rc2+ #25
     Hardware name: Google Google Compute Engine/Google Compute Engine, BIOS
     Google 01/01/2011
     Workqueue: usb_hub_wq hub_event
     Call Trace:
       __dump_stack lib/dump_stack.c:77 [inline]
       dump_stack+0xca/0x13e lib/dump_stack.c:113
       panic+0x2a3/0x6da kernel/panic.c:219
       __warn.cold+0x20/0x4a kernel/panic.c:576
       report_bug+0x262/0x2a0 lib/bug.c:186
       fixup_bug arch/x86/kernel/traps.c:179 [inline]
       fixup_bug arch/x86/kernel/traps.c:174 [inline]
       do_error_trap+0x12b/0x1e0 arch/x86/kernel/traps.c:272
       do_invalid_op+0x32/0x40 arch/x86/kernel/traps.c:291
       invalid_op+0x23/0x30 arch/x86/entry/entry_64.S:1026
     RIP: 0010:__flush_work.cold+0x2c/0x36 kernel/workqueue.c:3031
     Code: 9a 22 00 48 c7 c7 20 e4 c5 85 e8 d9 3a 0d 00 0f 0b 45 31 e4 e9 98 86
     ff ff e8 51 9a 22 00 48 c7 c7 20 e4 c5 85 e8 be 3a 0d 00 <0f> 0b 45 31 e4
     e9 7d 86 ff ff e8 36 9a 22 00 48 c7 c7 20 e4 c5 85
     RSP: 0018:ffff8881da20f720 EFLAGS: 00010286
     RAX: 0000000000000024 RBX: dfffc0000000000 RCX: 0000000000000000
     RDX: 0000000000000000 RSI: ffffffff8128a0fd RDI: ffffed103b441ed6
     RBP: ffff8881da20f888 R08: 0000000000000024 R09: fffffbfff11acd9a
     R10: fffffbfff11acd99 R11: ffffffff88d66ccf R12: 0000000000000000
     R13: 0000000000000001 R14: ffff8881c6685df8 R15: ffff8881d2a85b78
       flush_request_modules drivers/media/usb/em28xx/em28xx-cards.c:3325 [inline]
       em28xx_usb_disconnect.cold+0x280/0x2a6
     drivers/media/usb/em28xx/em28xx-cards.c:4023
       usb_unbind_interface+0x1bd/0x8a0 drivers/usb/core/driver.c:423
       __device_release_driver drivers/base/dd.c:1120 [inline]
```

```
       device_release_driver_internal+0x404/0x4c0 drivers/base/dd.c:1151
       bus_remove_device+0x2dc/0x4a0 drivers/base/bus.c:556
       device_del+0x420/0xb10 drivers/base/core.c:2288
       usb_disable_device+0x211/0x690 drivers/usb/core/message.c:1237
       usb_disconnect+0x284/0x8d0 drivers/usb/core/hub.c:2199
       hub_port_connect drivers/usb/core/hub.c:4949 [inline]
       hub_port_connect_change drivers/usb/core/hub.c:5213 [inline]
       port_event drivers/usb/core/hub.c:5359 [inline]
       hub_event+0x1454/0x3640 drivers/usb/core/hub.c:5441
       process_one_work+0x92b/0x1530 kernel/workqueue.c:2269
       process_scheduled_works kernel/workqueue.c:2331 [inline]
       worker_thread+0x7ab/0xe20 kernel/workqueue.c:2417
       kthread+0x318/0x420 kernel/kthread.c:255
       ret_from_fork+0x24/0x30 arch/x86/entry/entry_64.S:352
     Kernel Offset: disabled
     Rebooting in 86400 seconds..

     Fixes: be7fd3c3a8c5e ("media: em28xx: Hauppauge DualHD second tuner functionality)
     Reviewed-by: Ezequiel Garcia <ezequiel@collabora.com>
     Reviewed-by: Brad Love <brad@nextdimension.cc>
     Reported-by: syzbot+b7f57261c521087d89bb@syzkaller.appspotmail.com
     Signed-off-by: Sean Young <sean@mess.org>
     Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit db0a74c7e5acc373a0a1c966c2f4813ca338842c
  Author: Geert Uytterhoeven <geert+renesas@glider.be>
  Date:   Tue Jul 9 06:59:52 2019 -0300

     media: fdp1: Reduce FCP not found message level to debug

     [ Upstream commit 4fd22938569c14f6092c05880ca387409d78355f ]

     When support for the IPMMU is not enabled, the FDP driver may be
     probe-deferred multiple times, causing several messages to be printed
     like:

         rcar_fdp1 fe940000.fdp1: FCP not found (-517)
         rcar_fdp1 fe944000.fdp1: FCP not found (-517)

     Fix this by reducing the message level to debug level, as is done in the
     VSP1 driver.

     Fixes: 4710b752e029f3f8 ("[media] v4l: Add Renesas R-Car FDP1 Driver")
     Signed-off-by: Geert Uytterhoeven <geert+renesas@glider.be>
     Reviewed-by: Kieran Bingham <kieran.bingham+renesas@ideasonboard.com>
     Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
     Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 42f1a25f89f7824e9f4df67960be63642a9d6b2f
  Author: Wolfram Sang <wsa+renesas@sang-engineering.com>
  Date:   Fri Aug 9 13:52:15 2019 -0300

     media: i2c: tda1997x: prevent potential NULL pointer access

     [ Upstream commit 2f822f1da08ac5c93e351e79d22920f08fa51baf ]

     i2c_new_dummy() can fail returning a NULL pointer. This is not checked
     and the returned pointer is blindly used. Convert to
     devm_i2c_new_dummy_client() which returns an ERR_PTR and also add a
     validity check. Using devm_* here also fixes a leak because the dummy
     client was not released in the probe error path.

     Signed-off-by: Wolfram Sang <wsa+renesas@sang-engineering.com>
```

```
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3c9b7867785670ebcd28f01d3028339063582559
Author: Matthias Brugger <matthias.bgg@gmail.com>
Date:   Fri Jun 21 08:32:50 2019 -0300

    media: mtk-mdp: fix reference count on old device tree

    [ Upstream commit 864919ea0380e62adb2503b89825fe358acb8216 ]

    of_get_next_child() increments the reference count of the returning
    device_node. Decrement it in the check if we are using the old or the
    new DTB.

    Fixes: ba1f1f70c2c0 ("[media] media: mtk-mdp: Fix mdp device tree")
    Signed-off-by: Matthias Brugger <matthias.bgg@gmail.com>
    Acked-by: Houlong Wei <houlong.wei@mediatek.com>
    [hverkuil-cisco@xs4all.nl: use node instead of parent as temp variable]
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 392574fcd9315477d3c455e35c3deba6d3ce7b4a
Author: Arnaldo Carvalho de Melo <acme@redhat.com>
Date:   Tue Jul 30 11:37:44 2019 -0300

    perf test vfs_getname: Disable ~/.perfconfig to get default output

    [ Upstream commit 4fe94ce1c6ba678b5f12b94bb9996eea4fc99e85 ]

    To get the expected output we have to ignore whatever changes the user
    has in its ~/.perfconfig file, so set PERF_CONFIG to /dev/null to
    achieve that.

    Before:

      # egrep 'trace|show_' ~/.perfconfig
      [trace]
            show_zeros = yes
            show_duration = no
            show_timestamp = no
            show_arg_names = no
            show_prefix = yes
      # echo $PERF_CONFIG

      # perf test "trace + vfs_getname"
      70: Check open filename arg using perf trace + vfs_getname: FAILED!
      # export PERF_CONFIG=/dev/null
      # perf test "trace + vfs_getname"
      70: Check open filename arg using perf trace + vfs_getname: Ok
      #

    After:

      # egrep 'trace|show_' ~/.perfconfig
      [trace]
            show_zeros = yes
            show_duration = no
            show_timestamp = no
            show_arg_names = no
            show_prefix = yes
      # echo $PERF_CONFIG
```

```
       # perf test "trace + vfs_getname"
       70: Check open filename arg using perf trace + vfs_getname: Ok
       #

    Cc: Adrian Hunter <adrian.hunter@intel.com>
    Cc: Jiri Olsa <jolsa@kernel.org>
    Cc: Luis Cláudio Gonçalves <lclaudio@redhat.com>
    Cc: Namhyung Kim <namhyung@kernel.org>
    Cc: Taeung Song <treeze.taeung@gmail.com>
    Link: https://lkml.kernel.org/n/tip-3up27pexg5i3exuzqrvt4m8u@git.kernel.org
    Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b9ef8043c9b97c83422ca12fc949a6b54944084c
Author: Arnaldo Carvalho de Melo <acme@redhat.com>
Date:   Tue Jul 30 11:20:55 2019 -0300

    perf config: Honour $PERF_CONFIG env var to specify alternate .perfconfig

    [ Upstream commit 61a461fcbd62d42c29a1ea6a9cc3838ad9f49401 ]

    We had this comment in Documentation/perf_counter/config.c, i.e. since
    when we got this from the git sources, but never really did that
    getenv("PERF_CONFIG"), do it now as I need to disable whatever
    ~/.perfconfig root has so that tests parsing tool output are done for
    the expected default output or that we specify an alternate config file
    that when read will make the tools produce expected output.

    Cc: Adrian Hunter <adrian.hunter@intel.com>
    Cc: Jiri Olsa <jolsa@kernel.org>
    Cc: Luis Cláudio Gonçalves <lclaudio@redhat.com>
    Cc: Namhyung Kim <namhyung@kernel.org>
    Cc: Taeung Song <treeze.taeung@gmail.com>
    Fixes: 078006012401 ("perf_counter tools: add in basic glue from Git")
    Link: https://lkml.kernel.org/n/tip-jo209zac9rut0dz1rqvbdlgm@git.kernel.org
    Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a6f247eda9ed95857499796ea5d7d9ff55eda324
Author: Hans Verkuil <hverkuil-cisco@xs4all.nl>
Date:   Fri Aug 16 03:38:13 2019 -0300

    media: gspca: zero usb_buf on error

    [ Upstream commit 4843a543fad3bf8221cf14e5d5f32d15cee89e84 ]

    If reg_r() fails, then gspca_dev->usb_buf was left uninitialized,
    and some drivers used the contents of that buffer in logic.

    This caused several syzbot errors:

    https://syzkaller.appspot.com/bug?extid=397fd082ce5143e2f67d
    https://syzkaller.appspot.com/bug?extid=1a35278dd0ebfb3a038a
    https://syzkaller.appspot.com/bug?extid=06ddf1788cfd048c5e82

    I analyzed the gspca drivers and zeroed the buffer where needed.

    Reported-and-tested-by: syzbot+1a35278dd0ebfb3a038a@syzkaller.appspotmail.com
    Reported-and-tested-by: syzbot+397fd082ce5143e2f67d@syzkaller.appspotmail.com
    Reported-and-tested-by: syzbot+06ddf1788cfd048c5e82@syzkaller.appspotmail.com

    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 60b31d55b0e7cc3dd177380efd4a5250e23e7a70
Author: zhengbin <zhengbin13@huawei.com>
Date:   Tue Jul 23 22:10:42 2019 +0800

    blk-mq: Fix memory leak in blk_mq_init_allocated_queue error handling

    [ Upstream commit 73d9c8d4c0017e21e1ff519474ceb1450484dc9a ]

    If blk_mq_init_allocated_queue->elevator_init_mq fails, need to release
    the previously requested resources.

    Fixes: d34849913819 ("blk-mq-sched: allow setting of default IO scheduler")
    Signed-off-by: zhengbin <zhengbin13@huawei.com>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ca9d5758f075a7156a1da7723fd99f47c0ebf954
Author: Peter Zijlstra <peterz@infradead.org>
Date:   Wed Jun 5 07:46:43 2019 -0700

    idle: Prevent late-arriving interrupts from disrupting offline

    [ Upstream commit e78a7614f3876ac649b3df608789cb6ef74d0480 ]

    Scheduling-clock interrupts can arrive late in the CPU-offline process,
    after idle entry and the subsequent call to cpuhp_report_idle_dead().
    Once execution passes the call to rcu_report_dead(), RCU is ignoring
    the CPU, which results in lockdep complaints when the interrupt handler
    uses RCU:

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

    =============================
    WARNING: suspicious RCU usage
    5.2.0-rc1+ #681 Not tainted
    -----------------------------
    kernel/sched/fair.c:9542 suspicious rcu_dereference_check() usage!

    other info that might help us debug this:

    RCU used illegally from offline CPU!
    rcu_scheduler_active = 2, debug_locks = 1
    no locks held by swapper/5/0.

    stack backtrace:
    CPU: 5 PID: 0 Comm: swapper/5 Not tainted 5.2.0-rc1+ #681
    Hardware name: QEMU Standard PC (Q35 + ICH9, 2009), BIOS Bochs 01/01/2011
    Call Trace:
     <IRQ>
     dump_stack+0x5e/0x8b
     trigger_load_balance+0xa8/0x390
     ? tick_sched_do_timer+0x60/0x60
     update_process_times+0x3b/0x50
     tick_sched_handle+0x2f/0x40
     tick_sched_timer+0x32/0x70
     __hrtimer_run_queues+0xd3/0x3b0
     hrtimer_interrupt+0x11d/0x270
     ? sched_clock_local+0xc/0x74
     smp_apic_timer_interrupt+0x79/0x200
     apic_timer_interrupt+0xf/0x20
     </IRQ>
    RIP: 0010:delay_tsc+0x22/0x50
    Code: ff 0f 1f 80 00 00 00 00 65 44 8b 05 18 a7 11 48 0f ae e8 0f 31 48 89 d6 48 c1 e6
20 48 09 c6 eb 0e f3 90 65 8b 05 fe a6 11 48 <41> 39 c0 75 18 0f ae e8 0f 31 48 c1 e2 20 48
09 c2 48 89 d0 48 29
```

```
     RSP: 0000:ffff8f92c0157ed0 EFLAGS: 00000212 ORIG_RAX: ffffffffffffff13
     RAX: 0000000000000005 RBX: ffff8c861f356400 RCX: ffff8f92c0157e64
     RDX: 000000321214c8cc RSI: 00000032120daa7f RDI: 0000000000260f15
     RBP: 0000000000000005 R08: 0000000000000005 R09: 0000000000000000
     R10: 0000000000000001 R11: 0000000000000001 R12: 0000000000000000
     R13: 0000000000000000 R14: ffff8c861ee18000 R15: ffff8c861ee18000
      cpuhp_report_idle_dead+0x31/0x60
      do_idle+0x1d5/0x200
      ? _raw_spin_unlock_irqrestore+0x2d/0x40
      cpu_startup_entry+0x14/0x20
      start_secondary+0x151/0x170
      secondary_startup_64+0xa4/0xb0

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

     This happens rarely, but can be forced by happen more often by
     placing delays in cpuhp_report_idle_dead() following the call to
     rcu_report_dead().  With this in place, the following rcutorture
     scenario reproduces the problem within a few minutes:

     tools/testing/selftests/rcutorture/bin/kvm.sh --cpus 8 --duration 5 --kconfig
  "CONFIG_DEBUG_LOCK_ALLOC=y CONFIG_PROVE_LOCKING=y" --configs "TREE04"

     This commit uses the crude but effective expedient of moving the disabling
     of interrupts within the idle loop to precede the cpu_is_offline()
     check.  It also invokes tick_nohz_idle_stop_tick() instead of
     tick_nohz_idle_stop_tick_protected() to shut off the scheduling-clock
     interrupt.

     Signed-off-by: Peter Zijlstra <peterz@infradead.org>
     Cc: Frederic Weisbecker <fweisbec@gmail.com>
     Cc: Thomas Gleixner <tglx@linutronix.de>
     Cc: Ingo Molnar <mingo@kernel.org>
     [ paulmck: Revert tick_nohz_idle_stop_tick_protected() removal, new callers. ]
     Signed-off-by: Paul E. McKenney <paulmck@linux.ibm.com>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 7bdf8c0ac8b3109e54ecd4791eddfa430b6f43f9
  Author: Phil Auld <pauld@redhat.com>
  Date:   Thu Aug 1 09:37:49 2019 -0400

     sched/fair: Use rq_lock/unlock in online_fair_sched_group

     [ Upstream commit a46d14eca7b75fffe35603aa8b81df654353d80f ]

     Enabling WARN_DOUBLE_CLOCK in /sys/kernel/debug/sched_features causes
     warning to fire in update_rq_clock. This seems to be caused by onlining
     a new fair sched group not using the rq lock wrappers.

       [] rq->clock_update_flags & RQCF_UPDATED
       [] WARNING: CPU: 5 PID: 54385 at kernel/sched/core.c:210 update_rq_clock+0xec/0x150

       [] Call Trace:
       []  online_fair_sched_group+0x53/0x100
       []  cpu_cgroup_css_online+0x16/0x20
       []  online_css+0x1c/0x60
       []  cgroup_apply_control_enable+0x231/0x3b0
       []  cgroup_mkdir+0x41b/0x530
       []  kernfs_iop_mkdir+0x61/0xa0
       []  vfs_mkdir+0x108/0x1a0
       []  do_mkdirat+0x77/0xe0
       []  do_syscall_64+0x55/0x1d0
       []  entry_SYSCALL_64_after_hwframe+0x44/0xa9

     Using the wrappers in online_fair_sched_group instead of the raw locking
```

```
    removes this warning.

    [ tglx: Use rq_*lock_irq() ]

    Signed-off-by: Phil Auld <pauld@redhat.com>
    Signed-off-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Cc: Ingo Molnar <mingo@redhat.com>
    Cc: Vincent Guittot <vincent.guittot@linaro.org>
    Cc: Ingo Molnar <mingo@kernel.org>
    Link: https://lkml.kernel.org/r/20190801133749.11033-1-pauld@redhat.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5de4deebd5490b25a1f0ddfbf43f144204264233
Author: Sudeep Holla <sudeep.holla@arm.com>
Date:   Mon Jul 8 15:48:36 2019 +0100

    firmware: arm_scmi: Check if platform has released shmem before using

    [ Upstream commit 9dc34d635c67e57051853855c43249408641a5ab ]

    Sometimes platfom may take too long to respond to the command and OS
    might timeout before platform transfer the ownership of the shared
    memory region to the OS with the response.

    Since the mailbox channel associated with the channel is freed and new
    commands are dispatch on the same channel, OS needs to wait until it
    gets back the ownership. If not, either OS may end up overwriting the
    platform response for the last command(which is fine as OS timed out
    that command) or platform might overwrite the payload for the next
    command with the response for the old.

    The latter is problematic as platform may end up interpretting the
    response as the payload. In order to avoid such race, let's wait until
    the OS gets back the ownership before we prepare the shared memory with
    the payload for the next command.

    Reported-by: Jim Quinlan <james.quinlan@broadcom.com>
    Signed-off-by: Sudeep Holla <sudeep.holla@arm.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 96857ff7e953dafd0b2f3b89fff055cf57dd9f57
Author: Xiaofei Tan <tanxiaofei@huawei.com>
Date:   Fri Jul 26 09:43:37 2019 +0800

    efi: cper: print AER info of PCIe fatal error

    [ Upstream commit b194a77fcc4001dc40aecdd15d249648e8a436d1 ]

    AER info of PCIe fatal error is not printed in the current driver.
    Because APEI driver will panic directly for fatal error, and can't
    run to the place of printing AER info.

    An example log is as following:
    {763}[Hardware Error]: Hardware error from APEI Generic Hardware Error Source: 11
    {763}[Hardware Error]: event severity: fatal
    {763}[Hardware Error]:  Error 0, type: fatal
    {763}[Hardware Error]:   section_type: PCIe error
    {763}[Hardware Error]:   port_type: 0, PCIe end point
    {763}[Hardware Error]:   version: 4.0
    {763}[Hardware Error]:   command: 0x0000, status: 0x0010
    {763}[Hardware Error]:   device_id: 0000:82:00.0
    {763}[Hardware Error]:   slot: 0
    {763}[Hardware Error]:   secondary_bus: 0x00
    {763}[Hardware Error]:   vendor_id: 0x8086, device_id: 0x10fb
```

```
    {763}[Hardware Error]:   class_code: 000002
    Kernel panic - not syncing: Fatal hardware error!

    This issue was imported by the patch, '37448adfc7ce ("aerdrv: Move
    cper_print_aer() call out of interrupt context")'. To fix this issue,
    this patch adds print of AER info in cper_print_pcie() for fatal error.

    Here is the example log after this patch applied:
    {24}[Hardware Error]: Hardware error from APEI Generic Hardware Error Source: 10
    {24}[Hardware Error]: event severity: fatal
    {24}[Hardware Error]:  Error 0, type: fatal
    {24}[Hardware Error]:   section_type: PCIe error
    {24}[Hardware Error]:   port_type: 0, PCIe end point
    {24}[Hardware Error]:   version: 4.0
    {24}[Hardware Error]:   command: 0x0546, status: 0x4010
    {24}[Hardware Error]:   device_id: 0000:01:00.0
    {24}[Hardware Error]:   slot: 0
    {24}[Hardware Error]:   secondary_bus: 0x00
    {24}[Hardware Error]:   vendor_id: 0x15b3, device_id: 0x1019
    {24}[Hardware Error]:   class_code: 000002
    {24}[Hardware Error]:   aer_uncor_status: 0x00040000, aer_uncor_mask: 0x00000000
    {24}[Hardware Error]:   aer_uncor_severity: 0x00062010
    {24}[Hardware Error]:   TLP Header: 000000c0 01010000 00000001 00000000
    Kernel panic - not syncing: Fatal hardware error!

    Fixes: 37448adfc7ce ("aerdrv: Move cper_print_aer() call out of interrupt context")
    Signed-off-by: Xiaofei Tan <tanxiaofei@huawei.com>
    Reviewed-by: James Morse <james.morse@arm.com>
    [ardb: put parens around terms of && operator]
    Signed-off-by: Ard Biesheuvel <ard.biesheuvel@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2dd043fd143197a62866faab6b593a441fdba6d2
Author: Stephen Douthit <stephend@silicom-usa.com>
Date:   Fri Aug 9 14:18:02 2019 +0000

    EDAC, pnd2: Fix ioremap() size in dnv_rd_reg()

    [ Upstream commit 29a3388bfcce7a6d087051376ea02bf8326a957b ]

    Depending on how BIOS has marked the reserved region containing the 32KB
    MCHBAR you can get warnings like:

    resource sanity check: requesting [mem 0xfed10000-0xfed1ffff], which spans more than
reserved [mem 0xfed10000-0xfed17fff]
    caller dnv_rd_reg+0xc8/0x240 [pnd2_edac] mapping multiple BARs

    Not all of the mmio regions used in dnv_rd_reg() are the same size.  The
    MCHBAR window is 32KB and the sideband ports are 64KB.  Pass the correct
    size to ioremap() depending on which resource we're reading from.

    Signed-off-by: Stephen Douthit <stephend@silicom-usa.com>
    Signed-off-by: Tony Luck <tony.luck@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 54f9a5fe6cd30fe0c06bad3aa2114f329d2cdfe2
Author: Luke Mujica <lukemujica@google.com>
Date:   Fri Jul 19 13:22:53 2019 -0700

    perf tools: Fix paths in include statements

    [ Upstream commit 2b75863b0845764529e01014a5c90664d8044cbe ]

    These paths point to the wrong location but still work because they get
    picked up by a -I flag that happens to direct to the correct file. Fix
```

```
    paths to lead to the actual file location without help from include
    flags.

    Signed-off-by: Luke Mujica <lukemujica@google.com>
    Cc: Alexander Shishkin <alexander.shishkin@linux.intel.com>
    Cc: Ian Rogers <irogers@google.com>
    Cc: Jiri Olsa <jolsa@redhat.com>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Cc: Stephane Eranian <eranian@google.com>
    Link: http://lkml.kernel.org/r/20190719202253.220261-1-lukemujica@google.com
    Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5de2b249b71026329b3a251b08f3a2d6b827c8e5
Author: Alessio Balsini <balsini@android.com>
Date:   Wed Aug 7 01:48:28 2019 +0100

    loop: Add LOOP_SET_DIRECT_IO to compat ioctl

    [ Upstream commit fdbe4eeeb1aac219b14f10c0ed31ae5d1123e9b8 ]

    Enabling Direct I/O with loop devices helps reducing memory usage by
    avoiding double caching.  32 bit applications running on 64 bits systems
    are currently not able to request direct I/O because is missing from the
    lo_compat_ioctl.

    This patch fixes the compatibility issue mentioned above by exporting
    LOOP_SET_DIRECT_IO as additional lo_compat_ioctl() entry.
    The input argument for this ioctl is a single long converted to a 1-bit
    boolean, so compatibility is preserved.

    Cc: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Alessio Balsini <balsini@android.com>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1bbd3c54593f7b8a386fd4e60671f3aa47cf9f45
Author: Jiri Slaby <jslaby@suse.cz>
Date:   Wed Aug 7 13:10:37 2019 +0200

    ACPI / processor: don't print errors for processorIDs == 0xff

    [ Upstream commit 2c2b005f549544c13ef4cfb0e4842949066889bc ]

    Some platforms define their processors in this manner:
        Device (SCK0)
        {
            Name (_HID, "ACPI0004" /* Module Device */)  // _HID: Hardware ID
            Name (_UID, "CPUSCK0")  // _UID: Unique ID
            Processor (CP00, 0x00, 0x00000410, 0x06){}
            Processor (CP01, 0x02, 0x00000410, 0x06){}
            Processor (CP02, 0x04, 0x00000410, 0x06){}
            Processor (CP03, 0x06, 0x00000410, 0x06){}
            Processor (CP04, 0x01, 0x00000410, 0x06){}
            Processor (CP05, 0x03, 0x00000410, 0x06){}
            Processor (CP06, 0x05, 0x00000410, 0x06){}
            Processor (CP07, 0x07, 0x00000410, 0x06){}
            Processor (CP08, 0xFF, 0x00000410, 0x06){}
            Processor (CP09, 0xFF, 0x00000410, 0x06){}
            Processor (CP0A, 0xFF, 0x00000410, 0x06){}
            Processor (CP0B, 0xFF, 0x00000410, 0x06){}
    ...

    The processors marked as 0xff are invalid, there are only 8 of them in
    this case.
```

```
    So do not print an error on ids == 0xff, just print an info message.
    Actually, we could return ENODEV even on the first CPU with ID 0xff, but
    ACPI spec does not forbid the 0xff value to be a processor ID. Given
    0xff could be a correct one, we would break working systems if we
    returned ENODEV.

    Signed-off-by: Jiri Slaby <jslaby@suse.cz>
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9135a04592ea90ca71114841bc4f06a3a62cc3cf
Author: Keyon Jie <yang.jie@linux.intel.com>
Date:   Wed Aug 7 09:50:30 2019 -0500

    ASoC: hdac_hda: fix page fault issue by removing race

    [ Upstream commit 804cbf4bb063204ca6c2471baa694548aab02ce3 ]

    There is a race between hda codec device removing and the
    jack-detecting work, which will lead to a page fault issue as the
    latter work is accessing codec device which could be already removed.

    Here add the cancellation of jack-detecting work before codecs are actually
    removed to avoid the race and fix the issue.

    Bug: https://github.com/thesofproject/linux/issues/1067
    Signed-off-by: Keyon Jie <yang.jie@linux.intel.com>
    Signed-off-by: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com>
    Link: https://lore.kernel.org/r/20190807145030.26117-1-pierre-
louis.bossart@linux.intel.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c8852b2a4082239dbd1faebb90ff477dd9965391
Author: Valdis Kletnieks <valdis.kletnieks@vt.edu>
Date:   Thu Aug 8 16:32:27 2019 +0200

    RAS: Build debugfs.o only when enabled in Kconfig

    [ Upstream commit b6ff24f7b5101101ff897dfdde3f37924e676bc2 ]

    In addition, the 0day bot reported this build error:

      >> drivers/ras/debugfs.c:10:5: error: redefinition of 'ras_userspace_consumers'
          int ras_userspace_consumers(void)
              ^~~~~~~~~~~~~~~~~~~~~~~~
         In file included from drivers/ras/debugfs.c:3:0:
         include/linux/ras.h:14:19: note: previous definition of 'ras_userspace_consumers'
was here
          static inline int ras_userspace_consumers(void) { return 0; }
                            ^~~~~~~~~~~~~~~~~~~~~~~~

    for a riscv-specific .config where CONFIG_DEBUG_FS is not set. Fix all
    that by making debugfs.o depend on that define.

     [ bp: Rewrite commit message. ]

    Reported-by: kbuild test robot <lkp@intel.com>
    Signed-off-by: Valdis Kletnieks <valdis.kletnieks@vt.edu>
    Signed-off-by: Borislav Petkov <bp@suse.de>
    Cc: Tony Luck <tony.luck@intel.com>
    Cc: linux-edac@vger.kernel.org
    Cc: x86@kernel.org
    Link: http://lkml.kernel.org/r/7053.1565218556@turing-police
```

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

commit 57e2eda9a744c5ce44ab56aa9f1d24092a27544b
Author: YueHaibing <yuehaibing@huawei.com>
Date:   Wed Jul 24 23:41:29 2019 -0300

    media: staging: tegra-vde: Fix build error

    [ Upstream commit 6b2265975239ab655069d796b822835a593e1cc7 ]

    If IOMMU_SUPPORT is not set, and COMPILE_TEST is y,
    IOMMU_IOVA may be set to m. So building will fails:

    drivers/staging/media/tegra-vde/iommu.o: In function `tegra_vde_iommu_map':
    iommu.c:(.text+0x41): undefined reference to `alloc_iova'
    iommu.c:(.text+0x56): undefined reference to `__free_iova'

    Select IOMMU_IOVA while COMPILE_TEST is set to fix this.

    Reported-by: Hulk Robot <hulkci@huawei.com>
    Suggested-by: Dmitry Osipenko <digetx@gmail.com>
    Fixes: b301f8de1925 ("media: staging: media: tegra-vde: Add IOMMU support")
    Signed-off-by: YueHaibing <yuehaibing@huawei.com>
    Acked-by: Dmitry Osipenko <digetx@gmail.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2e984ea2f81563b6620b2bdfe361ea69be9ad98d
Author: Randy Dunlap <rdunlap@infradead.org>
Date:   Tue Aug 6 19:01:36 2019 -0300

    media: media/platform: fsl-viu.c: fix build for MICROBLAZE

    [ Upstream commit 6898dd580a045341f844862ceb775144156ec1af ]

    arch/microblaze/ defines out_be32() and in_be32(), so don't do that
    again in the driver source.

    Fixes these build warnings:

    ../drivers/media/platform/fsl-viu.c:36: warning: "out_be32" redefined
    ../arch/microblaze/include/asm/io.h:50: note: this is the location of the previous
definition
    ../drivers/media/platform/fsl-viu.c:37: warning: "in_be32" redefined
    ../arch/microblaze/include/asm/io.h:53: note: this is the location of the previous
definition

    Fixes: 29d750686331 ("media: fsl-viu: allow building it with COMPILE_TEST")
    Signed-off-by: Randy Dunlap <rdunlap@infradead.org>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0e1b9820f5fb5c4b660e440b830fb91125591153
Author: Guoqing Jiang <jgq516@gmail.com>
Date:   Wed Jul 24 11:09:20 2019 +0200

    md: don't set In_sync if array is frozen

    [ Upstream commit 062f5b2ae12a153644c765e7ba3b0f825427be1d ]

    When a disk is added to array, the following path is called in mdadm.

    Manage_subdevs -> sysfs_freeze_array
```

```
                    -> Manage_add
                    -> sysfs_set_str(&info, NULL, "sync_action","idle")

    Then from kernel side, Manage_add invokes the path (add_new_disk ->
    validate_super = super_1_validate) to set In_sync flag.

    Since In_sync means "device is in_sync with rest of array", and the new
    added disk need to resync thread to help the synchronization of data.
    And md_reap_sync_thread would call spare_active to set In_sync for the
    new added disk finally. So don't set In_sync if array is in frozen.

    Signed-off-by: Guoqing Jiang <guoqing.jiang@cloud.ionos.com>
    Signed-off-by: Song Liu <songliubraving@fb.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9f356ce1bc6fd33e9bd953dafe3bc3424b63e50d
Author: Guoqing Jiang <jgq516@gmail.com>
Date:   Wed Jul 24 11:09:21 2019 +0200

    md: don't call spare_active in md_reap_sync_thread if all member devices can't work

    [ Upstream commit 0d8ed0e9bf9643f27f4816dca61081784dedb38d ]

    When add one disk to array, the md_reap_sync_thread is responsible
    to activate the spare and set In_sync flag for the new member in
    spare_active().

    But if raid1 has one member disk A, and disk B is added to the array.
    Then we offline A before all the datas are synchronized from A to B,
    obviously B doesn't have the latest data as A, but B is still marked
    with In_sync flag.

    So let's not call spare_active under the condition, otherwise B is
    still showed with 'U' state which is not correct.

    Signed-off-by: Guoqing Jiang <guoqing.jiang@cloud.ionos.com>
    Signed-off-by: Song Liu <songliubraving@fb.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 344242d50f468a250bf4b6136934392758412ed8
Author: Yufen Yu <yuyufen@huawei.com>
Date:   Fri Jul 19 13:48:46 2019 +0800

    md/raid1: end bio when the device faulty

    [ Upstream commit eeba6809d8d58908b5ed1b5ceb5fcb09a98a7cad ]

    When write bio return error, it would be added to conf->retry_list
    and wait for raid1d thread to retry write and acknowledge badblocks.

    In narrow_write_error(), the error bio will be split in the unit of
    badblock shift (such as one sector) and raid1d thread issues them
    one by one. Until all of the splited bio has finished, raid1d thread
    can go on processing other things, which is time consuming.

    But, there is a scene for error handling that is not necessary.
    When the device has been set faulty, flush_bio_list() may end
    bios in pending_bio_list with error status. Since these bios
    has not been issued to the device actually, error handlding to
    retry write and acknowledge badblocks make no sense.

    Even without that scene, when the device is faulty, badblocks info
    can not be written out to the device. Thus, we also no need to
    handle the error IO.
```

```
    Signed-off-by: Yufen Yu <yuyufen@huawei.com>
    Signed-off-by: Song Liu <songliubraving@fb.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 620f925861d3703c84b8d3407ee9a858cbcb43f5
Author: Qian Cai <cai@lca.pw>
Date:   Mon Aug 5 23:05:03 2019 -0400

    arm64/prefetch: fix a -Wtype-limits warning

    [ Upstream commit b99286b088ea843b935dcfb29f187697359fe5cd ]

    The commit d5370f754875 ("arm64: prefetch: add alternative pattern for
    CPUs without a prefetcher") introduced MIDR_IS_CPU_MODEL_RANGE() to be
    used in has_no_hw_prefetch() with rv_min=0 which generates a compilation
    warning from GCC,

    In file included from ./arch/arm64/include/asm/cache.h:8,
                     from ./include/linux/cache.h:6,
                     from ./include/linux/printk.h:9,
                     from ./include/linux/kernel.h:15,
                     from ./include/linux/cpumask.h:10,
                     from arch/arm64/kernel/cpufeature.c:11:
    arch/arm64/kernel/cpufeature.c: In function 'has_no_hw_prefetch':
    ./arch/arm64/include/asm/cputype.h:59:26: warning: comparison of
    unsigned expression >= 0 is always true [-Wtype-limits]
    _model == (model) && rv >= (rv_min) && rv <= (rv_max);  \
                              ^~
    arch/arm64/kernel/cpufeature.c:889:9: note: in expansion of macro
    'MIDR_IS_CPU_MODEL_RANGE'
    return MIDR_IS_CPU_MODEL_RANGE(midr, MIDR_THUNDERX,
           ^~~~~~~~~~~~~~~~~~~~~~~

    Fix it by converting MIDR_IS_CPU_MODEL_RANGE to a static inline
    function.

    Signed-off-by: Qian Cai <cai@lca.pw>
    Signed-off-by: Will Deacon <will@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8194280426df67b3a54e1cf84acb777ad280394d
Author: Kuninori Morimoto <kuninori.morimoto.gx@renesas.com>
Date:   Tue Aug 6 12:45:38 2019 +0900

    ASoC: rsnd: don't call clk_get_rate() under atomic context

    [ Upstream commit 06e8f5c842f2dbb232897ba967ea7b422745c271 ]

    ADG is using clk_get_rate() under atomic context, thus, we might
    have scheduling issue.
    To avoid this issue, we need to get/keep clk rate under
    non atomic context.

    We need to handle ADG as special device at Renesas Sound driver.
    From SW point of view, we want to impletent it as
    rsnd_mod_ops :: prepare, but it makes code just complicate.

    To avoid complicated code/patch, this patch adds new clk_rate[] array,
    and keep clk IN rate when rsnd_adg_clk_enable() was called.

    Reported-by: Leon Kong <Leon.KONG@cn.bosch.com>
    Signed-off-by: Kuninori Morimoto <kuninori.morimoto.gx@renesas.com>
    Tested-by: Leon Kong <Leon.KONG@cn.bosch.com>
    Link: https://lore.kernel.org/r/87v9vb0xkp.wl-kuninori.morimoto.gx@renesas.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
```

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

commit 9e5e4ecf382a8ba1091c6be672c8feb0d579b0f1
Author: Dan Carpenter <dan.carpenter@oracle.com>
Date:   Mon Jun 24 16:47:17 2019 +0300

    EDAC/altera: Use the proper type for the IRQ status bits

    [ Upstream commit 8faa1cf6ed82f33009f63986c3776cc48af1b7b2 ]

    Smatch complains about the cast of a u32 pointer to unsigned long:

      drivers/edac/altera_edac.c:1878 altr_edac_a10_irq_handler()
      warn: passing casted pointer '&irq_status' to 'find_first_bit()'

    This code wouldn't work on a 64 bit big endian system because it would
    read past the end of &irq_status.

     [ bp: massage. ]

    Fixes: 13ab8448d2c9 ("EDAC, altera: Add ECC Manager IRQ controller support")
    Signed-off-by: Dan Carpenter <dan.carpenter@oracle.com>
    Signed-off-by: Borislav Petkov <bp@suse.de>
    Reviewed-by: Thor Thayer <thor.thayer@linux.intel.com>
    Cc: James Morse <james.morse@arm.com>
    Cc: kernel-janitors@vger.kernel.org
    Cc: linux-edac <linux-edac@vger.kernel.org>
    Cc: Mauro Carvalho Chehab <mchehab@kernel.org>
    Cc: Tony Luck <tony.luck@intel.com>
    Link: https://lkml.kernel.org/r/20190624134717.GA1754@mwanda
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e25b7563e82e1dd8ac04cce3c27d099516b79195
Author: chenzefeng <chenzefeng2@huawei.com>
Date:   Tue Aug 6 15:46:33 2019 +0800

    ia64:unwind: fix double free for mod->arch.init_unw_table

    [ Upstream commit c5e5c48c16422521d363c33cfb0dcf58f88c119b ]

    The function free_module in file kernel/module.c as follow:

    void free_module(struct module *mod) {
            ......
            module_arch_cleanup(mod);
            ......
            module_arch_freeing_init(mod);
            ......
    }

    Both module_arch_cleanup and module_arch_freeing_init function
    would free the mod->arch.init_unw_table, which cause double free.

    Here, set mod->arch.init_unw_table = NULL after remove the unwind
    table to avoid double free.

    Signed-off-by: chenzefeng <chenzefeng2@huawei.com>
    Signed-off-by: Tony Luck <tony.luck@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2c065494301f102e120ec863c9b9566c25d06956
Author: Ard van Breemen <ard@kwaak.net>
Date:   Fri Aug 2 13:52:14 2019 +0200

    ALSA: usb-audio: Skip bSynchAddress endpoint check if it is invalid
```

```
    [ Upstream commit 1b34121d9f26d272b0b2334209af6b6fc82d4bf1 ]

    The Linux kernel assumes that get_endpoint(alts,0) and
    get_endpoint(alts,1) are eachothers feedback endpoints.
    To reassure that validity it will test bsynchaddress to comply with that
    assumption. But if the bsyncaddress is 0 (invalid), it will flag that as
    a wrong assumption and return an error.
    Fix: Skip the test if bSynchAddress is 0.
    Note: those with a valid bSynchAddress should have a code quirck added.

    Signed-off-by: Ard van Breemen <ard@kwaak.net>
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d284cf4c49c2d82f0b8eb83a96aa26aa46764e18
Author: Vinod Koul <vkoul@kernel.org>
Date:   Wed Jul 24 04:05:12 2019 +0530

    base: soc: Export soc_device_register/unregister APIs

    [ Upstream commit f7ccc7a397cf2ef64aebb2f726970b93203858d2 ]

    Qcom Socinfo driver can be built as a module, so
    export these two APIs.

    Tested-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Vaishali Thakkar <vaishali.thakkar@linaro.org>
    Reviewed-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Reviewed-by: Stephen Boyd <swboyd@chromium.org>
    Reviewed-by: Bjorn Andersson <bjorn.andersson@linaro.org>
    Signed-off-by: Bjorn Andersson <bjorn.andersson@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b997464eea0c471ae0d6ab5e4430ab72392a13af
Author: Neil Armstrong <narmstrong@baylibre.com>
Date:   Mon Jul 29 15:02:17 2019 +0200

    soc: amlogic: meson-clk-measure: protect measure with a mutex

    [ Upstream commit 3a760d986568b67d1f8411dab64608075817b90d ]

    In order to protect clock measuring when multiple process asks for
    a measure, protect the main measure function with mutexes.

    Reviewed-by: Kevin Hilman <khilman@baylibre.com>
    Reviewed-by: Martin Blumenstingl <martin.blumenstingl@googlemail.com>
    Signed-off-by: Neil Armstrong <narmstrong@baylibre.com>
    Signed-off-by: Kevin Hilman <khilman@baylibre.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5ffa57980798c9effdf329bb71961e7999c84ffd
Author: Junhua Huang <huang.junhua@zte.com.cn>
Date:   Sat Jul 6 14:41:15 2019 +0800

    arm64: mm: free the initrd reserved memblock in a aligned manner

    [ Upstream commit 13776f9d40a028a245bb766269e360f5b7a62721 ]

    We should free the initrd reserved memblock in an aligned manner,
    because the initrd reserves the memblock in an aligned manner
    in arm64_memblock_init().
    Otherwise there are some fragments in memblock_reserved regions
    after free_initrd_mem(). e.g.:
```

```
     /sys/kernel/debug/memblock # cat reserved
        0: 0x0000000080080000..0x00000000817fafff
        1: 0x0000000083400000..0x0000000083ffffff
        2: 0x0000000090000000..0x000000009000407f
        3: 0x00000000b0000000..0x00000000b000003f
        4: 0x00000000b26184ea..0x00000000b2618fff
     The fragments like the ranges from b0000000 to b000003f and
     from b26184ea to b2618fff should be freed.

     And we can do free_reserved_area() after memblock_free(),
     as free_reserved_area() calls __free_pages(), once we've done
     that it could be allocated somewhere else,
     but memblock and iomem still say this is reserved memory.

     Fixes: 05c58752f9dc ("arm64: To remove initrd reserved area entry from memblock")
     Signed-off-by: Junhua Huang <huang.junhua@zte.com.cn>
     Signed-off-by: Will Deacon <will@kernel.org>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 04b5983525daa0c3c1458d4fe94a2b154ea06d1c
Author: Charles Keepax <ckeepax@opensource.cirrus.com>
Date:   Mon Jul 22 10:07:48 2019 +0100

     gpio: madera: Add support for Cirrus Logic CS47L92

     [ Upstream commit 74d2d0e68701bcd53d2cf771dd3b3cb9f84bed5c ]

     As the gpio is common to all madera codecs all that is needed
     is to setup the correct number of GPIO pins for the CS47L92.

     Signed-off-by: Charles Keepax <ckeepax@opensource.cirrus.com>
     Link: https://lore.kernel.org/r/20190722090748.20807-4-ckeepax@opensource.cirrus.com
     Signed-off-by: Linus Walleij <linus.walleij@linaro.org>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5cdb1aa26a64250ec23b85f534aca1ba32115376
Author: Richard Fitzgerald <rf@opensource.cirrus.com>
Date:   Mon Jul 22 10:07:47 2019 +0100

     gpio: madera: Add support for Cirrus Logic CS47L15

     [ Upstream commit d06be8bc290aa255b9fd8602e60fb9f487aa0f48 ]

     As the gpio is common to all madera codecs all that is needed
     is to setup the correct number of GPIO pins for the CS47L15.

     Signed-off-by: Richard Fitzgerald <rf@opensource.cirrus.com>
     Signed-off-by: Charles Keepax <ckeepax@opensource.cirrus.com>
     Link: https://lore.kernel.org/r/20190722090748.20807-3-ckeepax@opensource.cirrus.com
     Signed-off-by: Linus Walleij <linus.walleij@linaro.org>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4c6f453bee59681311c014cc29eb253c8660ead9
Author: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
Date:   Fri Jul 19 12:12:42 2019 +0200

     cpuidle: teo: Allow tick to be stopped if PM QoS is used

     [ Upstream commit cab09f3d2d2a0a6cb3dfb678660d67a2c3764f50 ]

     The TEO goveror prevents the scheduler tick from being stopped (unless
     stopped already) if there is a PM QoS latency constraint for the given
     CPU and the target residency of the deepest idle state matching that
     constraint is below the tick boundary.
```

```
    However, that is problematic if CPUs with PM QoS latency constraints
    are idle for long times, because it effectively causes the tick to
    run on them all the time which is wasteful.  [It is also confusing
    and questionable if they are full dynticks CPUs.]

    To address that issue, modify the TEO governor to carry out the
    entire search for the most suitable idle state (from the target
    residency perspective) even if a latency constraint is present,
    to allow it to determine the expected idle duration in all cases.

    Also, when using the last several measured idle duration values
    to refine the idle state selection, make it compare those values
    with the current expected idle duration value (instead of
    comparing them with the target residency of the idle state
    selected so far) which should prevent the tick from being
    retained when it makes sense to stop it sometimes (especially
    in the presence of PM QoS latency constraints).

    Fixes: b26bf6ab716f ("cpuidle: New timer events oriented governor for tickless systems")
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5c9f1e66212363e1d06093dee983ccc4b7d82436
Author: Oliver Neukum <oneukum@suse.com>
Date:   Tue Jul 30 05:50:44 2019 -0300

    media: iguanair: add sanity checks

    [ Upstream commit ab1cbdf159beba7395a13ab70bc71180929ca064 ]

    The driver needs to check the endpoint types, too, as opposed
    to the number of endpoints. This also requires moving the check earlier.

    Reported-by: syzbot+01a77b82edaa374068e1@syzkaller.appspotmail.com
    Signed-off-by: Oliver Neukum <oneukum@suse.com>
    Signed-off-by: Sean Young <sean@mess.org>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f14bfe863ef94b0fcdaf0e5755bda935d91b3ebd
Author: Anson Huang <Anson.Huang@nxp.com>
Date:   Sat Jun 29 18:21:57 2019 +0800

    arm64: dts: imx8mq: Correct OPP table according to latest datasheet

    [ Upstream commit 9eced3a2f224a62a233761e8af18c907c532e192 ]

    According to latest datasheet (Rev.1, 10/2018) from below links,
    in the consumer datasheet, 1.5GHz is mentioned as highest opp but
    depends on speed grading fuse, and in the industrial datasheet,
    1.3GHz is mentioned as highest opp but depends on speed grading
    fuse. 1.5GHz and 1.3GHz opp use same voltage, so no need for
    consumer part to support 1.3GHz opp, with same voltage, CPU should
    run at highest frequency in order to go into idle as quick as
    possible, this can save power.

    That means for consumer part, 1GHz/1.5GHz are supported, for
    industrial part, 800MHz/1.3GHz are supported, and then check the
    speed grading fuse to limit the highest CPU frequency further.
    Correct the market segment bits in opp table to make them work
    according to datasheets.

    https://www.nxp.com/docs/en/data-sheet/IMX8MDQLQIEC.pdf
    https://www.nxp.com/docs/en/data-sheet/IMX8MDQLQCEC.pdf
```

```
    Fixes: 12629c5c3749 ("arm64: dts: imx8mq: Add cpu speed grading and all OPPs")
    Signed-off-by: Anson Huang <Anson.Huang@nxp.com>
    Reviewed-by: Leonard Crestez <leonard.crestez@nxp.com>
    Signed-off-by: Shawn Guo <shawnguo@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 58c5355b63b9eba540ae46fd7944834d1c3a74fe
Author: Robert Richter <rrichter@marvell.com>
Date:   Mon Jun 24 15:08:55 2019 +0000

    EDAC/mc: Fix grain_bits calculation

    [ Upstream commit 3724ace582d9f675134985727fd5e9811f23c059 ]

    The grain in EDAC is defined as "minimum granularity for an error
    report, in bytes". The following calculation of the grain_bits in
    edac_mc is wrong:

            grain_bits = fls_long(e->grain) + 1;

    Where grain_bits is defined as:

            grain = 1 << grain_bits

    Example:

            grain = 8       # 64 bit (8 bytes)
            grain_bits = fls_long(8) + 1
            grain_bits = 4 + 1 = 5

            grain = 1 << grain_bits
            grain = 1 << 5 = 32

    Replace it with the correct calculation:

            grain_bits = fls_long(e->grain - 1);

    The example gives now:

            grain_bits = fls_long(8 - 1)
            grain_bits = fls_long(7)
            grain_bits = 3

            grain = 1 << 3 = 8

    Also, check if the hardware reports a reasonable grain != 0 and fallback
    with a warning to 1 byte granularity otherwise.

     [ bp: massage a bit. ]

    Signed-off-by: Robert Richter <rrichter@marvell.com>
    Signed-off-by: Borislav Petkov <bp@suse.de>
    Cc: "linux-edac@vger.kernel.org" <linux-edac@vger.kernel.org>
    Cc: James Morse <james.morse@arm.com>
    Cc: Mauro Carvalho Chehab <mchehab@kernel.org>
    Cc: Tony Luck <tony.luck@intel.com>
    Link: https://lkml.kernel.org/r/20190624150758.6695-2-rrichter@marvell.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cd027ea82d264c849caf2d8461a0178be72c5bdd
Author: Paul E. McKenney <paulmck@linux.ibm.com>
Date:   Wed Jun 19 15:42:51 2019 -0700

    rcu: Add destroy_work_on_stack() to match INIT_WORK_ONSTACK()
```

```
    [ Upstream commit fbad01af8c3bb9618848abde8054ab7e0c2330fe ]

    The synchronize_rcu_expedited() function has an INIT_WORK_ONSTACK(),
    but lacks the corresponding destroy_work_on_stack().  This commit
    therefore adds destroy_work_on_stack().

    Reported-by: Andrea Arcangeli <aarcange@redhat.com>
    Signed-off-by: Paul E. McKenney <paulmck@linux.ibm.com>
    Acked-by: Andrea Arcangeli <aarcange@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ddfdd3173d8a764aeff17bd8fe9e22df2b7ceba4
Author: Jia-Ju Bai <baijiaju1990@gmail.com>
Date:   Fri Jul 26 10:14:42 2019 +0800

    ALSA: i2c: ak4xxx-adda: Fix a possible null pointer dereference in build_adc_controls()

    [ Upstream commit 2127c01b7f63b06a21559f56a8c81a3c6535bd1a ]

    In build_adc_controls(), there is an if statement on line 773 to check
    whether ak->adc_info is NULL:
        if (! ak->adc_info ||
            ! ak->adc_info[mixer_ch].switch_name)

    When ak->adc_info is NULL, it is used on line 792:
        knew.name = ak->adc_info[mixer_ch].selector_name;

    Thus, a possible null-pointer dereference may occur.

    To fix this bug, referring to lines 773 and 774, ak->adc_info
    and ak->adc_info[mixer_ch].selector_name are checked before being used.

    This bug is found by a static analysis tool STCheck written by us.

    Signed-off-by: Jia-Ju Bai <baijiaju1990@gmail.com>
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 90bff388574a9d1c4653af1c325430f8926a6671
Author: Takashi Iwai <tiwai@suse.de>
Date:   Fri Jul 26 11:42:34 2019 +0200

    ALSA: hda - Show the fatal CORB/RIRB error more clearly

    [ Upstream commit dd65f7e19c6961ba6a69f7c925021b7a270cb950 ]

    The last fallback of CORB/RIRB communication error recovery is to turn
    on the single command mode, and this last resort usually means that
    something is really screwed up.  Instead of a normal dev_err(), show
    the error more clearly with dev_WARN() with the caller stack trace.

    Also, show the bus-reset fallback also as an error, too.

    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b8b6fc5844f13da88990ec96aa8aee6c9b4c827c
Author: Thomas Gleixner <tglx@linutronix.de>
Date:   Mon Jul 22 20:47:08 2019 +0200

    x86/apic: Soft disable APIC before initializing it

    [ Upstream commit 2640da4cccf5cc613bf26f0998b9e340f4b5f69c ]

    If the APIC was already enabled on entry of setup_local_APIC() then
```

```
    disabling it soft via the SPIV register makes a lot of sense.

    That masks all LVT entries and brings it into a well defined state.

    Otherwise previously enabled LVTs which are not touched in the setup
    function stay unmasked and might surprise the just booting kernel.

    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Acked-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Link: https://lkml.kernel.org/r/20190722105219.068290579@linutronix.de
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1b2a1a99a361f91610493b158ffb13c66921939a
Author: Juri Lelli <juri.lelli@redhat.com>
Date:   Fri Jul 19 15:59:59 2019 +0200

    rcu/tree: Call setschedule() gp ktread to SCHED_FIFO outside of atomic region

    [ Upstream commit 1a763fd7c6335e3122c1cc09576ef6c99ada4267 ]

    sched_setscheduler() needs to acquire cpuset_rwsem, but it is currently
    called from an invalid (atomic) context by rcu_spawn_gp_kthread().

    Fix that by simply moving sched_setscheduler_nocheck() call outside of
    the atomic region, as it doesn't actually require to be guarded by
    rcu_node lock.

    Suggested-by: Peter Zijlstra <peterz@infradead.org>
    Tested-by: Dietmar Eggemann <dietmar.eggemann@arm.com>
    Signed-off-by: Juri Lelli <juri.lelli@redhat.com>
    Signed-off-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Cc: Linus Torvalds <torvalds@linux-foundation.org>
    Cc: Thomas Gleixner <tglx@linutronix.de>
    Cc: bristot@redhat.com
    Cc: claudio@evidence.eu.com
    Cc: lizefan@huawei.com
    Cc: longman@redhat.com
    Cc: luca.abeni@santannapisa.it
    Cc: mathieu.poirier@linaro.org
    Cc: rostedt@goodmis.org
    Cc: tj@kernel.org
    Cc: tommaso.cucinotta@santannapisa.it
    Link: https://lkml.kernel.org/r/20190719140000.31694-8-juri.lelli@redhat.com
    Signed-off-by: Ingo Molnar <mingo@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 387d74e9bdb0f9c6f2eaa443e6f9b2f4a9d10714
Author: Grzegorz Halat <ghalat@redhat.com>
Date:   Fri Jun 28 14:28:13 2019 +0200

    x86/reboot: Always use NMI fallback when shutdown via reboot vector IPI fails

    [ Upstream commit 747d5a1bf293dcb33af755a6d285d41b8c1ea010 ]

    A reboot request sends an IPI via the reboot vector and waits for all other
    CPUs to stop. If one or more CPUs are in critical regions with interrupts
    disabled then the IPI is not handled on those CPUs and the shutdown hangs
    if native_stop_other_cpus() is called with the wait argument set.

    Such a situation can happen when one CPU was stopped within a lock held
    section and another CPU is trying to acquire that lock with interrupts
    disabled. There are other scenarios which can cause such a lockup as well.

    In theory the shutdown should be attempted by an NMI IPI after the timeout
    period elapsed. Though the wait loop after sending the reboot vector IPI
```

```
    prevents this. It checks the wait request argument and the timeout. If wait
    is set, which is true for sys_reboot() then it won't fall through to the
    NMI shutdown method after the timeout period has finished.

    This was an oversight when the NMI shutdown mechanism was added to handle
    the 'reboot IPI is not working' situation. The mechanism was added to deal
    with stuck panic shutdowns, which do not have the wait request set, so the
    'wait request' case was probably not considered.

    Remove the wait check from the post reboot vector IPI wait loop and enforce
    that the wait loop in the NMI fallback path is invoked even if NMI IPIs are
    disabled or the registration of the NMI handler fails. That second wait
    loop will then hang if not all CPUs shutdown and the wait argument is set.

    [ tglx: Avoid the hard to parse line break in the NMI fallback path,
            add comments and massage the changelog ]

    Fixes: 7d007d21e539 ("x86/reboot: Use NMI to assist in shutting down if IRQ fails")
    Signed-off-by: Grzegorz Halat <ghalat@redhat.com>
    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Cc: Don Zickus <dzickus@redhat.com>
    Link: https://lkml.kernel.org/r/20190628122813.15500-1-ghalat@redhat.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 92ead878755b7a5c04691249dcf7e00ecc746bbe
Author: Juri Lelli <juri.lelli@redhat.com>
Date:   Fri Jul 19 15:59:56 2019 +0200

    sched/deadline: Fix bandwidth accounting at all levels after offline migration

    [ Upstream commit 59d06cea1198d665ba11f7e8c5f45b00ff2e4812 ]

    If a task happens to be throttled while the CPU it was running on gets
    hotplugged off, the bandwidth associated with the task is not correctly
    migrated with it when the replenishment timer fires (offline_migration).

    Fix things up, for this_bw, running_bw and total_bw, when replenishment
    timer fires and task is migrated (dl_task_offline_migration()).

    Tested-by: Dietmar Eggemann <dietmar.eggemann@arm.com>
    Signed-off-by: Juri Lelli <juri.lelli@redhat.com>
    Signed-off-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Cc: Linus Torvalds <torvalds@linux-foundation.org>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Cc: Thomas Gleixner <tglx@linutronix.de>
    Cc: bristot@redhat.com
    Cc: claudio@evidence.eu.com
    Cc: lizefan@huawei.com
    Cc: longman@redhat.com
    Cc: luca.abeni@santannapisa.it
    Cc: mathieu.poirier@linaro.org
    Cc: rostedt@goodmis.org
    Cc: tj@kernel.org
    Cc: tommaso.cucinotta@santannapisa.it
    Link: https://lkml.kernel.org/r/20190719140000.31694-5-juri.lelli@redhat.com
    Signed-off-by: Ingo Molnar <mingo@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dcf395c38dca640a55d97ee83a49e7ae7710034d
Author: Thomas Gleixner <tglx@linutronix.de>
Date:   Mon Jul 22 20:47:09 2019 +0200

    x86/apic: Make apic_pending_intr_clear() more robust

    [ Upstream commit cc8bf191378c1da8ad2b99cf470ee70193ace84e ]
```

```
    In course of developing shorthand based IPI support issues with the
    function which tries to clear eventually pending ISR bits in the local APIC
    were observed.

      1) O-day testing triggered the WARN_ON() in apic_pending_intr_clear().

         This warning is emitted when the function fails to clear pending ISR
         bits or observes pending IRR bits which are not delivered to the CPU
         after the stale ISR bit(s) are ACK'ed.

         Unfortunately the function only emits a WARN_ON() and fails to dump
         the IRR/ISR content. That's useless for debugging.

         Feng added spot on debug printk's which revealed that the stale IRR
         bit belonged to the APIC timer interrupt vector, but adding ad hoc
         debug code does not help with sporadic failures in the field.

         Rework the loop so the full IRR/ISR contents are saved and on failure
         dumped.

      2) The loop termination logic is interesting at best.

         If the machine has no TSC or cpu_khz is not known yet it tries 1
         million times to ack stale IRR/ISR bits. What?

         With TSC it uses the TSC to calculate the loop termination. It takes a
         timestamp at entry and terminates the loop when:

              (rdtsc() - start_timestamp) >= (cpu_hkz << 10)

         That's roughly one second.

         Both methods are problematic. The APIC has 256 vectors, which means
         that in theory max. 256 IRR/ISR bits can be set. In practice this is
         impossible and the chance that more than a few bits are set is close
         to zero.

         With the pure loop based approach the 1 million retries are complete
         overkill.

         With TSC this can terminate too early in a guest which is running on a
         heavily loaded host even with only a couple of IRR/ISR bits set. The
         reason is that after acknowledging the highest priority ISR bit,
         pending IRRs must get serviced first before the next round of
         acknowledge can take place as the APIC (real and virtualized) does not
         honour EOI without a preceeding interrupt on the CPU. And every APIC
         read/write takes a VMEXIT if the APIC is virtualized. While trying to
         reproduce the issue 0-day reported it was observed that the guest was
         scheduled out long enough under heavy load that it terminated after 8
         iterations.

         Make the loop terminate after 512 iterations. That's plenty enough
         in any case and does not take endless time to complete.

    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Acked-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Link: https://lkml.kernel.org/r/20190722105219.158847694@linutronix.de
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 57cb8b92df7f707582e002db6affd4dd14bd71ac
Author: Juri Lelli <juri.lelli@redhat.com>
Date:   Fri Jul 19 08:34:55 2019 +0200

    sched/core: Fix CPU controller for !RT_GROUP_SCHED
```

```
    [ Upstream commit a07db5c0865799ebed1f88be0df50c581fb65029 ]

    On !CONFIG_RT_GROUP_SCHED configurations it is currently not possible to
    move RT tasks between cgroups to which CPU controller has been attached;
    but it is oddly possible to first move tasks around and then make them
    RT (setschedule to FIFO/RR).

    E.g.:

      # mkdir /sys/fs/cgroup/cpu,cpuacct/group1
      # chrt -fp 10 $$
      # echo $$ > /sys/fs/cgroup/cpu,cpuacct/group1/tasks
      bash: echo: write error: Invalid argument
      # chrt -op 0 $$
      # echo $$ > /sys/fs/cgroup/cpu,cpuacct/group1/tasks
      # chrt -fp 10 $$
      # cat /sys/fs/cgroup/cpu,cpuacct/group1/tasks
      2345
      2598
      # chrt -p 2345
      pid 2345's current scheduling policy: SCHED_FIFO
      pid 2345's current scheduling priority: 10

    Also, as Michal noted, it is currently not possible to enable CPU
    controller on unified hierarchy with !CONFIG_RT_GROUP_SCHED (if there
    are any kernel RT threads in root cgroup, they can't be migrated to the
    newly created CPU controller's root in cgroup_update_dfl_csses()).

    Existing code comes with a comment saying the "we don't support RT-tasks
    being in separate groups". Such comment is however stale and belongs to
    pre-RT_GROUP_SCHED times. Also, it doesn't make much sense for
    !RT_GROUP_ SCHED configurations, since checks related to RT bandwidth
    are not performed at all in these cases.

    Make moving RT tasks between CPU controller groups viable by removing
    special case check for RT (and DEADLINE) tasks.

    Signed-off-by: Juri Lelli <juri.lelli@redhat.com>
    Signed-off-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Reviewed-by: Michal Koutný <mkoutny@suse.com>
    Reviewed-by: Daniel Bristot de Oliveira <bristot@redhat.com>
    Acked-by: Tejun Heo <tj@kernel.org>
    Cc: Linus Torvalds <torvalds@linux-foundation.org>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Cc: Thomas Gleixner <tglx@linutronix.de>
    Cc: lizefan@huawei.com
    Cc: longman@redhat.com
    Cc: luca.abeni@santannapisa.it
    Cc: rostedt@goodmis.org
    Link: https://lkml.kernel.org/r/20190719063455.27328-1-juri.lelli@redhat.com
    Signed-off-by: Ingo Molnar <mingo@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ce3244a39199a366db328639d7d610aea849efda
Author: Vincent Guittot <vincent.guittot@linaro.org>
Date:   Mon Jul 1 17:47:02 2019 +0200

    sched/fair: Fix imbalance due to CPU affinity

    [ Upstream commit f6cad8df6b30a5d2bbbd2e698f74b4cafb9fb82b ]

    The load_balance() has a dedicated mecanism to detect when an imbalance
    is due to CPU affinity and must be handled at parent level. In this case,
    the imbalance field of the parent's sched_group is set.
```

```
    The description of sg_imbalanced() gives a typical example of two groups
    of 4 CPUs each and 4 tasks each with a cpumask covering 1 CPU of the first
    group and 3 CPUs of the second group. Something like:

            { 0 1 2 3 } { 4 5 6 7 }
                    *     * * *

    But the load_balance fails to fix this UC on my octo cores system
    made of 2 clusters of quad cores.

    Whereas the load_balance is able to detect that the imbalanced is due to
    CPU affinity, it fails to fix it because the imbalance field is cleared
    before letting parent level a chance to run. In fact, when the imbalance is
    detected, the load_balance reruns without the CPU with pinned tasks. But
    there is no other running tasks in the situation described above and
    everything looks balanced this time so the imbalance field is immediately
    cleared.

    The imbalance field should not be cleared if there is no other task to move
    when the imbalance is detected.

    Signed-off-by: Vincent Guittot <vincent.guittot@linaro.org>
    Signed-off-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Cc: Linus Torvalds <torvalds@linux-foundation.org>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Cc: Thomas Gleixner <tglx@linutronix.de>
    Link: https://lkml.kernel.org/r/1561996022-28829-1-git-send-email-
vincent.guittot@linaro.org
    Signed-off-by: Ingo Molnar <mingo@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b112fe3c4db30156418a2df751cfc47c35f0cebf
Author: Paul E. McKenney <paulmck@linux.ibm.com>
Date:   Tue Jun 25 09:52:38 2019 -0700

    time/tick-broadcast: Fix tick_broadcast_offline() lockdep complaint

    [ Upstream commit 84ec3a0787086fcd25f284f59b3aa01fd6fc0a5d ]

    time/tick-broadcast: Fix tick_broadcast_offline() lockdep complaint

    The TASKS03 and TREE04 rcutorture scenarios produce the following
    lockdep complaint:

            WARNING: inconsistent lock state
            5.2.0-rc1+ #513 Not tainted
            --------------------------------
            inconsistent {IN-HARDIRQ-W} -> {HARDIRQ-ON-W} usage.
            migration/1/14 [HC0[0]:SC0[0]:HE1:SE1] takes:
            (____ptrval____) (tick_broadcast_lock){?...}, at:
tick_broadcast_offline+0xf/0x70
            {IN-HARDIRQ-W} state was registered at:
              lock_acquire+0xb0/0x1c0
              _raw_spin_lock_irqsave+0x3c/0x50
              tick_broadcast_switch_to_oneshot+0xd/0x40
              tick_switch_to_oneshot+0x4f/0xd0
              hrtimer_run_queues+0xf3/0x130
              run_local_timers+0x1c/0x50
              update_process_times+0x1c/0x50
              tick_periodic+0x26/0xc0
              tick_handle_periodic+0x1a/0x60
              smp_apic_timer_interrupt+0x80/0x2a0
              apic_timer_interrupt+0xf/0x20
              _raw_spin_unlock_irqrestore+0x4e/0x60
```

```
               rcu_nocb_gp_kthread+0x15d/0x590
               kthread+0xf3/0x130
               ret_from_fork+0x3a/0x50
             irq event stamp: 171
             hardirqs last  enabled at (171): [<ffffffff8a201a37>]
  trace_hardirqs_on_thunk+0x1a/0x1c
             hardirqs last disabled at (170): [<ffffffff8a201a53>]
  trace_hardirqs_off_thunk+0x1a/0x1c
             softirqs last  enabled at (0): [<ffffffff8a264ee0>]
  copy_process.part.56+0x650/0x1cb0
             softirqs last disabled at (0): [<0000000000000000>] 0x0

             [...]

     To reproduce, run the following rcutorture test:

      $ tools/testing/selftests/rcutorture/bin/kvm.sh --duration 5 --kconfig
  "CONFIG_DEBUG_LOCK_ALLOC=y CONFIG_PROVE_LOCKING=y" --configs "TASKS03 TREE04"

     It turns out that tick_broadcast_offline() was an innocent bystander.
     After all, interrupts are supposed to be disabled throughout
     take_cpu_down(), and therefore should have been disabled upon entry to
     tick_offline_cpu() and thus to tick_broadcast_offline().  This suggests
     that one of the CPU-hotplug notifiers was incorrectly enabling interrupts,
     and leaving them enabled on return.

     Some debugging code showed that the culprit was sched_cpu_dying().
     It had irqs enabled after return from sched_tick_stop().  Which in turn
     had irqs enabled after return from cancel_delayed_work_sync().  Which is a
     wrapper around __cancel_work_timer().  Which can sleep in the case where
     something else is concurrently trying to cancel the same delayed work,
     and as Thomas Gleixner pointed out on IRC, sleeping is a decidedly bad
     idea when you are invoked from take_cpu_down(), regardless of the state
     you leave interrupts in upon return.

     Code inspection located no reason why the delayed work absolutely
     needed to be canceled from sched_tick_stop():  The work is not
     bound to the outgoing CPU by design, given that the whole point is
     to collect statistics without disturbing the outgoing CPU.

     This commit therefore simply drops the cancel_delayed_work_sync() from
     sched_tick_stop().  Instead, a new ->state field is added to the tick_work
     structure so that the delayed-work handler function sched_tick_remote()
     can avoid reposting itself.  A cpu_is_offline() check is also added to
     sched_tick_remote() to avoid mucking with the state of an offlined CPU
     (though it does appear safe to do so).  The sched_tick_start() and
     sched_tick_stop() functions also update ->state, and sched_tick_start()
     also schedules the delayed work if ->state indicates that it is not
     already in flight.

     Signed-off-by: Paul E. McKenney <paulmck@linux.ibm.com>
     [ paulmck: Apply Peter Zijlstra and Frederic Weisbecker atomics feedback. ]
     Signed-off-by: Peter Zijlstra (Intel) <peterz@infradead.org>
     Reviewed-by: Frederic Weisbecker <frederic@kernel.org>
     Cc: Linus Torvalds <torvalds@linux-foundation.org>
     Cc: Peter Zijlstra <peterz@infradead.org>
     Cc: Thomas Gleixner <tglx@linutronix.de>
     Link: https://lkml.kernel.org/r/20190625165238.GJ26519@linux.ibm.com
     Signed-off-by: Ingo Molnar <mingo@kernel.org>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 346c4b44b48041619a320db77161d7c055a872ae
  Author: Fabio Estevam <festevam@gmail.com>
  Date:   Fri Jun 28 07:00:34 2019 -0400
```

```
    media: i2c: ov5640: Check for devm_gpiod_get_optional() error

    [ Upstream commit 8791a102ce579346cea9d2f911afef1c1985213c ]

    The power down and reset GPIO are optional, but the return value
    from devm_gpiod_get_optional() needs to be checked and propagated
    in the case of error, so that probe deferral can work.

    Signed-off-by: Fabio Estevam <festevam@gmail.com>
    Signed-off-by: Sakari Ailus <sakari.ailus@linux.intel.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 06fb82ee180d8456f1dd54b73b51fcf762d6f2b7
Author: Luke Nowakowski-Krijger <lnowakow@eng.ucsd.edu>
Date:   Wed Jul 17 10:19:46 2019 -0400

    media: hdpvr: Add device num check and handling

    [ Upstream commit d4a6a9537bc32811486282206ecfb7c53754b74d ]

    Add hdpvr device num check and error handling

    We need to increment the device count atomically before we checkout a
    device to make sure that we do not reach the max count, otherwise we get
    out-of-bounds errors as reported by syzbot.

    Reported-and-tested-by: syzbot+aac8d0d7205f112045d2@syzkaller.appspotmail.com

    Signed-off-by: Luke Nowakowski-Krijger <lnowakow@eng.ucsd.edu>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4e666d94e36aa12fca81e65521a4f8c7bba2a8d3
Author: Arnd Bergmann <arnd@arndb.de>
Date:   Thu Jul 18 10:16:43 2019 -0400

    media: vivid: work around high stack usage with clang

    [ Upstream commit 1a03f91c2c2419c3709c4554952c66695575e91c ]

    Building a KASAN-enabled kernel with clang ends up in a case where too
    much is inlined into vivid_thread_vid_cap() and the stack usage grows
    a lot, possibly when the register allocation fails to produce efficient
    code and spills a lot of temporaries to the stack. This uses more
    than twice the amount of stack than the sum of the individual functions
    when they are not inlined:

    drivers/media/platform/vivid/vivid-kthread-cap.c:766:12: error: stack frame size of 2208
bytes in function 'vivid_thread_vid_cap' [-Werror,-Wframe-larger-than=]

    Marking two of the key functions in here as 'noinline_for_stack' avoids
    the pathological case in clang without any apparent downside for gcc.

    Signed-off-by: Arnd Bergmann <arnd@arndb.de>
    Acked-by: Nick Desaulniers <ndesaulniers@google.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 13321882232bff1deec3dbbdab645d1caf7d5b00
Author: Michael Tretter <m.tretter@pengutronix.de>
Date:   Thu Jun 27 08:44:32 2019 -0400
```

```
    media: vb2: reorder checks in vb2_poll()

    [ Upstream commit 8d86a15649957c182e90fa2b1267c16699bc12f1 ]

    When reaching the end of stream, V4L2 clients may expect the
    V4L2_EOS_EVENT before being able to dequeue the last buffer, which has
    the V4L2_BUF_FLAG_LAST flag set.

    If the vb2_poll() function first checks for events and afterwards if
    buffers are available, a driver can queue the V4L2_EOS_EVENT event and
    return the buffer after the check for events but before the check for
    buffers. This causes vb2_poll() to signal that the buffer with
    V4L2_BUF_FLAG_LAST can be read without the V4L2_EOS_EVENT being
    available.

    First, check for available buffers and afterwards for events to ensure
    that if vb2_poll() signals POLLIN | POLLRDNORM for the
    V4L2_BUF_FLAG_LAST buffer, it also signals POLLPRI for the
    V4L2_EOS_EVENT.

    Signed-off-by: Michael Tretter <m.tretter@pengutronix.de>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a3863512341aa67eda5531181676423cb8a6cf05
Author: Vandana BN <bnvandana@gmail.com>
Date:   Thu Jun 27 04:26:43 2019 -0400

    media: vivid:add sanity check to avoid divide error and set value to 1 if 0.

    [ Upstream commit aa9c2182c45421d54ed27c2a1765f7adedce291b ]

    Syzbot reported divide error in vivid_thread_vid_cap, which has been
    seen only once and does not have a reproducer.
    This patch adds sanity checks for the
    denominator value with WARN_ON if it is 0 and replaces it with 1.

    divide error: 0000 [#1] PREEMPT SMP KASAN
    kobject: 'tx-0' (0000000017161f7f): kobject_uevent_env
    CPU: 0 PID: 23689 Comm: vivid-003-vid-c Not tainted 5.0.0-rc4+ #58
    Hardware name: Google Google Compute Engine/Google Compute Engine, BIOS
    Google 01/01/2011
    RIP: 0010:vivid_cap_update_frame_period
    drivers/media/platform/vivid/vivid-kthread-cap.c:661 [inline]
    RIP: 0010:vivid_thread_vid_cap+0x221/0x284
    drivers/media/platform/vivid/vivid-kthread-cap.c:789
    Code: 48 c1 e9 03 0f b6 0c 11 48 89 f2 48 69 c0 00 ca 9a 3b 83 c2 03 38
    ca
    7c 08 84 c9 0f 85 f0 1e 00 00 41 8b 8f 24 64 00 00 31 d2 <48> f7 f1 49
    89
    c4 48 89 c3 49 8d 87 28 64 00 00 48 89 c2 48 89 45
    RSP: 0018:ffff88808b4afd68 EFLAGS: 00010246
    kobject: 'tx-0' (0000000017161f7f): fill_kobj_path: path
    = '/devices/virtual/net/gre0/queues/tx-0'
    RAX: 000000de5a6f8e00 RBX: 0000000100047b22 RCX: 0000000000000000
    RDX: 0000000000000000 RSI: 0000000000000004 RDI: 0000000000000004
    RBP: ffff88808b4aff00 R08: ffff88804862e1c0 R09: ffffffff89997008
    R10: ffffffff89997010 R11: 0000000000000001 R12: 00000000fffffffc
    R13: ffff8880a17e0500 R14: ffff88803e40f760 R15: ffff8882182b0140
    FS:  0000000000000000(0000) GS:ffff8880ae800000(0000)
    knlGS:0000000000000000
    CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    CR2: 00000000004cdc90 CR3: 000000005d827000 CR4: 00000000001426f0
    Call Trace:
```

```
    kobject: 'gretap0' (00000000d7549098): kobject_add_internal: parent:
    'net',
    set: 'devices'
    kobject: 'loop2' (0000000094ed4ee4): kobject_uevent_env
    kobject: 'loop2' (0000000094ed4ee4): fill_kobj_path: path
    = '/devices/virtual/block/loop2'
      kthread+0x357/0x430 kernel/kthread.c:246
    kobject: 'gretap0' (00000000d7549098): kobject_uevent_env
      ret_from_fork+0x3a/0x50 arch/x86/entry/entry_64.S:352
    Modules linked in:
    kobject: 'gretap0' (00000000d7549098): fill_kobj_path: path
    = '/devices/virtual/net/gretap0'
    ---[ end trace bc5c8b25b64d768f ]---
    kobject: 'loop1' (0000000032036b86): kobject_uevent_env
    RIP: 0010:vivid_cap_update_frame_period
    drivers/media/platform/vivid/vivid-kthread-cap.c:661 [inline]
    RIP: 0010:vivid_thread_vid_cap+0x221/0x2840
    drivers/media/platform/vivid/vivid-kthread-cap.c:789
    kobject: 'loop1' (0000000032036b86): fill_kobj_path: path
    = '/devices/virtual/block/loop1'
    Code: 48 c1 e9 03 0f b6 0c 11 48 89 f2 48 69 c0 00 ca 9a 3b 83 c2 03 38
    ca
    7c 08 84 c9 0f 85 f0 1e 00 00 41 8b 8f 24 64 00 00 31 d2 <48> f7 f1 49
    89
    c4 48 89 c3 49 8d 87 28 64 00 00 48 89 c2 48 89 45
    kobject: 'loop0' (00000000dd9927c3): kobject_uevent_env
    RSP: 0018:ffff88808b4afd68 EFLAGS: 00010246
    RAX: 000000de5a6f8e00 RBX: 0000000100047b22 RCX: 0000000000000000
    kobject: 'queues' (000000007ed20666): kobject_add_internal:
    parent: 'gretap0', set: '<NULL>'
    RDX: 0000000000000000 RSI: 0000000000000004 RDI: 0000000000000004
    RBP: ffff88808b4aff00 R08: ffff88804862e1c0 R09: ffffffff89997008
    kobject: 'loop0' (00000000dd9927c3): fill_kobj_path: path
    = '/devices/virtual/block/loop0'
    R10: ffffffff89997010 R11: 0000000000000001 R12: 00000000fffffffc
    kobject: 'queues' (000000007ed20666): kobject_uevent_env
    R13: ffff8880a17e0500 R14: ffff88803e40f760 R15: ffff8882182b0140
    FS:  0000000000000000(0000) GS:ffff8880ae800000(0000)
    knlGS:0000000000000000
    kobject: 'loop5' (00000000a41f9e79): kobject_uevent_env
    CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    kobject: 'queues' (000000007ed20666): kobject_uevent_env: filter
    function
    caused the event to drop!
    CR2: 00000000004cdc90 CR3: 000000005d827000 CR4: 00000000001426f0
    kobject: 'loop5' (00000000a41f9e79): fill_kobj_path: path
    = '/devices/virtual/block/loop5'

    Reported-by: syz...@syzkaller.appspotmail.com
    Signed-off-by: Vandana BN <bnvandana@gmail.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 83590d1f2eac038a046cad1ff0da775a4551707e
Author: Wen Yang <wen.yang99@zte.com.cn>
Date:   Thu Jun 27 23:01:15 2019 -0400

    media: exynos4-is: fix leaked of_node references

    [ Upstream commit da79bf41a4d170ca93cc8f3881a70d734a071c37 ]

    The call to of_get_child_by_name returns a node pointer with refcount
    incremented thus it must be explicitly decremented after the last
    usage.
```

```
    Detected by coccinelle with the following warnings:
    drivers/media/platform/exynos4-is/fimc-is.c:813:2-8: ERROR: missing of_node_put;
acquired a node pointer with refcount incremented on line 807, but without a corresponding
object release within this function.
    drivers/media/platform/exynos4-is/fimc-is.c:870:1-7: ERROR: missing of_node_put;
acquired a node pointer with refcount incremented on line 807, but without a corresponding
object release within this function.
    drivers/media/platform/exynos4-is/fimc-is.c:885:1-7: ERROR: missing of_node_put;
acquired a node pointer with refcount incremented on line 807, but without a corresponding
object release within this function.
    drivers/media/platform/exynos4-is/media-dev.c:545:1-7: ERROR: missing of_node_put;
acquired a node pointer with refcount incremented on line 541, but without a corresponding
object release within this function.
    drivers/media/platform/exynos4-is/media-dev.c:528:1-7: ERROR: missing of_node_put;
acquired a node pointer with refcount incremented on line 499, but without a corresponding
object release within this function.
    drivers/media/platform/exynos4-is/media-dev.c:534:1-7: ERROR: missing of_node_put;
acquired a node pointer with refcount incremented on line 499, but without a corresponding
object release within this function.

    Signed-off-by: Wen Yang <wen.yang99@zte.com.cn>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6d1e2de93a52e0db237d5b384474168c2fa2e78a
Author: Pan Xiuli <xiuli.pan@linux.intel.com>
Date:   Mon Jul 22 09:13:42 2019 -0500

    ASoC: SOF: pci: mark last_busy value at runtime PM init

    [ Upstream commit f1b1b9b136827915624136624ff54aba5890a15b ]

    If last_busy value is not set at runtime PM enable, the device will be
    suspend immediately after usage counter is 0. Set the last_busy value to
    make sure delay is working at first boot up.

    Signed-off-by: Pan Xiuli <xiuli.pan@linux.intel.com>
    Signed-off-by: Ranjani Sridharan <ranjani.sridharan@linux.intel.com>
    Signed-off-by: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com>
    Link: https://lore.kernel.org/r/20190722141402.7194-2-pierre-
louis.bossart@linux.intel.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6c79b1e3180ac4e70dab7609ef4a99ed4b703431
Author: Sean Young <sean@mess.org>
Date:   Fri Jul 12 18:47:00 2019 -0400

    media: mtk-cir: lower de-glitch counter for rc-mm protocol

    [ Upstream commit 5dd4b89dc098bf22cd13e82a308f42a02c102b2b ]

    The rc-mm protocol can't be decoded by the mtk-cir since the de-glitch
    filter removes pulses/spaces shorter than 294 microseconds.

    Tested on a BananaPi R2.

    Signed-off-by: Sean Young <sean@mess.org>
    Acked-by: Sean Wang <sean.wang@kernel.org>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9d77b64f896f0976bdc784c06102fb8ef8bcfd5c
```

```
Author: Arnd Bergmann <arnd@arndb.de>
Date:   Fri Jun 28 08:14:53 2019 -0400

    media: dib0700: fix link error for dibx000_i2c_set_speed

    [ Upstream commit 765bb8610d305ee488b35d07e2a04ae52fb2df9c ]

    When CONFIG_DVB_DIB9000 is disabled, we can still compile code that
    now fails to link against dibx000_i2c_set_speed:

    drivers/media/usb/dvb-usb/dib0700_devices.o: In function
`dib01x0_pmu_update.constprop.7':
    dib0700_devices.c:(.text.unlikely+0x1c9c): undefined reference to
`dibx000_i2c_set_speed'

    The call sites are both through dib01x0_pmu_update(), which gets passed
    an 'i2c' pointer from dib9000_get_i2c_master(), which has returned
    NULL. Checking this pointer seems to be a good idea anyway, and it avoids
    the link failure in most cases.

    Sean Young found another case that is not fixed by that, where certain
    gcc versions leave an unused function in place that causes the link error,
    but adding an explict IS_ENABLED() check also solves this.

    Fixes: b7f54910ce01 ("V4L/DVB (4647): Added module for DiB0700 based devices")
    Signed-off-by: Arnd Bergmann <arnd@arndb.de>
    Signed-off-by: Sean Young <sean@mess.org>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 45471fab16353eb62bccfff73a9a1e2046f8f110
Author: Kai Vehmanen <kai.vehmanen@linux.intel.com>
Date:   Mon Jul 22 09:13:43 2019 -0500

    ASoC: SOF: reset DMA state in prepare

    [ Upstream commit 04c8027764bc82a325d3abc6f39a6a4642a937cb ]

    When application goes through SUSPEND/STOP->PREPARE->START
    cycle, we should always reprogram the SOF device to start
    DMA from a known state so that hw_ptr/appl_ptrs remain valid.
    This is expected by ALSA core as it resets the buffer
    state as part of prepare (see snd_pcm_do_prepare()).

    Fix the issue by forcing reconfiguration of the FW with
    STREAM_PCM_PARAMS in prepare(). Use combined logic to handle
    prepare and the existing flow to reprogram hw-params after
    system suspend.

    Without the fix, first call to pcm pointer() will return
    an invalid hw_ptr and application may immediately observe XRUN
    status, unless "start_threshold" SW parameter is set to maximum
    value by the application.

    Signed-off-by: Kai Vehmanen <kai.vehmanen@linux.intel.com>
    Signed-off-by: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com>
    Link: https://lore.kernel.org/r/20190722141402.7194-3-pierre-
louis.bossart@linux.intel.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 89e2e06fc09fe6bc2ac2aa54f39fae758c6aedfd
Author: Nick Stoughton <nstoughton@logitech.com>
Date:   Wed Jul 17 14:56:06 2019 -0700
```

```
    leds: leds-lp5562 allow firmware files up to the maximum length

    [ Upstream commit ed2abfebb041473092b41527903f93390d38afa7 ]

    Firmware files are in ASCII, using 2 hex characters per byte. The
    maximum length of a firmware string is therefore

    16 (commands) * 2 (bytes per command) * 2 (characters per byte) = 64

    Fixes: ff45262a85db ("leds: add new LP5562 LED driver")
    Signed-off-by: Nick Stoughton <nstoughton@logitech.com>
    Acked-by: Pavel Machek <pavel@ucw.cz>
    Signed-off-by: Jacek Anaszewski <jacek.anaszewski@gmail.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0171410bb64b10e79e0a79ce4e9485ff0fbac99f
Author: Stefan Wahren <wahrenst@gmx.net>
Date:   Tue Jul 16 19:15:18 2019 +0200

    dmaengine: bcm2835: Print error in case setting DMA mask fails

    [ Upstream commit 72503b25ee363827aafffc3e8d872e6a92a7e422 ]

    During enabling of the RPi 4, we found out that the driver doesn't provide
    a helpful error message in case setting DMA mask fails. So add one.

    Signed-off-by: Stefan Wahren <wahrenst@gmx.net>
    Link: https://lore.kernel.org/r/1563297318-4900-1-git-send-email-wahrenst@gmx.net
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 11a55b9134a04c154d7853ec0f4687b9476d5415
Author: Stephen Boyd <swboyd@chromium.org>
Date:   Fri May 17 14:09:21 2019 -0700

    firmware: qcom_scm: Use proper types for dma mappings

    [ Upstream commit 6e37ccf78a53296c6c7bf426065762c27829eb84 ]

    We need to use the proper types and convert between physical addresses
    and dma addresses here to avoid mismatch warnings. This is especially
    important on systems with a different size for dma addresses and
    physical addresses. Otherwise, we get the following warning:

      drivers/firmware/qcom_scm.c: In function "qcom_scm_assign_mem":
      drivers/firmware/qcom_scm.c:469:47: error: passing argument 3 of "dma_alloc_coherent"
from incompatible pointer type [-Werror=incompatible-pointer-types]

    We also fix the size argument to dma_free_coherent() because that size
    doesn't need to be aligned after it's already aligned on the allocation
    size. In fact, dma debugging expects the same arguments to be passed to
    both the allocation and freeing sides of the functions so changing the
    size is incorrect regardless.

    Reported-by: Ian Jackson <ian.jackson@citrix.com>
    Cc: Ian Jackson <ian.jackson@citrix.com>
    Cc: Julien Grall <julien.grall@arm.com>
    Cc: Bjorn Andersson <bjorn.andersson@linaro.org>
    Cc: Avaneesh Kumar Dwivedi <akdwived@codeaurora.org>
    Tested-by: Bjorn Andersson <bjorn.andersson@linaro.org>
    Signed-off-by: Stephen Boyd <swboyd@chromium.org>
    Signed-off-by: Bjorn Andersson <bjorn.andersson@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e9c530538b96242262b89655960c0bad492ed787
```

```
Author: Oleksandr Suvorov <oleksandr.suvorov@toradex.com>
Date:   Fri Jul 19 10:05:37 2019 +0000

    ASoC: sgtl5000: Fix charge pump source assignment

    [ Upstream commit b6319b061ba279577fd7030a9848fbd6a17151e3 ]

    If VDDA != VDDIO and any of them is greater than 3.1V, charge pump
    source can be assigned automatically [1].

    [1] https://www.nxp.com/docs/en/data-sheet/SGTL5000.pdf

    Signed-off-by: Oleksandr Suvorov <oleksandr.suvorov@toradex.com>
    Reviewed-by: Marcel Ziswiler <marcel.ziswiler@toradex.com>
    Reviewed-by: Igor Opaniuk <igor.opaniuk@toradex.com>
    Reviewed-by: Fabio Estevam <festevam@gmail.com>
    Link: https://lore.kernel.org/r/20190719100524.23300-7-oleksandr.suvorov@toradex.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e9f621efaebddbefc0b4243a263644e740d639e8
Author: Oleksandr Suvorov <oleksandr.suvorov@toradex.com>
Date:   Fri Jul 19 10:05:35 2019 +0000

    ASoC: sgtl5000: Fix of unmute outputs on probe

    [ Upstream commit 631bc8f0134ae9620d86a96b8c5f9445d91a2dca ]

    To enable "zero cross detect" for ADC/HP, change
    HP_ZCD_EN/ADC_ZCD_EN bits only instead of writing the whole
    CHIP_ANA_CTRL register.

    Signed-off-by: Oleksandr Suvorov <oleksandr.suvorov@toradex.com>
    Reviewed-by: Marcel Ziswiler <marcel.ziswiler@toradex.com>
    Reviewed-by: Igor Opaniuk <igor.opaniuk@toradex.com>
    Reviewed-by: Fabio Estevam <festevam@gmail.com>
    Link: https://lore.kernel.org/r/20190719100524.23300-6-oleksandr.suvorov@toradex.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8bb449437c566f7c9f6ef0b2f81820b73f522517
Author: Lucas Stach <l.stach@pengutronix.de>
Date:   Fri Jul 19 16:36:37 2019 +0200

    ASoC: tlv320aic31xx: suppress error message for EPROBE_DEFER

    [ Upstream commit b7e814deae33eb30f8f8c6528e8e69b107978d88 ]

    Both the supplies and reset GPIO might need a probe deferral for the
    resource to be available. Don't print a error message in that case, as
    it is a normal operating condition.

    Signed-off-by: Lucas Stach <l.stach@pengutronix.de>
    Acked-by: Andrew F. Davis <afd@ti.com>
    Link: https://lore.kernel.org/r/20190719143637.2018-1-l.stach@pengutronix.de
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dc1db1791d36ebc1ba140f5dc15cc5c7fe63986b
Author: Andy Shevchenko <andriy.shevchenko@linux.intel.com>
Date:   Wed Jul 10 14:42:43 2019 +0300

    spi: dw-mmio: Clock should be shut when error occurs

    [ Upstream commit 3da9834d9381dd99273f2ad4e6d096c9187dc4f2 ]
```

```
    When optional clock requesting fails, the main clock is still up and running,
    we should shut it down in such caee.

    Fixes: 560ee7e91009 ("spi: dw: Add support for an optional interface clock")
    Cc: Phil Edworthy <phil.edworthy@renesas.com>
    Cc: Gareth Williams <gareth.williams.jx@renesas.com>
    Signed-off-by: Andy Shevchenko <andriy.shevchenko@linux.intel.com>
    Reviewed-by: Gareth Williams <gareth.williams.jx@renesas.com>
    Link: https://lore.kernel.org/r/20190710114243.30101-1-andriy.shevchenko@linux.intel.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c83f9794da451c090764536c64302224c317c20f
Author: Axel Lin <axel.lin@ingics.com>
Date:   Wed Jun 26 21:26:32 2019 +0800

    regulator: lm363x: Fix n_voltages setting for lm36274

    [ Upstream commit 962f170d9344e5d9edb3903971c591f42d55e226 ]

    According to the datasheet http://www.ti.com/lit/ds/symlink/lm36274.pdf:
    Table 23. VPOS Bias Register Field Descriptions VPOS[5:0]:
    VPOS voltage (50-mV steps): VPOS = 4 V + (Code × 50 mV), 6.5 V max
    000000 = 4 V
    000001 = 4.05 V
    :
    011110 = 5.5 V (Default)
    :
    110010 = 6.5 V
    110011 to 111111 map to 6.5 V

    So the LM36274_LDO_VSEL_MAX should be 0b110010 (0x32).
    The valid selectors are 0 ... LM36274_LDO_VSEL_MAX, n_voltages should be
    LM36274_LDO_VSEL_MAX + 1. Similarly, the n_voltages should be
    LM36274_BOOST_VSEL_MAX + 1 for LM36274_BOOST.

    Fixes: bff5e8071533 ("regulator: lm363x: Add support for LM36274")
    Signed-off-by: Axel Lin <axel.lin@ingics.com>
    Link: https://lore.kernel.org/r/20190626132632.32629-2-axel.lin@ingics.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a19262655b8727f29c2bf0cf1f47efe7520ff811
Author: Axel Lin <axel.lin@ingics.com>
Date:   Wed Jun 26 21:26:31 2019 +0800

    regulator: lm363x: Fix off-by-one n_voltages for lm3632 ldo_vpos/ldo_vneg

    [ Upstream commit 1e2cc8c5e0745b545d4974788dc606d678b6e564 ]

    According to the datasheet https://www.ti.com/lit/ds/symlink/lm3632a.pdf
    Table 20. VPOS Bias Register Field Descriptions VPOS[5:0]
    Sets the Positive Display Bias (LDO) Voltage (50 mV per step)
    000000: 4 V
    000001: 4.05 V
    000010: 4.1 V
    ..................
    011101: 5.45 V
    011110: 5.5 V (Default)
    011111: 5.55 V
    ..................
    100111: 5.95 V
    101000: 6 V
    Note: Codes 101001 to 111111 map to 6 V
```

```
    The LM3632_LDO_VSEL_MAX should be 0b101000 (0x28), so the maximum voltage
    can match the datasheet.

    Fixes: 3a8d1a73a037 ("regulator: add LM363X driver")
    Signed-off-by: Axel Lin <axel.lin@ingics.com>
    Link: https://lore.kernel.org/r/20190626132632.32629-1-axel.lin@ingics.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit acc7ede76e645be10075950afd188976b5914e12
Author: Takashi Iwai <tiwai@suse.de>
Date:   Wed Jul 17 14:30:23 2019 +0200

    ALSA: hda/hdmi - Don't report spurious jack state changes

    [ Upstream commit 551626ec0ad28dc43cae3094c35be7088cc625ab ]

    The HDMI jack handling reports the state change always via
    snd_jack_report() whenever hdmi_present_sense() is called, even if the
    state itself doesn't change from the previous time.  This is mostly
    harmless but still a bit confusing to user-space.

    This patch reduces such spurious jack state changes and reports only
    when the state really changed.  Also, as a minor optimization, avoid
    overwriting the pin ELD data when the state is identical.

    Reviewed-by: Kai Vehmanen <kai.vehmanen@linux.intel.com>
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7e8e29512eb09c4e8d31b5a266175c4497bd9ecf
Author: Hariprasad Kelam <hariprasad.kelam@gmail.com>
Date:   Sun Jul 21 23:38:15 2019 +0530

    cpufreq: ap806: Add NULL check after kcalloc

    [ Upstream commit 3355c91b79394593ebbb197c8e930a91826f4ff3 ]

    Add NULL check  after kcalloc.

    Fix below issue reported by coccicheck
    ./drivers/cpufreq/armada-8k-cpufreq.c:138:1-12: alloc with no test,
    possible model on line 151

    Signed-off-by: Hariprasad Kelam <hariprasad.kelam@gmail.com>
    Signed-off-by: Viresh Kumar <viresh.kumar@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f9c806835a4dee6a6d18836b2636f7f8f9868dcb
Author: Ranjani Sridharan <ranjani.sridharan@linux.intel.com>
Date:   Wed Jun 26 00:04:50 2019 -0700

    ASoC: SOF: Intel: hda: Make hdac_device device-managed

    [ Upstream commit ef9bec27485fefb6b93168fea73fda0dc9638046 ]

    snd_hdac_ext_bus_device_exit() has been recently modified
    to no longer free the hdac device. SOF allocates memory for
    hdac_device and hda_hda_priv with kzalloc. Make them
    device-managed instead so that they will be freed when the
    SOF driver is unloaded.

    Because of the above change, hda_codec is device-managed and
    it will be freed when the ASoC device is removed. Freeing
```

```
    the codec in snd_hda_codec_dev_release() leads to kernel
    panic while unloading and reloading the ASoC driver. So,
    avoid freeing the hda_codec for ASoC driver. This is done in
    the same patch to avoid bisect failure.

    Signed-off-by: Libin Yang <libin.yang@intel.com>
    Signed-off-by: Ranjani Sridharan <ranjani.sridharan@linux.intel.com>
    Reviewed-by: Takashi Iwai <tiwai@suse.de>
    Link: https://lore.kernel.org/r/20190626070450.7229-1-ranjani.sridharan@linux.intel.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 175b350048345a66bbfecc59da3c1339f6133422
Author: Chris Wilson <chris@chris-wilson.co.uk>
Date:   Sat Jul 20 12:33:37 2019 +0100

    ALSA: hda: Flush interrupts on disabling

    [ Upstream commit caa8422d01e983782548648e125fd617cadcec3f ]

    I was looking at

    <4> [241.835158] general protection fault: 0000 [#1] PREEMPT SMP PTI
    <4> [241.835181] CPU: 1 PID: 214 Comm: kworker/1:3 Tainted: G     U            5.2.0-CI-
CI_DRM_6509+ #1
    <4> [241.835199] Hardware name: Dell Inc.                OptiPlex 745
/0GW726, BIOS 2.3.1  05/21/2007
    <4> [241.835234] Workqueue: events snd_hdac_bus_process_unsol_events [snd_hda_core]
    <4> [241.835256] RIP: 0010:input_handle_event+0x16d/0x5e0
    <4> [241.835270] Code: 48 8b 93 58 01 00 00 8b 52 08 89 50 04 8b 83 f8 06 00 00 48 8b 93
00 07 00 00 8d 70 01 48 8d 04 c2 83 e1 08 89 b3 f8 06 00 00 <66> 89 28 66 44 89 60 02 44 89
68 04 8b 93 f8 06 00 00 0f 84 fd fe
    <4> [241.835304] RSP: 0018:ffffc9000019fda0 EFLAGS: 00010046
    <4> [241.835317] RAX: 6b6b6b6ec6c6c6c3 RBX: ffff8880290fefc8 RCX: 0000000000000000
    <4> [241.835332] RDX: 000000006b6b6b6b RSI: 000000006b6b6b6c RDI: 0000000000000046
    <4> [241.835347] RBP: 0000000000000005 R08: 0000000000000000 R09: 0000000000000001
    <4> [241.835362] R10: ffffc9000019faa0 R11: 0000000000000000 R12: 0000000000000004
    <4> [241.835377] R13: 0000000000000000 R14: ffff8880290ff1d0 R15: 0000000000000293
    <4> [241.835392] FS:  0000000000000000(0000) GS:ffff88803de80000(0000)
knlGS:0000000000000000
    <4> [241.835409] CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    <4> [241.835422] CR2: 00007ffe9a99e9b7 CR3: 000000002f588000 CR4: 00000000000006e0
    <4> [241.835436] Call Trace:
    <4> [241.835449]  input_event+0x45/0x70
    <4> [241.835464]  snd_jack_report+0xdc/0x100
    <4> [241.835490]  snd_hda_jack_report_sync+0x83/0xc0 [snd_hda_codec]
    <4> [241.835512]  snd_hdac_bus_process_unsol_events+0x5a/0x70 [snd_hda_core]
    <4> [241.835530]  process_one_work+0x245/0x610

    which has the hallmarks of a worker queued from interrupt after it was
    supposedly cancelled (note the POISON_FREE), and I could not see where
    the interrupt would be flushed on shutdown so added the likely suspects.

    Bugzilla: https://bugs.freedesktop.org/show_bug.cgi?id=111174
    Signed-off-by: Chris Wilson <chris@chris-wilson.co.uk>
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1d34ee30a424a97a9caede7880a845fedb60a690
Author: Ori Nimron <orinimron123@gmail.com>
Date:   Fri Sep 20 09:35:49 2019 +0200

    nfc: enforce CAP_NET_RAW for raw sockets

    [ Upstream commit 3a359798b176183ef09efb7a3dc59abad1cc7104 ]
```

```
    When creating a raw AF_NFC socket, CAP_NET_RAW needs to be checked
    first.

    Signed-off-by: Ori Nimron <orinimron123@gmail.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 41bdf4400adfe119140df2eaf44ca2ce809985d9
Author: Ori Nimron <orinimron123@gmail.com>
Date:   Fri Sep 20 09:35:48 2019 +0200

    ieee802154: enforce CAP_NET_RAW for raw sockets

    [ Upstream commit e69dbd4619e7674c1679cba49afd9dd9ac347eef ]

    When creating a raw AF_IEEE802154 socket, CAP_NET_RAW needs to be
    checked first.

    Signed-off-by: Ori Nimron <orinimron123@gmail.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Acked-by: Stefan Schmidt <stefan@datenfreihafen.org>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 0bdac683881231d261ba4651d0b248767cc02c27
Author: Ori Nimron <orinimron123@gmail.com>
Date:   Fri Sep 20 09:35:47 2019 +0200

    ax25: enforce CAP_NET_RAW for raw sockets

    [ Upstream commit 0614e2b73768b502fc32a75349823356d98aae2c ]

    When creating a raw AF_AX25 socket, CAP_NET_RAW needs to be checked
    first.

    Signed-off-by: Ori Nimron <orinimron123@gmail.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e7158355e964e5392f01431056f1f9b35355d1aa
Author: Ori Nimron <orinimron123@gmail.com>
Date:   Fri Sep 20 09:35:46 2019 +0200

    appletalk: enforce CAP_NET_RAW for raw sockets

    [ Upstream commit 6cc03e8aa36c51f3b26a0d21a3c4ce2809c842ac ]

    When creating a raw AF_APPLETALK socket, CAP_NET_RAW needs to be checked
    first.

    Signed-off-by: Ori Nimron <orinimron123@gmail.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 4a307942fd0cc0430e48dd79cc218d09935abe00
Author: Ori Nimron <orinimron123@gmail.com>
Date:   Fri Sep 20 09:35:45 2019 +0200

    mISDN: enforce CAP_NET_RAW for raw sockets

    [ Upstream commit b91ee4aa2a2199ba4d4650706c272985a5a32d80 ]
```

```
    When creating a raw AF_ISDN socket, CAP_NET_RAW needs to be checked
    first.

    Signed-off-by: Ori Nimron <orinimron123@gmail.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3a09b9bd110fba2336e352a4ada00fae840024b4
Author: Bodong Wang <bodong@mellanox.com>
Date:   Mon Aug 26 16:34:12 2019 -0500

    net/mlx5: Add device ID of upcoming BlueField-2

    [ Upstream commit d19a79ee38c8fda6d297e4227e80db8bf51c71a6 ]

    Add the device ID of upcoming BlueField-2 integrated ConnectX-6 Dx
    network controller. Its VFs will be using the generic VF device ID:
    0x101e "ConnectX Family mlx5Gen Virtual Function".

    Fixes: 2e9d3e83ab82 ("net/mlx5: Update the list of the PCI supported devices")
    Signed-off-by: Bodong Wang <bodong@mellanox.com>
    Signed-off-by: Saeed Mahameed <saeedm@mellanox.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f95ca2621d9be6811ad7fc4b36a1660a0d1a3480
Author: Eric Dumazet <edumazet@google.com>
Date:   Thu Sep 26 15:42:51 2019 -0700

    tcp: better handle TCP_USER_TIMEOUT in SYN_SENT state

    [ Upstream commit a66b10c05ee2d744189e9a2130394b070883d289 ]

    Yuchung Cheng and Marek Majkowski independently reported a weird
    behavior of TCP_USER_TIMEOUT option when used at connect() time.

    When the TCP_USER_TIMEOUT is reached, tcp_write_timeout()
    believes the flow should live, and the following condition
    in tcp_clamp_rto_to_user_timeout() programs one jiffie timers :

        remaining = icsk->icsk_user_timeout - elapsed;
        if (remaining <= 0)
            return 1; /* user timeout has passed; fire ASAP */

    This silly situation ends when the max syn rtx count is reached.

    This patch makes sure we honor both TCP_SYNCNT and TCP_USER_TIMEOUT,
    avoiding these spurious SYN packets.

    Fixes: b701a99e431d ("tcp: Add tcp_clamp_rto_to_user_timeout() helper to improve
accuracy")
    Signed-off-by: Eric Dumazet <edumazet@google.com>
    Reported-by: Yuchung Cheng <ycheng@google.com>
    Reported-by: Marek Majkowski <marek@cloudflare.com>
    Cc: Jon Maxwell <jmaxwell37@gmail.com>
    Link: https://marc.info/?l=linux-netdev&m=156940118307949&w=2
    Acked-by: Jon Maxwell <jmaxwell37@gmail.com>
    Tested-by: Marek Majkowski <marek@cloudflare.com>
    Signed-off-by: Marek Majkowski <marek@cloudflare.com>
    Acked-by: Yuchung Cheng <ycheng@google.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 7b6ffb6f6cdee2884d5033857d7556a7b9ae1ad2
```

```
Author: Eric Dumazet <edumazet@google.com>
Date:   Wed Sep 18 12:57:04 2019 -0700

    net: sched: fix possible crash in tcf_action_destroy()

    [ Upstream commit 3d66b89c30f9220a72e92847768fc8ba4d027d88 ]

    If the allocation done in tcf_exts_init() failed,
    we end up with a NULL pointer in exts->actions.

    kasan: GPF could be caused by NULL-ptr deref or user memory access
    general protection fault: 0000 [#1] PREEMPT SMP KASAN
    CPU: 1 PID: 8198 Comm: syz-executor.3 Not tainted 5.3.0-rc8+ #0
    Hardware name: Google Google Compute Engine/Google Compute Engine, BIOS Google
01/01/2011
    RIP: 0010:tcf_action_destroy+0x71/0x160 net/sched/act_api.c:705
    Code: c3 08 44 89 ee e8 4f cb bb fb 41 83 fd 20 0f 84 c9 00 00 00 e8 c0 c9 bb fb 48 89
d8 48 b9 00 00 00 00 00 fc ff df 48 c1 e8 03 <80> 3c 08 00 0f 85 c0 00 00 00 4c 8b 33 4d 85
f6 0f 84 9d 00 00 00
    RSP: 0018:ffff888096e16ff0 EFLAGS: 00010246
    RAX: 0000000000000000 RBX: 0000000000000000 RCX: dffffc0000000000
    RDX: 0000000000040000 RSI: ffffffff85b6ab30 RDI: 0000000000000000
    RBP: ffff888096e17020 R08: ffff8880993f6140 R09: fffffbfff11cae67
    R10: fffffbfff11cae66 R11: ffffffff88e57333 R12: 0000000000000000
    R13: 0000000000000000 R14: ffff888096e177a0 R15: 0000000000000001
    FS:  00007f62bc84a700(0000) GS:ffff8880ae900000(0000) knlGS:0000000000000000
    CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    CR2: 0000000000758040 CR3: 0000000088b64000 CR4: 00000000001426e0
    DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000
    DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400
    Call Trace:
     tcf_exts_destroy+0x38/0xb0 net/sched/cls_api.c:3030
     tcindex_set_parms+0xf7f/0x1e50 net/sched/cls_tcindex.c:488
     tcindex_change+0x230/0x318 net/sched/cls_tcindex.c:519
     tc_new_tfilter+0xa4b/0x1c70 net/sched/cls_api.c:2152
     rtnetlink_rcv_msg+0x838/0xb00 net/core/rtnetlink.c:5214
     netlink_rcv_skb+0x177/0x450 net/netlink/af_netlink.c:2477
     rtnetlink_rcv+0x1d/0x30 net/core/rtnetlink.c:5241
     netlink_unicast_kernel net/netlink/af_netlink.c:1302 [inline]
     netlink_unicast+0x531/0x710 net/netlink/af_netlink.c:1328
     netlink_sendmsg+0x8a5/0xd60 net/netlink/af_netlink.c:1917
     sock_sendmsg_nosec net/socket.c:637 [inline]
     sock_sendmsg+0xd7/0x130 net/socket.c:657
     ___sys_sendmsg+0x3e2/0x920 net/socket.c:2311
     __sys_sendmmsg+0x1bf/0x4d0 net/socket.c:2413
     __do_sys_sendmmsg net/socket.c:2442 [inline]

    Fixes: 90b73b77d08e ("net: sched: change action API to use array of pointers to
actions")
    Signed-off-by: Eric Dumazet <edumazet@google.com>
    Reported-by: syzbot <syzkaller@googlegroups.com>
    Cc: Vlad Buslov <vladbu@mellanox.com>
    Cc: Jiri Pirko <jiri@mellanox.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b1cc82a236075ae73bd483e3b77349a98614e7fb
Author: Saeed Mahameed <saeedm@mellanox.com>
Date:   Wed Sep 11 07:50:13 2019 -0700

    net/mlx5e: Fix traffic duplication in ethtool steering

    [ Upstream commit d22fcc806b84b9818de08b32e494f3c05dd236c7 ]

    Before this patch, when adding multiple ethtool steering rules with
```

```
    identical classification, the driver used to append the new destination
    to the already existing hw rule, which caused the hw to forward the
    traffic to all destinations (rx queues).

    Here we avoid this by setting the "no append" mlx5 fs core flag when
    adding a new ethtool rule.

    Fixes: 6dc6071cfcde ("net/mlx5e: Add ethtool flow steering support")
    Signed-off-by: Saeed Mahameed <saeedm@mellanox.com>
    Reviewed-by: Maor Gottlieb <maorg@mellanox.com>
    Signed-off-by: Saeed Mahameed <saeedm@mellanox.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit fbb79dae48a93a14e764807635f4cdce50523d58
Author: David Ahern <dsahern@gmail.com>
Date:   Wed Sep 25 07:53:19 2019 -0700

    vrf: Do not attempt to create IPv6 mcast rule if IPv6 is disabled

    [ Upstream commit dac91170f8e9c73784af5fad6225e954b795601c ]

    A user reported that vrf create fails when IPv6 is disabled at boot using
    'ipv6.disable=1':
       https://bugzilla.kernel.org/show_bug.cgi?id=204903

    The failure is adding fib rules at create time. Add RTNL_FAMILY_IP6MR to
    the check in vrf_fib_rule if ipv6_mod_enabled is disabled.

    Fixes: e4a38c0c4b27 ("ipv6: add vrf table handling code for ipv6 mcast")
    Signed-off-by: David Ahern <dsahern@gmail.com>
    Cc: Patrick Ruddy <pruddy@vyatta.att-mail.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 8e6478aa0e04d7b4c7c24526a7b4560dde2e4412
Author: Cong Wang <xiyou.wangcong@gmail.com>
Date:   Wed Sep 18 18:44:43 2019 -0700

    net_sched: add policy validation for action attributes

    [ Upstream commit 199ce850ce112315cfc68d42b694bcaa27b097b7 ]

    Similar to commit 8b4c3cdd9dd8
    ("net: sched: Add policy validation for tc attributes"), we need
    to add proper policy validation for TC action attributes too.

    Cc: David Ahern <dsahern@gmail.com>
    Cc: Jamal Hadi Salim <jhs@mojatatu.com>
    Signed-off-by: Cong Wang <xiyou.wangcong@gmail.com>
    Acked-by: Jiri Pirko <jiri@mellanox.com>
    Signed-off-by: Jakub Kicinski <jakub.kicinski@netronome.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 79923eda8935477de4021ed13c646003b5c29785
Author: David Ahern <dsahern@gmail.com>
Date:   Tue Sep 17 10:39:49 2019 -0700

    ipv4: Revert removal of rt_uses_gateway

    [ Upstream commit 77d5bc7e6a6cf8bbeca31aab7f0c5449a5eee762 ]

    Julian noted that rt_uses_gateway has a more subtle use than 'is gateway
    set':
        https://lore.kernel.org/netdev/alpine.LFD.2.21.1909151104060.2546@ja.home.ssi.bg/
```

```
    Revert that part of the commit referenced in the Fixes tag.

    Currently, there are no u8 holes in 'struct rtable'. There is a 4-byte hole
    in the second cacheline which contains the gateway declaration. So move
    rt_gw_family down to the gateway declarations since they are always used
    together, and then re-use that u8 for rt_uses_gateway. End result is that
    rtable size is unchanged.

    Fixes: 1550c171935d ("ipv4: Prepare rtable for IPv6 gateway")
    Reported-by: Julian Anastasov <ja@ssi.bg>
    Signed-off-by: David Ahern <dsahern@gmail.com>
    Reviewed-by: Julian Anastasov <ja@ssi.bg>
    Signed-off-by: Jakub Kicinski <jakub.kicinski@netronome.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit bd2ee1d7c73af35c0293f6908bcb8451362575bd
Author: Vinicius Costa Gomes <vinicius.gomes@intel.com>
Date:   Mon Sep 23 22:04:58 2019 -0700

    net/sched: cbs: Fix not adding cbs instance to list

    [ Upstream commit 3e8b9bfa110896f95d602d8c98d5f9d67e41d78c ]

    When removing a cbs instance when offloading is enabled, the crash
    below can be observed.

    The problem happens because that when offloading is enabled, the cbs
    instance is not added to the list.

    Also, the current code doesn't handle correctly the case when offload
    is disabled without removing the qdisc: if the link speed changes the
    credit calculations will be wrong. When we create the cbs instance
    with offloading enabled, it's not added to the notification list, when
    later we disable offloading, it's not in the list, so link speed
    changes will not affect it.

    The solution for both issues is the same, add the cbs instance being
    created unconditionally to the global list, even if the link state
    notification isn't useful "right now".

    Crash log:

    [518758.189866] BUG: kernel NULL pointer dereference, address: 0000000000000000
    [518758.189870] #PF: supervisor read access in kernel mode
    [518758.189871] #PF: error_code(0x0000) - not-present page
    [518758.189872] PGD 0 P4D 0
    [518758.189874] Oops: 0000 [#1] SMP PTI
    [518758.189876] CPU: 3 PID: 4825 Comm: tc Not tainted 5.2.9 #1
    [518758.189877] Hardware name: Gigabyte Technology Co., Ltd. Z390 AORUS ULTRA/Z390 AORUS
ULTRA-CF, BIOS F7 03/14/2019
    [518758.189881] RIP: 0010:__list_del_entry_valid+0x29/0xa0
    [518758.189883] Code: 90 48 b8 00 01 00 00 00 00 ad de 55 48 8b 17 4c 8b 47 08 48 89 e5
48 39 c2 74 27 48 b8 00 02 00 00 00 00 ad de 49 39 c0 74 2d <49> 8b 30 48 39 fe 75 3d 48 8b
52 08 48 39 f2 75 4c b8 01 00 00 00
    [518758.189885] RSP: 0018:ffffa27e43903990 EFLAGS: 00010207
    [518758.189887] RAX: dead000000000200 RBX: ffff8bce69f0f000 RCX: 0000000000000000
    [518758.189888] RDX: 0000000000000000 RSI: ffff8bce69f0f064 RDI: ffff8bce69f0f1e0
    [518758.189890] RBP: ffffa27e43903990 R08: 0000000000000000 R09: ffff8bce69e788c0
    [518758.189891] R10: ffff8bce62acd400 R11: 00000000000003cb R12: ffff8bce69e78000
    [518758.189892] R13: ffff8bce69f0f140 R14: 0000000000000000 R15: 0000000000000000
    [518758.189894] FS:  00007fa1572c8f80(0000) GS:ffff8bce6e0c0000(0000)
knlGS:0000000000000000
    [518758.189895] CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    [518758.189896] CR2: 0000000000000000 CR3: 000000040a398006 CR4: 00000000003606e0
    [518758.189898] DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000
```

```
    [518758.189899] DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400
    [518758.189900] Call Trace:
    [518758.189904]  cbs_destroy+0x32/0xa0 [sch_cbs]
    [518758.189906]  qdisc_destroy+0x45/0x120
    [518758.189907]  qdisc_put+0x25/0x30
    [518758.189908]  qdisc_graft+0x2c1/0x450
    [518758.189910]  tc_get_qdisc+0x1c8/0x310
    [518758.189912]  ? get_page_from_freelist+0x91a/0xcb0
    [518758.189914]  rtnetlink_rcv_msg+0x293/0x360
    [518758.189916]  ? kmem_cache_alloc_node_trace+0x178/0x260
    [518758.189918]  ? __kmalloc_node_track_caller+0x38/0x50
    [518758.189920]  ? rtnl_calcit.isra.0+0xf0/0xf0
    [518758.189922]  netlink_rcv_skb+0x48/0x110
    [518758.189923]  rtnetlink_rcv+0x10/0x20
    [518758.189925]  netlink_unicast+0x15b/0x1d0
    [518758.189926]  netlink_sendmsg+0x1ea/0x380
    [518758.189929]  sock_sendmsg+0x2f/0x40
    [518758.189930]  ___sys_sendmsg+0x295/0x2f0
    [518758.189932]  ? ___sys_recvmsg+0x151/0x1e0
    [518758.189933]  ? do_wp_page+0x7e/0x450
    [518758.189935]  __sys_sendmsg+0x48/0x80
    [518758.189937]  __x64_sys_sendmsg+0x1a/0x20
    [518758.189939]  do_syscall_64+0x53/0x1f0
    [518758.189941]  entry_SYSCALL_64_after_hwframe+0x44/0xa9
    [518758.189942] RIP: 0033:0x7fa15755169a
    [518758.189944] Code: 48 c7 c0 ff ff ff ff eb be 0f 1f 80 00 00 00 00 f3 0f 1e fa 64 8b
 04 25 18 00 00 00 85 c0 75 18 b8 2e 00 00 00 c5 fc 77 0f 05 <48> 3d 00 f0 ff ff 77 5e c3 0f
 1f 44 00 00 48 83 ec 28 89 54 24 1c
    [518758.189946] RSP: 002b:00007ffda58b60b8 EFLAGS: 00000246 ORIG_RAX: 000000000000002e
    [518758.189948] RAX: ffffffffffffffda RBX: 000055e4b836d9a0 RCX: 00007fa15755169a
    [518758.189949] RDX: 0000000000000000 RSI: 00007ffda58b6128 RDI: 0000000000000003
    [518758.189951] RBP: 00007ffda58b6190 R08: 0000000000000001 R09: 000055e4b9d848a0
    [518758.189952] R10: 0000000000000000 R11: 0000000000000246 R12: 000000005d654b49
    [518758.189953] R13: 0000000000000000 R14: 00007ffda58b6230 R15: 00007ffda58b6210
    [518758.189955] Modules linked in: sch_cbs sch_etf sch_mqprio netlink_diag unix_diag
 e1000e igb intel_pch_thermal thermal video backlight pcc_cpufreq
    [518758.189960] CR2: 0000000000000000
    [518758.189961] ---[ end trace 6a13f7aaf5376019 ]---
    [518758.189963] RIP: 0010:__list_del_entry_valid+0x29/0xa0
    [518758.189964] Code: 90 48 b8 00 01 00 00 00 00 ad de 55 48 8b 17 4c 8b 47 08 48 89 e5
 48 39 c2 74 27 48 b8 00 02 00 00 00 00 ad de 49 39 c0 74 2d <49> 8b 30 48 39 fe 75 3d 48 8b
 52 08 48 39 f2 75 4c b8 01 00 00 00
    [518758.189967] RSP: 0018:ffffa27e43903990 EFLAGS: 00010207
    [518758.189968] RAX: dead000000000200 RBX: ffff8bce69f0f000 RCX: 0000000000000000
    [518758.189969] RDX: 0000000000000000 RSI: ffff8bce69f0f064 RDI: ffff8bce69f0f1e0
    [518758.189971] RBP: ffffa27e43903990 R08: 0000000000000000 R09: ffff8bce69e788c0
    [518758.189972] R10: ffff8bce62acd400 R11: 00000000000003cb R12: ffff8bce69e78000
    [518758.189973] R13: ffff8bce69f0f140 R14: 0000000000000000 R15: 0000000000000000
    [518758.189975] FS:  00007fa1572c8f80(0000) GS:ffff8bce6e0c0000(0000)
 knlGS:0000000000000000
    [518758.189976] CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    [518758.189977] CR2: 0000000000000000 CR3: 000000040a398006 CR4: 00000000003606e0
    [518758.189979] DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000
    [518758.189980] DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400

    Fixes: e0a7683d30e9 ("net/sched: cbs: fix port_rate miscalculation")
    Signed-off-by: Vinicius Costa Gomes <vinicius.gomes@intel.com>
    Acked-by: Cong Wang <xiyou.wangcong@gmail.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

 commit 934d16fbbee18e21f4ed06e559c7f9e89e1cb0ba
 Author: Hans Andersson <hans.andersson@cellavision.se>
 Date:   Thu Sep 26 09:54:37 2019 +0200
```

```
    net: phy: micrel: add Asym Pause workaround for KSZ9021

    [ Upstream commit 407d8098cb1ab338199f4753162799a488d87d23 ]

    The Micrel KSZ9031 PHY may fail to establish a link when the Asymmetric
    Pause capability is set. This issue is described in a Silicon Errata
    (DS80000691D or DS80000692D), which advises to always disable the
    capability.

    Micrel KSZ9021 has no errata, but has the same issue with Asymmetric Pause.
    This patch apply the same workaround as the one for KSZ9031.

    Fixes: 3aed3e2a143c ("net: phy: micrel: add Asym Pause workaround")
    Signed-off-by: Hans Andersson <hans.andersson@cellavision.se>
    Reviewed-by: Andrew Lunn <andrew@lunn.ch>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b0d3e891ac056c89a37eb29e822f4e02731ae33b
Author: David Ahern <dsahern@gmail.com>
Date:   Tue Sep 17 10:30:35 2019 -0700

    selftests: Update fib_nexthop_multiprefix to handle missing ping6

    [ Upstream commit e84622ce24482f6e9c1bf29d3bdd556eb587ff41 ]

    Some distributions (e.g., debian buster) do not install ping6. Re-use
    the hook in pmtu.sh to detect this and fallback to ping.

    Fixes: 735ab2f65dce ("selftests: Add test with multiple prefixes using single nexthop")
    Signed-off-by: David Ahern <dsahern@gmail.com>
    Signed-off-by: Jakub Kicinski <jakub.kicinski@netronome.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 82ca29a9e2ff65ae9a3d084050752e395d36cc3a
Author: Eric Dumazet <edumazet@google.com>
Date:   Thu Sep 19 10:12:36 2019 -0700

    ipv6: fix a typo in fib6_rule_lookup()

    [ Upstream commit 7b09c2d052db4b4ad0b27b97918b46a7746966fa ]

    Yi Ren reported an issue discovered by syzkaller, and bisected
    to the cited commit.

    Many thanks to Yi, this trivial patch does not reflect the patient
    work that has been done.

    Fixes: d64a1f574a29 ("ipv6: honor RT6_LOOKUP_F_DST_NOREF in rule lookup logic")
    Signed-off-by: Eric Dumazet <edumazet@google.com>
    Acked-by: Wei Wang <weiwan@google.com>
    Bisected-and-reported-by: Yi Ren <c4tren@gmail.com>
    Signed-off-by: Jakub Kicinski <jakub.kicinski@netronome.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 5bbc6b59204c8db7fee12506193ffcb3d05dcbec
Author: Dmytro Linkin <dmitrolin@mellanox.com>
Date:   Fri Sep 13 10:42:21 2019 +0000

    net/mlx5e: Fix matching on tunnel addresses type

    [ Upstream commit fe1587a7de94912ed75ba5ddbfabf0741f9f8239 ]

    In mlx5 parse_tunnel_attr() function dispatch on encap IP address type
    is performed by directly checking flow_rule_match_key() on
```

```
    FLOW_DISSECTOR_KEY_ENC_IPV4_ADDRS, and then on
    FLOW_DISSECTOR_KEY_ENC_IPV6_ADDRS. However, since those are stored in
    union, first check is always true if any type of encap address is set,
    which leads to IPv6 tunnel encap address being parsed as IPv4 by mlx5.
    Determine correct IP address type by checking control key first and if
    it set, take address type from match.key->addr_type.

    Fixes: d1bda7eecd88 ("net/mlx5e: Allow matching only enc_key_id/enc_dst_port for
decapsulation action")
    Signed-off-by: Dmytro Linkin <dmitrolin@mellanox.com>
    Reviewed-by: Vlad Buslov <vladbu@mellanox.com>
    Reviewed-by: Eli Britstein <elibr@mellanox.com>
    Reviewed-by: Roi Dayan <roid@mellanox.com>
    Signed-off-by: Saeed Mahameed <saeedm@mellanox.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a9f6c8350c5db7b26302cbef34ec77894fab1a48
Author: Ka-Cheong Poon <ka-cheong.poon@oracle.com>
Date:   Tue Sep 24 08:51:16 2019 -0700

    net/rds: Check laddr_check before calling it

    [ Upstream commit 05733434ee9ae6548723a808647248583e347cca ]

    In rds_bind(), laddr_check is called without checking if it is NULL or
    not.  And rs_transport should be reset if rds_add_bound() fails.

    Fixes: c5c1a030a7db ("net/rds: An rds_sock is added too early to the hash table")
    Reported-by: syzbot+fae39afd2101a17ec624@syzkaller.appspotmail.com
    Signed-off-by: Ka-Cheong Poon <ka-cheong.poon@oracle.com>
    Acked-by: Santosh Shilimkar <santosh.shilimkar@oracle.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a4e16dca3940030a0ca83c1b79b0d0aa4db21c61
Author: Oliver Neukum <oneukum@suse.com>
Date:   Thu Sep 19 10:23:08 2019 +0200

    usbnet: sanity checking of packet sizes and device mtu

    [ Upstream commit 280ceaed79f18db930c0cc8bb21f6493490bf29c ]

    After a reset packet sizes and device mtu can change and need
    to be reevaluated to calculate queue sizes.
    Malicious devices can set this to zero and we divide by it.
    Introduce sanity checking.

    Reported-and-tested-by:  syzbot+6102c120be558c885f04@syzkaller.appspotmail.com
    Signed-off-by: Oliver Neukum <oneukum@suse.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c33ffca13cb5f2b55f46204d9e404d3fa5a48b00
Author: Bjørn Mork <bjorn@mork.no>
Date:   Wed Sep 18 14:17:38 2019 +0200

    usbnet: ignore endpoints with invalid wMaxPacketSize

    [ Upstream commit 8d3d7c2029c1b360f1a6b0a2fca470b57eb575c0 ]

    Endpoints with zero wMaxPacketSize are not usable for transferring
    data. Ignore such endpoints when looking for valid in, out and
    status pipes, to make the drivers more robust against invalid and
    meaningless descriptors.
```

```
    The wMaxPacketSize of these endpoints are used for memory allocations
    and as divisors in many usbnet minidrivers. Avoiding zero is therefore
    critical.

    Signed-off-by: Bjørn Mork <bjorn@mork.no>
    Signed-off-by: Jakub Kicinski <jakub.kicinski@netronome.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 97ce956e015809018212ff0c6aeb9183eb4257db
Author: Kevin(Yudong) Yang <yyd@google.com>
Date:   Thu Sep 26 10:30:05 2019 -0400

    tcp_bbr: fix quantization code to not raise cwnd if not probing bandwidth

    [ Upstream commit 6b3656a60f2067738d1a423328199720806f0c44 ]

    There was a bug in the previous logic that attempted to ensure gain cycling
    gets inflight above BDP even for small BDPs. This code correctly raised and
    lowered target inflight values during the gain cycle. And this code
    correctly ensured that cwnd was raised when probing bandwidth. However, it
    did not correspondingly ensure that cwnd was *not* raised in this way when
    *not* probing for bandwidth. The result was that small-BDP flows that were
    always cwnd-bound could go for many cycles with a fixed cwnd, and not probe
    or yield bandwidth at all. This meant that multiple small-BDP flows could
    fail to converge in their bandwidth allocations.

    Fixes: 3c346b233c68 ("tcp_bbr: fix bw probing to raise in-flight data for very small
BDPs")
    Signed-off-by: Kevin(Yudong) Yang <yyd@google.com>
    Acked-by: Neal Cardwell <ncardwell@google.com>
    Acked-by: Yuchung Cheng <ycheng@google.com>
    Acked-by: Soheil Hassas Yeganeh <soheil@google.com>
    Acked-by: Priyaranjan Jha <priyarjha@google.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c8c2ed0f5c74f8ffac0bf1a07a025347c1fc2898
Author: Stephen Hemminger <stephen@networkplumber.org>
Date:   Fri Sep 20 18:18:26 2019 +0200

    skge: fix checksum byte order

    [ Upstream commit 5aafeb74b5bb65b34cc87c7623f9fa163a34fa3b ]

    Running old skge driver on PowerPC causes checksum errors
    because hardware reported 1's complement checksum is in little-endian
    byte order.

    Reported-by: Benoit <benoit.sansoni@gmail.com>
    Signed-off-by: Stephen Hemminger <stephen@networkplumber.org>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 063f700ec31104b0c4505b5d069ab0c3ffb6cd30
Author: David Ahern <dsahern@gmail.com>
Date:   Tue Sep 17 10:30:21 2019 -0700

    selftests: Update fib_tests to handle missing ping6

    [ Upstream commit 0360894a05ed52be268e3c4d40b2df9d94975fa6 ]

    Some distributions (e.g., debian buster) do not install ping6. Re-use
    the hook in pmtu.sh to detect this and fallback to ping.

    Fixes: a0e11da78f48 ("fib_tests: Add tests for metrics on routes")
```

```
    Signed-off-by: David Ahern <dsahern@gmail.com>
    Signed-off-by: Jakub Kicinski <jakub.kicinski@netronome.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 2a7d2e02042c999ea8b542675878e6a21789d21f
Author: Eric Dumazet <edumazet@google.com>
Date:   Wed Sep 18 08:05:39 2019 -0700

    sch_netem: fix a divide by zero in tabledist()

    [ Upstream commit b41d936b5ecfdb3a4abc525ce6402a6c49cffddc ]

    syzbot managed to crash the kernel in tabledist() loading
    an empty distribution table.

            t = dist->table[rnd % dist->size];

    Simply return an error when such load is attempted.

    Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
    Signed-off-by: Eric Dumazet <edumazet@google.com>
    Reported-by: syzbot <syzkaller@googlegroups.com>
    Signed-off-by: Jakub Kicinski <jakub.kicinski@netronome.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 7f643b14fd154af1becaad45eed0c937b390123a
Author: Takeshi Misawa <jeliantsurux@gmail.com>
Date:   Sun Sep 22 16:45:31 2019 +0900

    ppp: Fix memory leak in ppp_write

    [ Upstream commit 4c247de564f1ff614d11b3bb5313fb70d7b9598b ]

    When ppp is closing, __ppp_xmit_process() failed to enqueue skb
    and skb allocated in ppp_write() is leaked.

    syzbot reported :
    BUG: memory leak
    unreferenced object 0xffff88812a17bc00 (size 224):
      comm "syz-executor673", pid 6952, jiffies 4294942888 (age 13.040s)
      hex dump (first 32 bytes):
        00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
        00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
      backtrace:
        [<00000000d110fff9>] kmemleak_alloc_recursive include/linux/kmemleak.h:43 [inline]
        [<00000000d110fff9>] slab_post_alloc_hook mm/slab.h:522 [inline]
        [<00000000d110fff9>] slab_alloc_node mm/slab.c:3262 [inline]
        [<00000000d110fff9>] kmem_cache_alloc_node+0x163/0x2f0 mm/slab.c:3574
        [<000000002d616113>] __alloc_skb+0x6e/0x210 net/core/skbuff.c:197
        [<000000000167fc45>] alloc_skb include/linux/skbuff.h:1055 [inline]
        [<000000000167fc45>] ppp_write+0x48/0x120 drivers/net/ppp/ppp_generic.c:502
        [<000000009ab42c0b>] __vfs_write+0x43/0xa0 fs/read_write.c:494
        [<00000000086b2e22>] vfs_write fs/read_write.c:558 [inline]
        [<00000000086b2e22>] vfs_write+0xee/0x210 fs/read_write.c:542
        [<00000000a2b70ef9>] ksys_write+0x7c/0x130 fs/read_write.c:611
        [<00000000ce5e0fdd>] __do_sys_write fs/read_write.c:623 [inline]
        [<00000000ce5e0fdd>] __se_sys_write fs/read_write.c:620 [inline]
        [<00000000ce5e0fdd>] __x64_sys_write+0x1e/0x30 fs/read_write.c:620
        [<00000000d9d7b370>] do_syscall_64+0x76/0x1a0 arch/x86/entry/common.c:296
        [<0000000006e6d506>] entry_SYSCALL_64_after_hwframe+0x44/0xa9

    Fix this by freeing skb, if ppp is closing.

    Fixes: 6d066734e9f0 ("ppp: avoid loop in xmit recursion detection code")
    Reported-and-tested-by: syzbot+d9c8bf24e56416d7ce2c@syzkaller.appspotmail.com
```

```
    Signed-off-by: Takeshi Misawa <jeliantsurux@gmail.com>
    Reviewed-by: Guillaume Nault <gnault@redhat.com>
    Tested-by: Guillaume Nault <gnault@redhat.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit bfacdaac206d3541e389c75b1b7218a748d21194
Author: Li RongQing <lirongqing@baidu.com>
Date:   Tue Sep 24 19:11:52 2019 +0800

    openvswitch: change type of UPCALL_PID attribute to NLA_UNSPEC

    [ Upstream commit ea8564c865299815095bebeb4b25bef474218e4c ]

    userspace openvswitch patch "(dpif-linux: Implement the API
    functions to allow multiple handler threads read upcall)"
    changes its type from U32 to UNSPEC, but leave the kernel
    unchanged

    and after kernel 6e237d099fac "(netlink: Relax attr validation
    for fixed length types)", this bug is exposed by the below
    warning

            [   57.215841] netlink: 'ovs-vswitchd': attribute type 5 has an invalid length.

    Fixes: 5cd667b0a456 ("openvswitch: Allow each vport to have an array of 'port_id's")
    Signed-off-by: Li RongQing <lirongqing@baidu.com>
    Acked-by: Pravin B Shelar <pshelar@ovn.org>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 96996f7118c7896a1bc7191ec2c253f038ab611c
Author: Navid Emamdoost <navid.emamdoost@gmail.com>
Date:   Wed Sep 25 13:24:02 2019 -0500

    nfp: flower: prevent memory leak in nfp_flower_spawn_phy_reprs

    [ Upstream commit 8572cea1461a006bce1d06c0c4b0575869125fa4 ]

    In nfp_flower_spawn_phy_reprs, in the for loop over eth_tbl if any of
    intermediate allocations or initializations fail memory is leaked.
    requiered releases are added.

    Fixes: b94524529741 ("nfp: flower: add per repr private data for LAG offload")
    Signed-off-by: Navid Emamdoost <navid.emamdoost@gmail.com>
    Acked-by: Jakub Kicinski <jakub.kicinski@netronome.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 5c95791bd704bd04a7c89629880afbb49b4037b8
Author: Navid Emamdoost <navid.emamdoost@gmail.com>
Date:   Wed Sep 25 14:05:09 2019 -0500

    nfp: flower: fix memory leak in nfp_flower_spawn_vnic_reprs

    [ Upstream commit 8ce39eb5a67aee25d9f05b40b673c95b23502e3e ]

    In nfp_flower_spawn_vnic_reprs in the loop if initialization or the
    allocations fail memory is leaked. Appropriate releases are added.

    Fixes: b94524529741 ("nfp: flower: add per repr private data for LAG offload")
    Signed-off-by: Navid Emamdoost <navid.emamdoost@gmail.com>
    Acked-by: Jakub Kicinski <jakub.kicinski@netronome.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit b232ef2ae46ed92381273c88b6253a7193a93649
Author: Thierry Reding <treding@nvidia.com>
Date:   Mon Sep 23 11:59:15 2019 +0200

    net: stmmac: Fix page pool size

    [ Upstream commit 4f28bd956e081fc018fe9b41ffa31573f17bfb61 ]

    The size of individual pages in the page pool in given by an order. The
    order is the binary logarithm of the number of pages that make up one of
    the pages in the pool. However, the driver currently passes the number
    of pages rather than the order, so it ends up wasting quite a bit of
    memory.

    Fix this by taking the binary logarithm and passing that in the order
    field.

    Fixes: 2af6106ae949 ("net: stmmac: Introducing support for Page Pool")
    Signed-off-by: Thierry Reding <treding@nvidia.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 5d54dde9ecbf1fe44507aff9ae2296f37ba42ff0
Author: Cong Wang <xiyou.wangcong@gmail.com>
Date:   Wed Sep 18 16:24:12 2019 -0700

    net_sched: add max len check for TCA_KIND

    [ Upstream commit 62794fc4fbf52f2209dc094ea255eaef760e7d01 ]

    The TCA_KIND attribute is of NLA_STRING which does not check
    the NUL char. KMSAN reported an uninit-value of TCA_KIND which
    is likely caused by the lack of NUL.

    Change it to NLA_NUL_STRING and add a max len too.

    Fixes: 8b4c3cdd9dd8 ("net: sched: Add policy validation for tc attributes")
    Reported-and-tested-by: syzbot+618aacd49e8c8b8486bd@syzkaller.appspotmail.com
    Cc: Jamal Hadi Salim <jhs@mojatatu.com>
    Signed-off-by: Cong Wang <xiyou.wangcong@gmail.com>
    Reviewed-by: David Ahern <dsahern@gmail.com>
    Acked-by: Jiri Pirko <jiri@mellanox.com>
    Signed-off-by: Jakub Kicinski <jakub.kicinski@netronome.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 258718082e2a40f69a7d1c5259db7ffbe25027e3
Author: Davide Caratti <dcaratti@redhat.com>
Date:   Tue Sep 17 11:30:55 2019 +0200

    net/sched: act_sample: don't push mac header on ip6gre ingress

    [ Upstream commit 92974a1d006ad8b30d53047c70974c9e065eb7df ]

    current 'sample' action doesn't push the mac header of ingress packets if
    they are received by a layer 3 tunnel (like gre or sit); but it forgot to
    check for gre over ipv6, so the following script:

     # tc q a dev $d clsact
     # tc f a dev $d ingress protocol ip flower ip_proto icmp action sample \
     > group 100 rate 1
     # psample -v -g 100

    dumps everything, including outer header and mac, when $d is a gre tunnel
    over ipv6. Fix this adding a missing label for ARPHRD_IP6GRE devices.
```

```
    Fixes: 5c5670fae430 ("net/sched: Introduce sample tc action")
    Signed-off-by: Davide Caratti <dcaratti@redhat.com>
    Reviewed-by: Yotam Gigi <yotam.gi@gmail.com>
    Signed-off-by: Jakub Kicinski <jakub.kicinski@netronome.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 4f8ec2656bbddb40e3dc8aa51d4c4ae451e27ba3
Author: Bjorn Andersson <bjorn.andersson@linaro.org>
Date:   Wed Sep 18 10:21:17 2019 -0700

    net: qrtr: Stop rx_worker before freeing node

    [ Upstream commit 73f0c11d11329a0d6d205d4312b6e5d2512af7c5 ]

    As the endpoint is unregistered there might still be work pending to
    handle incoming messages, which will result in a use after free
    scenario. The plan is to remove the rx_worker, but until then (and for
    stable@) ensure that the work is stopped before the node is freed.

    Fixes: bdabad3e363d ("net: Add Qualcomm IPC router")
    Cc: stable@vger.kernel.org
    Signed-off-by: Bjorn Andersson <bjorn.andersson@linaro.org>
    Signed-off-by: Jakub Kicinski <jakub.kicinski@netronome.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 718218a2fef482011ef665c2422481912fd3b40a
Author: Peter Mamonov <pmamonov@gmail.com>
Date:   Wed Sep 18 19:27:55 2019 +0300

    net/phy: fix DP83865 10 Mbps HDX loopback disable function

    [ Upstream commit e47488b2df7f9cb405789c7f5d4c27909fc597ae ]

    According to the DP83865 datasheet "the 10 Mbps HDX loopback can be
    disabled in the expanded memory register 0x1C0.1". The driver erroneously
    used bit 0 instead of bit 1.

    Fixes: 4621bf129856 ("phy: Add file missed in previous commit.")
    Signed-off-by: Peter Mamonov <pmamonov@gmail.com>
    Reviewed-by: Andrew Lunn <andrew@lunn.ch>
    Signed-off-by: Jakub Kicinski <jakub.kicinski@netronome.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 81ab2c2abc45ef2aaba7182424632e6937f3b561
Author: Xin Long <lucien.xin@gmail.com>
Date:   Mon Sep 23 17:02:46 2019 +0800

    macsec: drop skb sk before calling gro_cells_receive

    [ Upstream commit ba56d8ce38c8252fff5b745db3899cf092578ede ]

    Fei Liu reported a crash when doing netperf on a topo of macsec
    dev over veth:

      [  448.919128] refcount_t: underflow; use-after-free.
      [  449.090460] Call trace:
      [  449.092895]  refcount_sub_and_test+0xb4/0xc0
      [  449.097155]  tcp_wfree+0x2c/0x150
      [  449.100460]  ip_rcv+0x1d4/0x3a8
      [  449.103591]  __netif_receive_skb_core+0x554/0xae0
      [  449.108282]  __netif_receive_skb+0x28/0x78
      [  449.112366]  netif_receive_skb_internal+0x54/0x100
      [  449.117144]  napi_gro_complete+0x70/0xc0
      [  449.121054]  napi_gro_flush+0x6c/0x90
```

```
        [  449.124703]  napi_complete_done+0x50/0x130
        [  449.128788]  gro_cell_poll+0x8c/0xa8
        [  449.132351]  net_rx_action+0x16c/0x3f8
        [  449.136088]  __do_softirq+0x128/0x320
```

    The issue was caused by skb's true_size changed without its sk's
    sk_wmem_alloc increased in tcp/skb_gro_receive(). Later when the
    skb is being freed and the skb's truesize is subtracted from its
    sk's sk_wmem_alloc in tcp_wfree(), underflow occurs.

    macsec is calling gro_cells_receive() to receive a packet, which
    actually requires skb->sk to be NULL. However when macsec dev is
    over veth, it's possible the skb->sk is still set if the skb was
    not unshared or expanded from the peer veth.

    ip_rcv() is calling skb_orphan() to drop the skb's sk for tproxy,
    but it is too late for macsec's calling gro_cells_receive(). So
    fix it by dropping the skb's sk earlier on rx path of macsec.

    Fixes: 5491e7c6b1a9 ("macsec: enable GRO and RPS on macsec devices")
    Reported-by: Xiumei Mu <xmu@redhat.com>
    Reported-by: Fei Liu <feliu@redhat.com>
    Signed-off-by: Xin Long <lucien.xin@gmail.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ecc265624956ea784cb2bd2b31a95bd54c4f5f13
Author: Jason A. Donenfeld <Jason@zx2c4.com>
Date: Tue Sep 24 16:01:28 2019 +0200

    ipv6: do not free rt if FIB_LOOKUP_NOREF is set on suppress rule

    [Upstream commit ca7a03c4175366a92cee0ccc4fec0038c3266e26]

    Commit 7d9e5f422150 removed references from certain dsts, but accounting
    for this never translated down into the fib6 suppression code. This bug
    was triggered by WireGuard users who use wg-quick(8), which uses the
    "suppress-prefix" directive to ip-rule(8) for routing all of their
    internet traffic without routing loops. The test case added here
    causes the reference underflow by causing packets to evaluate a suppress
    rule.

    Fixes: 7d9e5f422150 ("ipv6: convert major tx path to use RT6_LOOKUP_F_DST_NOREF")
    Signed-off-by: Jason A. Donenfeld <Jason@zx2c4.com>
    Acked-by: Wei Wang <weiwan@google.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit cf7665c17ad727ab056a55dbe6f670f285dd4864
Author: Bjørn Mork <bjorn@mork.no>
Date: Wed Sep 18 14:01:46 2019 +0200

    cdc_ncm: fix divide-by-zero caused by invalid wMaxPacketSize

    [Upstream commit 3fe4b3351301660653a2bc73f2226da0ebd2b95e]

    Endpoints with zero wMaxPacketSize are not usable for transferring
    data. Ignore such endpoints when looking for valid in, out and
    status pipes, to make the driver more robust against invalid and
    meaningless descriptors.

    The wMaxPacketSize of the out pipe is used as divisor. So this change
    fixes a divide-by-zero bug.

    Reported-by: syzbot+ce366e2b8296e25d84f5@syzkaller.appspotmail.com

```
    Signed-off-by: Bjørn Mork <bjorn@mork.no>
    Signed-off-by: Jakub Kicinski <jakub.kicinski@netronome.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f824b34ad978ced7a5f40700e5f8ad6fe8245d3c
Author: Uwe Kleine-König <u.kleine-koenig@pengutronix.de>
Date:   Fri Sep 20 16:08:21 2019 +0200

    arcnet: provide a buffer big enough to actually receive packets

    [ Upstream commit 108639aac35eb57f1d0e8333f5fc8c7ff68df938 ]

    struct archdr is only big enough to hold the header of various types of
    arcnet packets. So to provide enough space to hold the data read from
    hardware provide a buffer large enough to hold a packet with maximal
    size.

    The problem was noticed by the stack protector which makes the kernel
    oops.

    Signed-off-by: Uwe Kleine-König <u.kleine-koenig@pengutronix.de>
    Acked-by: Michael Grzeschik <m.grzeschik@pengutronix.de>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```