```
commit 577669778464e991c12291f5833bd627d2102bcb
Author: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
Date:   Fri Jul 26 09:11:12 2019 +0200

    Linux 5.2.3

commit b942dcdab8d19bdffe8170f5e0abcc206979c9b7
Author: Junxiao Bi <junxiao.bi@oracle.com>
Date:   Tue Jul 9 17:17:19 2019 -0700

    dm bufio: fix deadlock with loop device

    commit bd293d071ffe65e645b4d8104f9d8fe15ea13862 upstream.

    When thin-volume is built on loop device, if available memory is low,
    the following deadlock can be triggered:

    One process P1 allocates memory with GFP_FS flag, direct alloc fails,
    memory reclaim invokes memory shrinker in dm_bufio, dm_bufio_shrink_scan()
    runs, mutex dm_bufio_client->lock is acquired, then P1 waits for dm_buffer
    IO to complete in __try_evict_buffer().

    But this IO may never complete if issued to an underlying loop device
    that forwards it using direct-IO, which allocates memory using
    GFP_KERNEL (see: do_blockdev_direct_IO()).  If allocation fails, memory
    reclaim will invoke memory shrinker in dm_bufio, dm_bufio_shrink_scan()
    will be invoked, and since the mutex is already held by P1 the loop
    thread will hang, and IO will never complete.  Resulting in ABBA
    deadlock.

    Cc: stable@vger.kernel.org
    Signed-off-by: Junxiao Bi <junxiao.bi@oracle.com>
    Signed-off-by: Mike Snitzer <snitzer@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 8e4c803e7f5fdd611b0b9e887f40a2bd191dc26f
Author: Mike Snitzer <snitzer@redhat.com>
Date:   Tue Jul 2 15:50:08 2019 -0400

    dm thin metadata: check if in fail_io mode when setting needs_check

    commit 54fa16ee532705985e6c946da455856f18f63ee1 upstream.

    Check if in fail_io mode at start of dm_pool_metadata_set_needs_check().
    Otherwise dm_pool_metadata_set_needs_check()'s superblock_lock() can
    crash in dm_bm_write_lock() while accessing the block manager object
    that was previously destroyed as part of a failed
    dm_pool_abort_metadata() that ultimately set fail_io to begin with.

    Also, update DMERR() message to more accurately describe
    superblock_lock() failure.

    Cc: stable@vger.kernel.org
    Reported-by: Zdenek Kabelac <zkabelac@redhat.com>
    Signed-off-by: Mike Snitzer <snitzer@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 71e19cde733daa870f22d47791378735e4f6cab2
Author: Bjorn Andersson <bjorn.andersson@linaro.org>
Date:   Tue Jun 4 16:24:43 2019 -0700

    phy: qcom-qmp: Correct READY_STATUS poll break condition

    commit 885bd765963b42c380db442db7f1c0f2a26076fa upstream.
```

```
    After issuing a PHY_START request to the QMP, the hardware documentation
    states that the software should wait for the PCS_READY_STATUS to become
    1.

    With the introduction of commit c9b589791fc1 ("phy: qcom: Utilize UFS
    reset controller") an additional 1ms delay was introduced between the
    start request and the check of the status bit. This greatly increases
    the chances for the hardware to actually becoming ready before the
    status bit is read.

    The result can be seen in that UFS PHY enabling is now reported as a
    failure in 10% of the boots on SDM845, which is a clear regression from
    the previous rare/occasional failure.

    This patch fixes the "break condition" of the poll to check for the
    correct state of the status bit.

    Unfortunately PCIe on 8996 and 8998 does not specify the mask_pcs_ready
    register, which means that the code checks a bit that's always 0. So the
    patch also fixes these, in order to not regress these targets.

    Fixes: 73d7ec899bd8 ("phy: qcom-qmp: Add msm8998 PCIe QMP PHY support")
    Fixes: e78f3d15e115 ("phy: qcom-qmp: new qmp phy driver for qcom-chipsets")
    Cc: stable@vger.kernel.org
    Cc: Evan Green <evgreen@chromium.org>
    Cc: Marc Gonzalez <marc.w.gonzalez@free.fr>
    Cc: Vivek Gautam <vivek.gautam@codeaurora.org>
    Reviewed-by: Evan Green <evgreen@chromium.org>
    Reviewed-by: Niklas Cassel <niklas.cassel@linaro.org>
    Reviewed-by: Marc Gonzalez <marc.w.gonzalez@free.fr>
    Tested-by: Marc Gonzalez <marc.w.gonzalez@free.fr>
    Signed-off-by: Bjorn Andersson <bjorn.andersson@linaro.org>
    Signed-off-by: Kishon Vijay Abraham I <kishon@ti.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 6f46c8c5d091ed4be38f46ae7931b06dbf301735
Author: Norbert Manthey <nmanthey@amazon.de>
Date:   Fri Jul 5 15:06:00 2019 +0200

    pstore: Fix double-free in pstore_mkfile() failure path

    commit 4c6d80e1144bdf48cae6b602ae30d41f3e5c76a9 upstream.

    The pstore_mkfile() function is passed a pointer to a struct
    pstore_record. On success it consumes this 'record' pointer and
    references it from the created inode.

    On failure, however, it may or may not free the record. There are even
    two different code paths which return -ENOMEM -- one of which does and
    the other doesn't free the record.

    Make the behaviour deterministic by never consuming and freeing the
    record when returning failure, allowing the caller to do the cleanup
    consistently.

    Signed-off-by: Norbert Manthey <nmanthey@amazon.de>
    Link: https://lore.kernel.org/r/1562331960-26198-1-git-send-email-nmanthey@amazon.de
    Fixes: 83f70f0769ddd ("pstore: Do not duplicate record metadata")
    Fixes: 1dfff7dd67d1a ("pstore: Pass record contents instead of copying")
    Cc: stable@vger.kernel.org
    [kees: also move "private" allocation location, rename inode cleanup label]
    Signed-off-by: Kees Cook <keescook@chromium.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit e07f18330387bd166a9cb4fec061fad4608cb597
Author: Josua Mayer <josua@solid-run.com>
Date:   Tue Jul 9 15:00:58 2019 +0200

    dt-bindings: allow up to four clocks for orion-mdio

    commit 80785f5a22e9073e2ded5958feb7f220e066d17b upstream.

    Armada 8040 needs four clocks to be enabled for MDIO accesses to work.
    Update the binding to allow the extra clock to be specified.

    Cc: stable@vger.kernel.org
    Fixes: 6d6a331f44a1 ("dt-bindings: allow up to three clocks for orion-mdio")
    Reviewed-by: Andrew Lunn <andrew@lunn.ch>
    Signed-off-by: Josua Mayer <josua@solid-run.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f9559f00cb0e09716f71eee827b2a9fb1cebdc5b
Author: Josua Mayer <josua@solid-run.com>
Date:   Tue Jul 9 15:00:59 2019 +0200

    net: mvmdio: allow up to four clocks to be specified for orion-mdio

    commit 4aabed699c400810981d3dda170f05fa4d782905 upstream.

    Allow up to four clocks to be specified and enabled for the orion-mdio
    interface, which are required by the Armada 8k and defined in
    armada-cp110.dtsi.

    Fixes a hang in probing the mvmdio driver that was encountered on the
    Clearfog GT 8K with all drivers built as modules, but also affects other
    boards such as the MacchiatoBIN.

    Cc: stable@vger.kernel.org
    Fixes: 96cb43423822 ("net: mvmdio: allow up to three clocks to be specified for orion-
mdio")
    Reviewed-by: Andrew Lunn <andrew@lunn.ch>
    Signed-off-by: Josua Mayer <josua@solid-run.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit d483ea783a93f5cd59c93aa0314fde7267a0968f
Author: Tejun Heo <tj@kernel.org>
Date:   Thu Jun 13 15:30:38 2019 -0700

    blkcg: update blkcg_print_stat() to handle larger outputs

    commit f539da82f2158916e154d206054e0efd5df7ab61 upstream.

    Depending on the number of devices, blkcg stats can go over the
    default seqfile buf size.  seqfile normally retries with a larger
    buffer but since the ->pd_stat() addition, blkcg_print_stat() doesn't
    tell seqfile that overflow has happened and the output gets printed
    truncated.  Fix it by calling seq_commit() w/ -1 on possible
    overflows.

    Signed-off-by: Tejun Heo <tj@kernel.org>
    Fixes: 903d23f0a354 ("blk-cgroup: allow controllers to output their own stats")
    Cc: stable@vger.kernel.org # v4.19+
    Cc: Josef Bacik <jbacik@fb.com>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c600458bb020e592ad29bf2449957e0a43f026b2
```

```
Author: Tejun Heo <tj@kernel.org>
Date:   Thu Jun 13 15:30:37 2019 -0700

    blk-iolatency: clear use_delay when io.latency is set to zero

    commit 5de0073fcd50cc1f150895a7bb04d3cf8067b1d7 upstream.

    If use_delay was non-zero when the latency target of a cgroup was set
    to zero, it will stay stuck until io.latency is enabled on the cgroup
    again.  This keeps readahead disabled for the cgroup impacting
    performance negatively.

    Signed-off-by: Tejun Heo <tj@kernel.org>
    Cc: Josef Bacik <jbacik@fb.com>
    Fixes: d70675121546 ("block: introduce blk-iolatency io controller")
    Cc: stable@vger.kernel.org # v4.19+
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 86ace02c805962af262d03421949c2fa1d7a7fff
Author: Peng Fan <peng.fan@nxp.com>
Date:   Fri May 31 15:56:38 2019 +0800

    clk: imx: imx8mm: correct audio_pll2_clk to audio_pll2_out

    commit 5b933e28d8b1fbdc7fbac4bfc569f3b152c3dd59 upstream.

    There is no audio_pll2_clk registered, it should be audio_pll2_out.

    Cc: <stable@vger.kernel.org>
    Fixes: ba5625c3e272 ("clk: imx: Add clock driver support for imx8mm")
    Signed-off-by: Peng Fan <peng.fan@nxp.com>
    Signed-off-by: Shawn Guo <shawnguo@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a613a7957a05bea2bab9bd54c3af10b23613b5fb
Author: Konstantin Khlebnikov <khlebnikov@yandex-team.ru>
Date:   Mon Jul 8 18:29:57 2019 +0300

    blk-throttle: fix zero wait time for iops throttled group

    commit 3a10f999ffd464d01c5a05592a15470a3c4bbc36 upstream.

    After commit 991f61fe7e1d ("Blk-throttle: reduce tail io latency when
    iops limit is enforced") wait time could be zero even if group is
    throttled and cannot issue requests right now. As a result
    throtl_select_dispatch() turns into busy-loop under irq-safe queue
    spinlock.

    Fix is simple: always round up target time to the next throttle slice.

    Fixes: 991f61fe7e1d ("Blk-throttle: reduce tail io latency when iops limit is enforced")
    Signed-off-by: Konstantin Khlebnikov <khlebnikov@yandex-team.ru>
    Cc: stable@vger.kernel.org # v4.19+
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3201a60323f05b3600c95f3d41778862b52b5e3b
Author: Lee, Chiasheng <chiasheng.lee@intel.com>
Date:   Thu Jun 20 10:56:04 2019 +0300

    usb: Handle USB3 remote wakeup for LPM enabled devices correctly

    commit e244c4699f859cf7149b0781b1894c7996a8a1df upstream.
```

```
    With Link Power Management (LPM) enabled USB3 links transition to low
    power U1/U2 link states from U0 state automatically.

    Current hub code detects USB3 remote wakeups by checking if the software
    state still shows suspended, but the link has transitioned from suspended
    U3 to enabled U0 state.

    As it takes some time before the hub thread reads the port link state
    after a USB3 wake notification, the link may have transitioned from U0
    to U1/U2, and wake is not detected by hub code.

    Fix this by handling U1/U2 states in the same way as U0 in USB3 wakeup
    handling

    This patch should be added to stable kernels since 4.13 where LPM was
    kept enabled during suspend/resume

    Cc: <stable@vger.kernel.org> # v4.13+
    Signed-off-by: Lee, Chiasheng <chiasheng.lee@intel.com>
    Signed-off-by: Mathias Nyman <mathias.nyman@linux.intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 111b055e43cbca1761eaea0812e35dea556cb8d5
Author: Matthew Wilcox (Oracle) <willy@infradead.org>
Date:   Wed Jul 3 23:21:25 2019 -0400

    dax: Fix missed wakeup with PMD faults

    commit 23c84eb7837514e16d79ed6d849b13745e0ce688 upstream.

    RocksDB can hang indefinitely when using a DAX file.  This is due to
    a bug in the XArray conversion when handling a PMD fault and finding a
    PTE entry.  We use the wrong index in the hash and end up waiting on
    the wrong waitqueue.

    There's actually no need to wait; if we find a PTE entry while looking
    for a PMD entry, we can return immediately as we know we should fall
    back to a PTE fault (which may not conflict with the lock held).

    We reuse the XA_RETRY_ENTRY to signal a conflicting entry was found.
    This value can never be found in an XArray while holding its lock, so
    it does not create an ambiguity.

    Cc: <stable@vger.kernel.org>
    Link: http://lkml.kernel.org/r/CAPcyv4hwHpX-MkUEqxwdTj7wCCZCN4RV-
L4jsnuwLGyL_UEG4A@mail.gmail.com
    Fixes: b15cd800682f ("dax: Convert page fault handlers to XArray")
    Signed-off-by: Matthew Wilcox (Oracle) <willy@infradead.org>
    Tested-by: Dan Williams <dan.j.williams@intel.com>
    Reported-by: Robert Barror <robert.barror@intel.com>
    Reported-by: Seema Pandit <seema.pandit@intel.com>
    Reviewed-by: Jan Kara <jack@suse.cz>
    Signed-off-by: Dan Williams <dan.j.williams@intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit bd2828a819419601e5a204e92298bdfe5647cb0e
Author: Szymon Janc <szymon.janc@codecoup.pl>
Date:   Wed Jun 19 00:47:47 2019 +0200

    Bluetooth: Add SMP workaround Microsoft Surface Precision Mouse bug

    commit 1d87b88ba26eabd4745e158ecfd87c93a9b51dc2 upstream.

    Microsoft Surface Precision Mouse provides bogus identity address when
    pairing. It connects with Static Random address but provides Public
```

```
    Address in SMP Identity Address Information PDU. Address has same
    value but type is different. Workaround this by dropping IRK if ID
    address discrepancy is detected.

    > HCI Event: LE Meta Event (0x3e) plen 19
          LE Connection Complete (0x01)
            Status: Success (0x00)
            Handle: 75
            Role: Master (0x00)
            Peer address type: Random (0x01)
            Peer address: E0:52:33:93:3B:21 (Static)
            Connection interval: 50.00 msec (0x0028)
            Connection latency: 0 (0x0000)
            Supervision timeout: 420 msec (0x002a)
            Master clock accuracy: 0x00

    ....

    > ACL Data RX: Handle 75 flags 0x02 dlen 12
          SMP: Identity Address Information (0x09) len 7
            Address type: Public (0x00)
            Address: E0:52:33:93:3B:21

    Signed-off-by: Szymon Janc <szymon.janc@codecoup.pl>
    Tested-by: Maarten Fonville <maarten.fonville@gmail.com>
    Bugzilla: https://bugzilla.kernel.org/show_bug.cgi?id=199461
    Cc: stable@vger.kernel.org
    Signed-off-by: Marcel Holtmann <marcel@holtmann.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 58fae3632c7a90216e1969cafb1690e7a9525452
Author: Alexander Shishkin <alexander.shishkin@linux.intel.com>
Date:   Fri Jun 21 19:19:29 2019 +0300

    intel_th: msu: Fix single mode with disabled IOMMU

    commit 918b8646497b5dba6ae82d4a7325f01b258972b9 upstream.

    Commit 4e0eaf239fb3 ("intel_th: msu: Fix single mode with IOMMU") switched
    the single mode code to use dma mapping pages obtained from the page
    allocator, but with IOMMU disabled, that may lead to using SWIOTLB bounce
    buffers and without additional sync'ing, produces empty trace buffers.

    Fix this by using a DMA32 GFP flag to the page allocation in single mode,
    as the device supports full 32-bit DMA addressing.

    Signed-off-by: Alexander Shishkin <alexander.shishkin@linux.intel.com>
    Fixes: 4e0eaf239fb3 ("intel_th: msu: Fix single mode with IOMMU")
    Reviewed-by: Andy Shevchenko <andriy.shevchenko@linux.intel.com>
    Reported-by: Ammy Yi <ammy.yi@intel.com>
    Cc: stable <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20190621161930.60785-4-
alexander.shishkin@linux.intel.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 320a36ce38021b2628ba20ab6f339127614b6ec7
Author: YueHaibing <yuehaibing@huawei.com>
Date:   Fri Jun 21 19:19:28 2019 +0300

    intel_th: msu: Remove set but not used variable 'last'

    commit 9800db282dff675dd700d5985d90b605c34b5ccd upstream.

    Commit aad14ad3cf3a ("intel_th: msu: Add current window tracking") added
    the following gcc warning:
```

```
    > drivers/hwtracing/intel_th/msu.c: In function msc_win_switch:
    > drivers/hwtracing/intel_th/msu.c:1389:21: warning: variable last set but
    > not used [-Wunused-but-set-variable]

    Fix it by removing the variable.

    Signed-off-by: YueHaibing <yuehaibing@huawei.com>
    Fixes: aad14ad3cf3a ("intel_th: msu: Add current window tracking")
    Reviewed-by: Andy Shevchenko <andriy.shevchenko@linux.intel.com>
    Signed-off-by: Alexander Shishkin <alexander.shishkin@linux.intel.com>
    Cc: stable <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20190621161930.60785-3-
alexander.shishkin@linux.intel.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 13d7c7e75eb9562b680f4eb4bd85c869f29143b4
Author: liaoweixiong <liaoweixiong@allwinnertech.com>
Date:   Fri Jun 28 12:14:46 2019 +0800

    mtd: spinand: read returns badly if the last page has bitflips

    commit b83408b580eccf8d2797cd6cb9ae42c2a28656a7 upstream.

    In case of the last page containing bitflips (ret > 0),
    spinand_mtd_read() will return that number of bitflips for the last
    page while it should instead return max_bitflips like it does when the
    last page read returns with 0.

    Signed-off-by: Weixiong Liao <liaoweixiong@allwinnertech.com>
    Reviewed-by: Boris Brezillon <boris.brezillon@collabora.com>
    Reviewed-by: Frieder Schrempf <frieder.schrempf@kontron.de>
    Cc: stable@vger.kernel.org
    Fixes: 7529df465248 ("mtd: nand: Add core infrastructure to support SPI NANDs")
    Signed-off-by: Miquel Raynal <miquel.raynal@bootlin.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 6d1e846312f471666164483c7929f6d1837ea2e8
Author: Xiaolei Li <xiaolei.li@mediatek.com>
Date:   Tue May 7 18:25:38 2019 +0800

    mtd: rawnand: mtk: Correct low level time calculation of r/w cycle

    commit e1884ffddacc0424d7e785e6f8087bd12f7196db upstream.

    At present, the flow of calculating AC timing of read/write cycle in SDR
    mode is that:
    At first, calculate high hold time which is valid for both read and write
    cycle using the max value between tREH_min and tWH_min.
    Secondly, calculate WE# pulse width using tWP_min.
    Thridly, calculate RE# pulse width using the bigger one between tREA_max
    and tRP_min.

    But NAND SPEC shows that Controller should also meet write/read cycle time.
    That is write cycle time should be more than tWC_min and read cycle should
    be more than tRC_min. Obviously, we do not achieve that now.

    This patch corrects the low level time calculation to meet minimum
    read/write cycle time required. After getting the high hold time, WE# low
    level time will be promised to meet tWP_min and tWC_min requirement,
    and RE# low level time will be promised to meet tREA_max, tRP_min and
    tRC_min requirement.

    Fixes: edfee3619c49 ("mtd: nand: mtk: add ->setup_data_interface() hook")
    Cc: stable@vger.kernel.org # v4.17+
```

```
    Signed-off-by: Xiaolei Li <xiaolei.li@mediatek.com>
    Reviewed-by: Miquel Raynal <miquel.raynal@bootlin.com>
    Signed-off-by: Miquel Raynal <miquel.raynal@bootlin.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit abbd63bf086d03897569a5a6f56f87e6af222b58
Author: Dan Carpenter <dan.carpenter@oracle.com>
Date:   Wed Jul 4 12:35:56 2018 +0300

    eCryptfs: fix a couple type promotion bugs

    commit 0bdf8a8245fdea6f075a5fede833a5fcf1b3466c upstream.

    ECRYPTFS_SIZE_AND_MARKER_BYTES is type size_t, so if "rc" is negative
    that gets type promoted to a high positive value and treated as success.

    Fixes: 778aeb42a708 ("eCryptfs: Cleanup and optimize ecryptfs_lookup_interpose()")
    Signed-off-by: Dan Carpenter <dan.carpenter@oracle.com>
    [tyhicks: Use "if/else if" rather than "if/if"]
    Cc: stable@vger.kernel.org
    Signed-off-by: Tyler Hicks <tyhicks@canonical.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit d7245d252f55cc790b19b2f9c6ac40cfceab2a79
Author: Jorge Ramirez-Ortiz <jorge.ramirez-ortiz@linaro.org>
Date:   Mon Jul 1 17:01:25 2019 +0200

    mmc: sdhci-msm: fix mutex while in spinlock

    commit 5e6b6651d22de109ebf48ca00d0373bc2c0cc080 upstream.

    mutexes can sleep and therefore should not be taken while holding a
    spinlock. move clk_get_rate (can sleep) outside the spinlock protected
    region.

    Fixes: 83736352e0ca ("mmc: sdhci-msm: Update DLL reset sequence")
    Cc: stable@vger.kernel.org
    Signed-off-by: Jorge Ramirez-Ortiz <jorge.ramirez-ortiz@linaro.org>
    Reviewed-by: Bjorn Andersson <bjorn.andersson@linaro.org>
    Reviewed-by: Vinod Koul <vkoul@kernel.org>
    Acked-by: Adrian Hunter <adrian.hunter@intel.com>
    Signed-off-by: Ulf Hansson <ulf.hansson@linaro.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a63f2ef2082312a7678187296e67d2cb1273ad63
Author: Nathan Lynch <nathanl@linux.ibm.com>
Date:   Fri Jun 7 00:04:07 2019 -0500

    powerpc/pseries: Fix oops in hotplug memory notifier

    commit 0aa82c482ab2ece530a6f44897b63b274bb43c8e upstream.

    During post-migration device tree updates, we can oops in
    pseries_update_drconf_memory() if the source device tree has an
    ibm,dynamic-memory-v2 property and the destination has a
    ibm,dynamic_memory (v1) property. The notifier processes an "update"
    for the ibm,dynamic-memory property but it's really an add in this
    scenario. So make sure the old property object is there before
    dereferencing it.

    Fixes: 2b31e3aec1db ("powerpc/drmem: Add support for ibm, dynamic-memory-v2 property")
    Cc: stable@vger.kernel.org # v4.16+
    Signed-off-by: Nathan Lynch <nathanl@linux.ibm.com>
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit dd29ff8324260ae702227e5aba53f78efeec29c7
Author: Greg Kurz <groug@kaod.org>
Date:   Wed May 15 12:05:01 2019 +0200

    powerpc/pseries: Fix xive=off command line

    commit a3bf9fbdad600b1e4335dd90979f8d6072e4f602 upstream.

    On POWER9, if the hypervisor supports XIVE exploitation mode, the
    guest OS will unconditionally requests for the XIVE interrupt mode
    even if XIVE was deactivated with the kernel command line xive=off.
    Later on, when the spapr XIVE init code handles xive=off, it disables
    XIVE and tries to fall back on the legacy mode XICS.

    This discrepency causes a kernel panic because the hypervisor is
    configured to provide the XIVE interrupt mode to the guest :

      kernel BUG at arch/powerpc/sysdev/xics/xics-common.c:135!
      ...
      NIP xics_smp_probe+0x38/0x98
      LR  xics_smp_probe+0x2c/0x98
      Call Trace:
        xics_smp_probe+0x2c/0x98 (unreliable)
        pSeries_smp_probe+0x40/0xa0
        smp_prepare_cpus+0x62c/0x6ec
        kernel_init_freeable+0x148/0x448
        kernel_init+0x2c/0x148
        ret_from_kernel_thread+0x5c/0x68

    Look for xive=off during prom_init and don't ask for XIVE in this
    case. One exception though: if the host only supports XIVE, we still
    want to boot so we ignore xive=off.

    Similarly, have the spapr XIVE init code to looking at the interrupt
    mode negotiated during CAS, and ignore xive=off if the hypervisor only
    supports XIVE.

    Fixes: eac1e731b59e ("powerpc/xive: guest exploitation of the XIVE interrupt
controller")
    Cc: stable@vger.kernel.org # v4.20
    Reported-by: Pavithra R. Prakash <pavrampu@in.ibm.com>
    Signed-off-by: Greg Kurz <groug@kaod.org>
    Reviewed-by: Cédric Le Goater <clg@kaod.org>
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 82150014031da71a4ead7092550c1a1de037816a
Author: Alexey Kardashevskiy <aik@ozlabs.ru>
Date:   Fri Jun 28 16:53:00 2019 +1000

    powerpc/powernv: Fix stale iommu table base after VFIO

    commit 5636427d087a55842c1a199dfb839e6545d30e5d upstream.

    The powernv platform uses @dma_iommu_ops for non-bypass DMA. These ops
    need an iommu_table pointer which is stored in
    dev->archdata.iommu_table_base. It is initialized during
    pcibios_setup_device() which handles boot time devices. However when a
    device is taken from the system in order to pass it through, the
    default IOMMU table is destroyed but the pointer in a device is not
    updated; also when a device is returned back to the system, a new
    table pointer is not stored in dev->archdata.iommu_table_base either.
    So when a just returned device tries using IOMMU, it crashes on
    accessing stale iommu_table or its members.
```

```
    This calls set_iommu_table_base() when the default window is created.
    Note it used to be there before but was wrongly removed (see "fixes").
    It did not appear before as these days most devices simply use bypass.

    This adds set_iommu_table_base(NULL) when a device is taken from the
    system to make it clear that IOMMU DMA cannot be used past that point.

    Fixes: c4e9d3c1e65a ("powerpc/powernv/pseries: Rework device adding to IOMMU groups")
    Cc: stable@vger.kernel.org # v5.0+
    Signed-off-by: Alexey Kardashevskiy <aik@ozlabs.ru>
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit d71acd3f404e65b4cfcbdd3ae096c15c847d7d04
Author: Athira Rajeev <atrajeev@linux.vnet.ibm.com>
Date:   Tue Jul 2 16:28:36 2019 +0530

    powerpc/powernv/idle: Fix restore of SPRN_LDBAR for POWER9 stop state.

    commit f5a9e488d62360c91c5770bd55a0b40e419a71ce upstream.

    commit 10d91611f426 ("powerpc/64s: Reimplement book3s idle code in C")
    reimplemented book3S code to pltform/powernv/idle.c. But when doing so
    missed to add the per-thread LDBAR update in the core_woken path of
    the power9_idle_stop(). Patch fixes the same.

    Fixes: 10d91611f426 ("powerpc/64s: Reimplement book3s idle code in C")
    Cc: stable@vger.kernel.org # v5.2+
    Signed-off-by: Athira Rajeev <atrajeev@linux.vnet.ibm.com>
    Signed-off-by: Madhavan Srinivasan <maddy@linux.vnet.ibm.com>
    Reviewed-by: Nicholas Piggin <npiggin@gmail.com>
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Link: https://lore.kernel.org/r/20190702105836.26695-1-maddy@linux.vnet.ibm.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 03eccf2da689022410fb70164111ac13766c1b68
Author: Greg Kurz <groug@kaod.org>
Date:   Fri Apr 19 17:34:13 2019 +0200

    powerpc/powernv/npu: Fix reference leak

    commit 02c5f5394918b9b47ff4357b1b18335768cd867d upstream.

    Since 902bdc57451c, get_pci_dev() calls pci_get_domain_bus_and_slot(). This
    has the effect of incrementing the reference count of the PCI device, as
    explained in drivers/pci/search.c:

     * Given a PCI domain, bus, and slot/function number, the desired PCI
     * device is located in the list of PCI devices. If the device is
     * found, its reference count is increased and this function returns a
     * pointer to its data structure.  The caller must decrement the
     * reference count by calling pci_dev_put().  If no device is found,
     * %NULL is returned.

    Nothing was done to call pci_dev_put() and the reference count of GPU and
    NPU PCI devices rockets up.

    A natural way to fix this would be to teach the callers about the change,
    so that they call pci_dev_put() when done with the pointer. This turns
    out to be quite intrusive, as it affects many paths in npu-dma.c,
    pci-ioda.c and vfio_pci_nvlink2.c. Also, the issue appeared in 4.16 and
    some affected code got moved around since then: it would be problematic
    to backport the fix to stable releases.
```

```
    All that code never cared for reference counting anyway. Call pci_dev_put()
    from get_pci_dev() to revert to the previous behavior.

    Fixes: 902bdc57451c ("powerpc/powernv/idoa: Remove unnecessary pcidev from pci_dn")
    Cc: stable@vger.kernel.org # v4.16
    Signed-off-by: Greg Kurz <groug@kaod.org>
    Reviewed-by: Alexey Kardashevskiy <aik@ozlabs.ru>
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 2df756cdaf3d3c031bfa81d0d5dd03f1a86111ba
Author: Ravi Bangoria <ravi.bangoria@linux.ibm.com>
Date:   Thu Jun 13 09:00:14 2019 +0530

    powerpc/watchpoint: Restore NV GPRs while returning from exception

    commit f474c28fbcbe42faca4eb415172c07d76adcb819 upstream.

    powerpc hardware triggers watchpoint before executing the instruction.
    To make trigger-after-execute behavior, kernel emulates the
    instruction. If the instruction is 'load something into non-volatile
    register', exception handler should restore emulated register state
    while returning back, otherwise there will be register state
    corruption. eg, adding a watchpoint on a list can corrput the list:

      # cat /proc/kallsyms | grep kthread_create_list
      c00000000121c8b8 d kthread_create_list

    Add watchpoint on kthread_create_list->prev:

      # perf record -e mem:0xc00000000121c8c0

    Run some workload such that new kthread gets invoked. eg, I just
    logged out from console:

      list_add corruption. next->prev should be prev (c000000001214e00), \
            but was c00000000121c8b8. (next=c00000000121c8b8).
      WARNING: CPU: 59 PID: 309 at lib/list_debug.c:25 __list_add_valid+0xb4/0xc0
      CPU: 59 PID: 309 Comm: kworker/59:0 Kdump: loaded Not tainted 5.1.0-rc7+ #69
      ...
      NIP __list_add_valid+0xb4/0xc0
      LR __list_add_valid+0xb0/0xc0
      Call Trace:
      __list_add_valid+0xb0/0xc0 (unreliable)
      __kthread_create_on_node+0xe0/0x260
      kthread_create_on_node+0x34/0x50
      create_worker+0xe8/0x260
      worker_thread+0x444/0x560
      kthread+0x160/0x1a0
      ret_from_kernel_thread+0x5c/0x70

    List corruption happened because it uses 'load into non-volatile
    register' instruction:

    Snippet from __kthread_create_on_node:

      c000000000136be8:     addis   r29,r2,-19
      c000000000136bec:     ld      r29,31424(r29)
            if (!__list_add_valid(new, prev, next))
      c000000000136bf0:     mr      r3,r30
      c000000000136bf4:     mr      r5,r28
      c000000000136bf8:     mr      r4,r29
      c000000000136bfc:     bl      c00000000059a2f8 <__list_add_valid+0x8>

    Register state from WARN_ON():
```

```
      GPR00: c00000000059a3a0 c000007ff23afb50 c000000001344e00 0000000000000075
      GPR04: 0000000000000000 0000000000000000 0000001852af8bc1 0000000000000000
      GPR08: 0000000000000001 0000000000000007 0000000000000006 00000000000004aa
      GPR12: 0000000000000000 c000007ffffeb080 c000000000137038 c000005ff62aaa00
      GPR16: 0000000000000000 0000000000000000 c000007fffbe7600 c000007fffbe7370
      GPR20: c000007fffbe7320 c000007fffbe7300 c000000001373a00 0000000000000000
      GPR24: fffffffffffffef7 c00000000012e320 c000007ff23afcb0 c000000000cb8628
      GPR28: c00000000121c8b8 c000000001214e00 c000007fef5b17e8 c000007fef5b17c0

    Watchpoint hit at 0xc000000000136bec.

      addis   r29,r2,-19
       => r29 = 0xc000000001344e00 + (-19 << 16)
       => r29 = 0xc000000001214e00

      ld      r29,31424(r29)
       => r29 = *(0xc000000001214e00 + 31424)
       => r29 = *(0xc00000000121c8c0)

    0xc00000000121c8c0 is where we placed a watchpoint and thus this
    instruction was emulated by emulate_step. But because handle_dabr_fault
    did not restore emulated register state, r29 still contains stale
    value in above register state.

    Fixes: 5aae8a5370802 ("powerpc, hw_breakpoints: Implement hw_breakpoints for 64-bit
server processors")
    Signed-off-by: Ravi Bangoria <ravi.bangoria@linux.ibm.com>
    Cc: stable@vger.kernel.org # 2.6.36+
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 9f6a2e40dc9edb9b5a32639fa20834cc3bd0d692
Author: Andreas Schwab <schwab@linux-m68k.org>
Date:   Mon Jun 17 23:22:20 2019 +0200

    powerpc/mm/32s: fix condition that is always true

    commit 46c2478af610efb3212b8b08f74389d69899ef70 upstream.

    Move a misplaced paren that makes the condition always true.

    Fixes: 63b2bc619565 ("powerpc/mm/32s: Use BATs for STRICT_KERNEL_RWX")
    Cc: stable@vger.kernel.org # v5.1+
    Signed-off-by: Andreas Schwab <schwab@linux-m68k.org>
    Reviewed-by: Christophe Leroy <christophe.leroy@c-s.fr>
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1d2ee2a5757e2e04e90a3d777b391cf7bb8f99b1
Author: Christophe Leroy <christophe.leroy@c-s.fr>
Date:   Mon Jun 17 21:42:14 2019 +0000

    powerpc/32s: fix suspend/resume when IBATs 4-7 are used

    commit 6ecb78ef56e08d2119d337ae23cb951a640dc52d upstream.

    Previously, only IBAT1 and IBAT2 were used to map kernel linear mem.
    Since commit 63b2bc619565 ("powerpc/mm/32s: Use BATs for
    STRICT_KERNEL_RWX"), we may have all 8 BATs used for mapping
    kernel text. But the suspend/restore functions only save/restore
    BATs 0 to 3, and clears BATs 4 to 7.

    Make suspend and restore functions respectively save and reload
    the 8 BATs on CPUs having MMU_FTR_USE_HIGH_BATS feature.
```

```
    Reported-by: Andreas Schwab <schwab@linux-m68k.org>
    Cc: stable@vger.kernel.org
    Signed-off-by: Christophe Leroy <christophe.leroy@c-s.fr>
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit fd0a71299589acfff968fff4fa590707b11383ba
Author: Helge Deller <deller@gmx.de>
Date:   Tue Jul 16 21:43:11 2019 +0200

    parisc: Fix kernel panic due invalid values in IAOQ0 or IAOQ1

    commit 10835c854685393a921b68f529bf740fa7c9984d upstream.

    On parisc the privilege level of a process is stored in the lowest two bits of
    the instruction pointers (IAOQ0 and IAOQ1). On Linux we use privilege level 0
    for the kernel and privilege level 3 for user-space. So userspace should not be
    allowed to modify IAOQ0 or IAOQ1 of a ptraced process to change it's privilege
    level to e.g. 0 to try to gain kernel privileges.

    This patch prevents such modifications by always setting the two lowest bits to
    one (which relates to privilege level 3 for user-space) if IAOQ0 or IAOQ1 are
    modified via ptrace calls in the native and compat ptrace paths.

    Link: https://bugs.gentoo.org/481768
    Reported-by: Jeroen Roovers <jer@gentoo.org>
    Cc: <stable@vger.kernel.org>
    Tested-by: Rolf Eike Beer <eike-kernel@sf-tec.de>
    Signed-off-by: Helge Deller <deller@gmx.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1ca14ff51d9c2b336042160eb38909f78f7253eb
Author: Helge Deller <deller@gmx.de>
Date:   Tue Jul 16 21:16:26 2019 +0200

    parisc: Avoid kernel panic triggered by invalid kprobe

    commit 59a783dbc0d5fd6792aabff933055373b6dcbf2a upstream.

    When running gdb I was able to trigger this kernel panic:

     Kernel Fault: Code=26 (Data memory access rights trap) at addr 0000000000000060
     CPU: 0 PID: 1401 Comm: gdb-crash Not tainted 5.2.0-rc7-64bit+ #1053

          YZrvWESTHLNXBCVMcbcbcbcbOGFRQPDI
     PSW: 00001000000001000000000000001111 Not tainted
     r00-03  000000000804000f 0000000040dee1a0 0000000040c78cf0 00000000b8d50160
     r04-07  0000000040d2b1a0 000000004360a098 00000000bbbe87b8 0000000000000003
     r08-11  00000000fac20a70 00000000fac24160 00000000fac1bbe0 0000000000000000
     r12-15  00000000fabfb79a 00000000fac244a4 0000000000010000 0000000000000001
     r16-19  00000000bbbe87b8 00000000f8f02910 0000000000010034 0000000000000000
     r20-23  00000000fac24630 00000000fac24630 000000006474e552 00000000fac1aa52
     r24-27  0000000000000028 00000000bbbe87b8 00000000bbbe87b8 0000000040d2b1a0
     r28-31  0000000000000000 00000000b8d501c0 00000000b8d501f0 0000000003424000
     sr00-03  0000000000423000 0000000000000000 0000000000000000 0000000000423000
     sr04-07  0000000000000000 0000000000000000 0000000000000000 0000000000000000

     IASQ: 0000000000000000 0000000000000000 IAOQ: 0000000040c78cf0 0000000040c78cf4
      IIR: 539f00c0    ISR: 0000000000000000  IOR: 0000000000000060
      CPU:        0   CR30: 00000000b8d50000 CR31: 00000000d22345e2
      ORIG_R28: 0000000040250798
      IAOQ[0]: parisc_kprobe_ss_handler+0x58/0x170
      IAOQ[1]: parisc_kprobe_ss_handler+0x5c/0x170
      RP(r2): parisc_kprobe_ss_handler+0x58/0x170
```

```
       Backtrace:
        [<0000000040206ff8>] handle_interruption+0x178/0xbb8
       Kernel panic - not syncing: Kernel Fault
```

    Avoid this panic by checking the return value of kprobe_running() and
    skip kprobe if none is currently active.

    Cc: <stable@vger.kernel.org> # v5.2
    Acked-by: Sven Schnelle <svens@stackframe.org>
    Tested-by: Rolf Eike Beer <eike-kernel@sf-tec.de>
    Signed-off-by: Helge Deller <deller@gmx.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 325d2d7eedbf0401d798780e39098001e6ddceb3
Author: Helge Deller <deller@gmx.de>
Date: Thu Jul 4 03:44:17 2019 +0200

    parisc: Ensure userspace privilege for ptraced processes in regset functions

    commit 34c32fc603311a72cb558e5e337555434f64c27b upstream.

    On parisc the privilege level of a process is stored in the lowest two bits of
    the instruction pointers (IAOQ0 and IAOQ1). On Linux we use privilege level 0
    for the kernel and privilege level 3 for user-space. So userspace should not be
    allowed to modify IAOQ0 or IAOQ1 of a ptraced process to change it's privilege
    level to e.g. 0 to try to gain kernel privileges.

    This patch prevents such modifications in the regset support functions by
    always setting the two lowest bits to one (which relates to privilege level 3
    for user-space) if IAOQ0 or IAOQ1 are modified via ptrace regset calls.

    Link: https://bugs.gentoo.org/481768
    Cc: <stable@vger.kernel.org> # v4.7+
    Tested-by: Rolf Eike Beer <eike-kernel@sf-tec.de>
    Signed-off-by: Helge Deller <deller@gmx.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c40ecbe1f6ef867769590cec31a11dda60c17141
Author: Steve Longerbeam <slongerbeam@gmail.com>
Date: Tue May 21 18:03:13 2019 -0700

    gpu: ipu-v3: ipu-ic: Fix saturation bit offset in TPMEM

    commit 3d1f62c686acdedf5ed9642b763f3808d6a47d1e upstream.

    The saturation bit was being set at bit 9 in the second 32-bit word
    of the TPMEM CSC. This isn't correct, the saturation bit is bit 42,
    which is bit 10 of the second word.

    Fixes: 1aa8ea0d2bd5d ("gpu: ipu-v3: Add Image Converter unit")

    Signed-off-by: Steve Longerbeam <slongerbeam@gmail.com>
    Reviewed-by: Philipp Zabel <p.zabel@pengutronix.de>
    Cc: stable@vger.kernel.org
    Signed-off-by: Philipp Zabel <p.zabel@pengutronix.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1bbf2811db6ae281ee1ce460aecdee9f970c3c67
Author: Nadav Amit <namit@vmware.com>
Date: Thu Jul 18 15:57:31 2019 -0700

    resource: fix locking in find_next_iomem_res()

    commit 49f17c26c123b60fd1c74629eef077740d16ffc2 upstream.

```
    Since resources can be removed, locking should ensure that the resource
    is not removed while accessing it.  However, find_next_iomem_res() does
    not hold the lock while copying the data of the resource.

    Keep holding the lock while the data is copied.  While at it, change the
    return value to a more informative value.  It is disregarded by the
    callers.

    [akpm@linux-foundation.org: fix find_next_iomem_res() documentation]
    Link: http://lkml.kernel.org/r/20190613045903.4922-2-namit@vmware.com
    Fixes: ff3cc952d3f00 ("resource: Add remove_resource interface")
    Signed-off-by: Nadav Amit <namit@vmware.com>
    Reviewed-by: Andrew Morton <akpm@linux-foundation.org>
    Reviewed-by: Dan Williams <dan.j.williams@intel.com>
    Cc: Borislav Petkov <bp@suse.de>
    Cc: Toshi Kani <toshi.kani@hpe.com>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Cc: Dave Hansen <dave.hansen@linux.intel.com>
    Cc: Bjorn Helgaas <bhelgaas@google.com>
    Cc: Ingo Molnar <mingo@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 866981b4d498cf2de8f2c6334fc71c1dbb3bef7c
Author: Drew Davenport <ddavenport@chromium.org>
Date:   Tue Jul 16 16:30:18 2019 -0700

    include/asm-generic/bug.h: fix "cut here" for WARN_ON for __WARN_TAINT architectures

    commit 6b15f678fb7d5ef54e089e6ace72f007fe6e9895 upstream.

    For architectures using __WARN_TAINT, the WARN_ON macro did not print
    out the "cut here" string.  The other WARN_XXX macros would print "cut
    here" inside __warn_printk, which is not called for WARN_ON since it
    doesn't have a message to print.

    Link: http://lkml.kernel.org/r/20190624154831.163888-1-ddavenport@chromium.org
    Fixes: a7bed27af194 ("bug: fix "cut here" location for __WARN_TAINT architectures")
    Signed-off-by: Drew Davenport <ddavenport@chromium.org>
    Acked-by: Kees Cook <keescook@chromium.org>
    Tested-by: Kees Cook <keescook@chromium.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 c8ab9aafa60a92e95e148cd3b5af91843492fcd1
Author: Jan Harkes <jaharkes@cs.cmu.edu>
Date:   Tue Jul 16 16:28:04 2019 -0700

    coda: pass the host file in vma->vm_file on mmap

    commit 7fa0a1da3dadfd9216df7745a1331fdaa0940d1c upstream.

    Patch series "Coda updates".

    The following patch series is a collection of various fixes for Coda,
    most of which were collected from linux-fsdevel or linux-kernel but
    which have as yet not found their way upstream.

    This patch (of 22):

    Various file systems expect that vma->vm_file points at their own file
```

```
    handle, several use file_inode(vma->vm_file) to get at their inode or
    use vma->vm_file->private_data.  However the way Coda wrapped mmap on a
    host file broke this assumption, vm_file was still pointing at the Coda
    file and the host file systems would scribble over Coda's inode and
    private file data.

    This patch fixes the incorrect expectation and wraps vm_ops->open and
    vm_ops->close to allow Coda to track when the vm_area_struct is
    destroyed so we still release the reference on the Coda file handle at
    the right time.

    Link:
http://lkml.kernel.org/r/0e850c6e59c0b147dc2dcd51a3af004c948c3697.1558117389.git.jaharkes@cs
.cmu.edu
    Signed-off-by: Jan Harkes <jaharkes@cs.cmu.edu>
    Cc: Arnd Bergmann <arnd@arndb.de>
    Cc: Colin Ian King <colin.king@canonical.com>
    Cc: Dan Carpenter <dan.carpenter@oracle.com>
    Cc: David Howells <dhowells@redhat.com>
    Cc: Fabian Frederick <fabf@skynet.be>
    Cc: Mikko Rapeli <mikko.rapeli@iki.fi>
    Cc: Sam Protsenko <semen.protsenko@linaro.org>
    Cc: Yann Droneaud <ydroneaud@opteya.com>
    Cc: Zhouyang Jia <jiazhouyang09@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 6d468c7832f4bc1d3821e21810f92f11bbcc1add
Author: Henry Burns <henryburns@google.com>
Date:   Thu Jul 11 20:52:14 2019 -0700

    mm/z3fold.c: lock z3fold page before __SetPageMovable()

    commit 810481a246089117d98e3373a3cb735c3efc1f90 upstream.

    Following zsmalloc.c's example we call trylock_page() and unlock_page().
    Also make z3fold_page_migrate() assert that newpage is passed in locked,
    as per the documentation.

    [akpm@linux-foundation.org: fix trylock_page return value test, per Shakeel]
    Link: http://lkml.kernel.org/r/20190702005122.41036-1-henryburns@google.com
    Link: http://lkml.kernel.org/r/20190702233538.52793-1-henryburns@google.com
    Signed-off-by: Henry Burns <henryburns@google.com>
    Suggested-by: Vitaly Wool <vitalywool@gmail.com>
    Acked-by: Vitaly Wool <vitalywool@gmail.com>
    Acked-by: David Rientjes <rientjes@google.com>
    Reviewed-by: Shakeel Butt <shakeelb@google.com>
    Cc: Vitaly Vul <vitaly.vul@sony.com>
    Cc: Mike Rapoport <rppt@linux.vnet.ibm.com>
    Cc: Xidong Wang <wangxidong_97@163.com>
    Cc: Jonathan Adams <jwadams@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 83822dc2deb76399b82c09dc6ec995fe45646026
Author: Yafang Shao <laoar.shao@gmail.com>
Date:   Thu Jul 11 20:52:11 2019 -0700

    mm/memcontrol: fix wrong statistics in memory.stat

    commit dd9239900e12db84c198855b262ae7796db1123b upstream.
```

```
    When we calculate total statistics for memcg1_stats and memcg1_events,
    we use the the index 'i' in the for loop as the events index.  Actually
    we should use memcg1_stats[i] and memcg1_events[i] as the events index.

    Link: http://lkml.kernel.org/r/1562116978-19539-1-git-send-email-laoar.shao@gmail.com
    Fixes: 42a300353577 ("mm: memcontrol: fix recursive statistics correctness &
scalabilty").
    Signed-off-by: Yafang Shao <laoar.shao@gmail.com
    Reviewed-by: Shakeel Butt <shakeelb@google.com>
    Cc: Michal Hocko <mhocko@suse.com>
    Cc: Johannes Weiner <hannes@cmpxchg.org>
    Cc: Yafang Shao <shaoyafang@didiglobal.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 b60f290bc06836fbba6fefa3ca58dbfdbefda5b3
Author: Dan Williams <dan.j.williams@intel.com>
Date:   Thu Jul 18 15:58:36 2019 -0700

    libnvdimm/pfn: fix fsdax-mode namespace info-block zero-fields

    commit 7e3e888dfc138089f4c15a81b418e88f0978f744 upstream.

    At namespace creation time there is the potential for the "expected to
    be zero" fields of a 'pfn' info-block to be filled with indeterminate
    data.  While the kernel buffer is zeroed on allocation it is immediately
    overwritten by nd_pfn_validate() filling it with the current contents of
    the on-media info-block location.  For fields like, 'flags' and the
    'padding' it potentially means that future implementations can not rely on
    those fields being zero.

    In preparation to stop using the 'start_pad' and 'end_trunc' fields for
    section alignment, arrange for fields that are not explicitly
    initialized to be guaranteed zero.  Bump the minor version to indicate
    it is safe to assume the 'padding' and 'flags' are zero.  Otherwise,
    this corruption is expected to benign since all other critical fields
    are explicitly initialized.

    Note The cc: stable is about spreading this new policy to as many
    kernels as possible not fixing an issue in those kernels.  It is not
    until the change titled "libnvdimm/pfn: Stop padding pmem namespaces to
    section alignment" where this improper initialization becomes a problem.
    So if someone decides to backport "libnvdimm/pfn: Stop padding pmem
    namespaces to section alignment" (which is not tagged for stable), make
    sure this pre-requisite is flagged.

    Link: http://lkml.kernel.org/r/156092356065.979959.6681003754765958296.stgit@dwillia2-
desk3.amr.corp.intel.com
    Fixes: 32ab0a3f5170 ("libnvdimm, pmem: 'struct page' for pmem")
    Signed-off-by: Dan Williams <dan.j.williams@intel.com>
    Tested-by: Aneesh Kumar K.V <aneesh.kumar@linux.ibm.com>        [ppc64]
    Cc: <stable@vger.kernel.org>
    Cc: David Hildenbrand <david@redhat.com>
    Cc: Jane Chu <jane.chu@oracle.com>
    Cc: Jeff Moyer <jmoyer@redhat.com>
    Cc: Jérôme Glisse <jglisse@redhat.com>
    Cc: Jonathan Corbet <corbet@lwn.net>
    Cc: Logan Gunthorpe <logang@deltatee.com>
    Cc: Michal Hocko <mhocko@suse.com>
    Cc: Mike Rapoport <rppt@linux.ibm.com>
    Cc: Oscar Salvador <osalvador@suse.de>
    Cc: Pavel Tatashin <pasha.tatashin@soleen.com>
```

```
    Cc: Toshi Kani <toshi.kani@hpe.com>
    Cc: Vlastimil Babka <vbabka@suse.cz>
    Cc: Wei Yang <richardw.yang@linux.intel.com>
    Cc: Jason Gunthorpe <jgg@mellanox.com>
    Cc: Christoph Hellwig <hch@lst.de>
    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 8a7501921486b7783dbce46a215ce188d7a1c81d
Author: Aneesh Kumar K.V <aneesh.kumar@linux.ibm.com>
Date:   Thu Jul 11 20:52:08 2019 -0700

    mm/nvdimm: add is_ioremap_addr and use that to check ioremap address

    commit 9bd3bb6703d8c0a5fb8aec8e3287bd55b7341dcd upstream.

    Architectures like powerpc use different address range to map ioremap
    and vmalloc range.  The memunmap() check used by the nvdimm layer was
    wrongly using is_vmalloc_addr() to check for ioremap range which fails
    for ppc64.  This result in ppc64 not freeing the ioremap mapping.  The
    side effect of this is an unbind failure during module unload with
    papr_scm nvdimm driver

    Link: http://lkml.kernel.org/r/20190701134038.14165-1-aneesh.kumar@linux.ibm.com
    Signed-off-by: Aneesh Kumar K.V <aneesh.kumar@linux.ibm.com>
    Fixes: b5beae5e224f ("powerpc/pseries: Add driver for PAPR SCM regions")
    Cc: Dan Williams <dan.j.williams@intel.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 cc01bc548eff975ce88d12917b15cbd1f66de26e
Author: Kuo-Hsin Yang <vovoy@chromium.org>
Date:   Thu Jul 11 20:52:04 2019 -0700

    mm: vmscan: scan anonymous pages on file refaults

    commit 2c012a4ad1a2cd3fb5a0f9307b9d219f84eda1fa upstream.

    When file refaults are detected and there are many inactive file pages,
    the system never reclaim anonymous pages, the file pages are dropped
    aggressively when there are still a lot of cold anonymous pages and
    system thrashes.  This issue impacts the performance of applications
    with large executable, e.g.  chrome.

    With this patch, when file refault is detected, inactive_list_is_low()
    always returns true for file pages in get_scan_count() to enable
    scanning anonymous pages.

    The problem can be reproduced by the following test program.

    ---8<---
    void fallocate_file(const char *filename, off_t size)
    {
            struct stat st;
            int fd;

            if (!stat(filename, &st) && st.st_size >= size)
                    return;

            fd = open(filename, O_WRONLY | O_CREAT, 0600);
            if (fd < 0) {
                    perror("create file");
```

```
                exit(1);
        }
        if (posix_fallocate(fd, 0, size)) {
                perror("fallocate");
                exit(1);
        }
        close(fd);
}

long *alloc_anon(long size)
{
        long *start = malloc(size);
        memset(start, 1, size);
        return start;
}

long access_file(const char *filename, long size, long rounds)
{
        int fd, i;
        volatile char *start1, *end1, *start2;
        const int page_size = getpagesize();
        long sum = 0;

        fd = open(filename, O_RDONLY);
        if (fd == -1) {
                perror("open");
                exit(1);
        }

        /*
         * Some applications, e.g. chrome, use a lot of executable file
         * pages, map some of the pages with PROT_EXEC flag to simulate
         * the behavior.
         */
        start1 = mmap(NULL, size / 2, PROT_READ | PROT_EXEC, MAP_SHARED,
                      fd, 0);
        if (start1 == MAP_FAILED) {
                perror("mmap");
                exit(1);
        }
        end1 = start1 + size / 2;

        start2 = mmap(NULL, size / 2, PROT_READ, MAP_SHARED, fd, size / 2);
        if (start2 == MAP_FAILED) {
                perror("mmap");
                exit(1);
        }

        for (i = 0; i < rounds; ++i) {
                struct timeval before, after;
                volatile char *ptr1 = start1, *ptr2 = start2;
                gettimeofday(&before, NULL);
                for (; ptr1 < end1; ptr1 += page_size, ptr2 += page_size)
                        sum += *ptr1 + *ptr2;
                gettimeofday(&after, NULL);
                printf("File access time, round %d: %f (sec)
", i,
                       (after.tv_sec - before.tv_sec) +
                       (after.tv_usec - before.tv_usec) / 1000000.0);
        }
        return sum;
}

int main(int argc, char *argv[])
{
```

```
        const long MB = 1024 * 1024;
        long anon_mb, file_mb, file_rounds;
        const char filename[] = "large";
        long *ret1;
        long ret2;

        if (argc != 4) {
                printf("usage: thrash ANON_MB FILE_MB FILE_ROUNDS
");
                exit(0);
        }
        anon_mb = atoi(argv[1]);
        file_mb = atoi(argv[2]);
        file_rounds = atoi(argv[3]);

        fallocate_file(filename, file_mb * MB);
        printf("Allocate %ld MB anonymous pages
", anon_mb);
        ret1 = alloc_anon(anon_mb * MB);
        printf("Access %ld MB file pages
", file_mb);
        ret2 = access_file(filename, file_mb * MB, file_rounds);
        printf("Print result to prevent optimization: %ld
",
               *ret1 + ret2);
        return 0;
}
---8<---
```

Running the test program on 2GB RAM VM with kernel 5.2.0-rc5, the program fills ram with 2048 MB memory, access a 200 MB file for 10 times. Without this patch, the file cache is dropped aggresively and every access to the file is from disk.

```
$ ./thrash 2048 200 10
Allocate 2048 MB anonymous pages
Access 200 MB file pages
File access time, round 0: 2.489316 (sec)
File access time, round 1: 2.581277 (sec)
File access time, round 2: 2.487624 (sec)
File access time, round 3: 2.449100 (sec)
File access time, round 4: 2.420423 (sec)
File access time, round 5: 2.343411 (sec)
File access time, round 6: 2.454833 (sec)
File access time, round 7: 2.483398 (sec)
File access time, round 8: 2.572701 (sec)
File access time, round 9: 2.493014 (sec)
```

With this patch, these file pages can be cached.

```
$ ./thrash 2048 200 10
Allocate 2048 MB anonymous pages
Access 200 MB file pages
File access time, round 0: 2.475189 (sec)
File access time, round 1: 2.440777 (sec)
File access time, round 2: 2.411671 (sec)
File access time, round 3: 1.955267 (sec)
File access time, round 4: 0.029924 (sec)
File access time, round 5: 0.000808 (sec)
File access time, round 6: 0.000771 (sec)
File access time, round 7: 0.000746 (sec)
File access time, round 8: 0.000738 (sec)
File access time, round 9: 0.000747 (sec)
```

Checked the swap out stats during the test [1], 19006 pages swapped out

```
    with this patch, 3418 pages swapped out without this patch. There are
    more swap out, but I think it's within reasonable range when file backed
    data set doesn't fit into the memory.

    $ ./thrash 2000 100 2100 5 1 # ANON_MB FILE_EXEC FILE_NOEXEC ROUNDS
    PROCESSES Allocate 2000 MB anonymous pages active_anon: 1613644,
    inactive_anon: 348656, active_file: 892, inactive_file: 1384 (kB)
    pswpout: 7972443, pgpgin: 478615246 Access 100 MB executable file pages
    Access 2100 MB regular file pages File access time, round 0: 12.165,
    (sec) active_anon: 1433788, inactive_anon: 478116, active_file: 17896,
    inactive_file: 24328 (kB) File access time, round 1: 11.493, (sec)
    active_anon: 1430576, inactive_anon: 477144, active_file: 25440,
    inactive_file: 26172 (kB) File access time, round 2: 11.455, (sec)
    active_anon: 1427436, inactive_anon: 476060, active_file: 21112,
    inactive_file: 28808 (kB) File access time, round 3: 11.454, (sec)
    active_anon: 1420444, inactive_anon: 473632, active_file: 23216,
    inactive_file: 35036 (kB) File access time, round 4: 11.479, (sec)
    active_anon: 1413964, inactive_anon: 471460, active_file: 31728,
    inactive_file: 32224 (kB) pswpout: 7991449 (+ 19006), pgpgin: 489924366
    (+ 11309120)

    With 4 processes accessing non-overlapping parts of a large file, 30316
    pages swapped out with this patch, 5152 pages swapped out without this
    patch.  The swapout number is small comparing to pgpgin.

    [1]: https://github.com/vovo/testing/blob/master/mem_thrash.c

    Link: http://lkml.kernel.org/r/20190701081038.GA83398@google.com
    Fixes: e9868505987a ("mm,vmscan: only evict file pages when we have plenty")
    Fixes: 7c5bd705d8f9 ("mm: memcg: only evict file pages when we have plenty")
    Signed-off-by: Kuo-Hsin Yang <vovoy@chromium.org>
    Acked-by: Johannes Weiner <hannes@cmpxchg.org>
    Cc: Michal Hocko <mhocko@suse.com>
    Cc: Sonny Rao <sonnyrao@chromium.org>
    Cc: Mel Gorman <mgorman@techsingularity.net>
    Cc: Rik van Riel <riel@redhat.com>
    Cc: Vladimir Davydov <vdavydov.dev@gmail.com>
    Cc: Minchan Kim <minchan@kernel.org>
    Cc: <stable@vger.kernel.org>    [4.12+]
    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 dd3239bb3b08e86ff8d3e4698d9f5dbf33c488d8
Author: Aaron Armstrong Skomra <skomra@gmail.com>
Date:   Fri May 10 15:34:18 2019 -0700

    HID: wacom: correct touch resolution x/y typo

    commit 68c20cc2164cc5c7c73f8012ae6491afdb1f7f72 upstream.

    This affects the 2nd-gen Intuos Pro Medium and Large
    when using their Bluetooth connection.

    Fixes: 4922cd26f03c ("HID: wacom: Support 2nd-gen Intuos Pro's Bluetooth classic
interface")
    Cc: <stable@vger.kernel.org> # v4.11+
    Signed-off-by: Aaron Armstrong Skomra <aaron.skomra@wacom.com>
    Reviewed-by: Jason Gerecke <jason.gerecke@wacom.com>
    Signed-off-by: Jiri Kosina <jkosina@suse.cz>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f43f71434c005dc30c3203979191b972a01282f4
Author: Aaron Armstrong Skomra <skomra@gmail.com>
Date:   Fri May 10 15:34:17 2019 -0700
```

```
    HID: wacom: generic: Correct pad syncing

    commit d4b8efeb46d99a5d02e7f88ac4eaccbe49370770 upstream.

    Only sync the pad once per report, not once per collection.
    Also avoid syncing the pad on battery reports.

    Fixes: f8b6a74719b5 ("HID: wacom: generic: Support multiple tools per report")
    Cc: <stable@vger.kernel.org> # v4.17+
    Signed-off-by: Aaron Armstrong Skomra <aaron.skomra@wacom.com>
    Signed-off-by: Jiri Kosina <jkosina@suse.cz>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ccd466996707b47e561cbcf08b88934c6db9636f
Author: Aaron Armstrong Skomra <skomra@gmail.com>
Date:   Fri May 10 15:31:16 2019 -0700

    HID: wacom: generic: only switch the mode on devices with LEDs

    commit d8e9806005f28bbb49899dab2068e3359e22ba35 upstream.

    Currently, the driver will attempt to set the mode on all
    devices with a center button, but some devices with a center
    button lack LEDs, and attempting to set the LEDs on devices
    without LEDs results in the kernel error message of the form:

    "leds input8::wacom-0.1: Setting an LED's brightness failed (-32)"

    This is because the generic codepath erroneously assumes that the
    BUTTON_CENTER usage indicates that the device has LEDs, the
    previously ignored TOUCH_RING_SETTING usage is a more accurate
    indication of the existence of LEDs on the device.

    Fixes: 10c55cacb8b2 ("HID: wacom: generic: support LEDs")
    Cc: <stable@vger.kernel.org> # v4.11+
    Signed-off-by: Aaron Armstrong Skomra <aaron.skomra@wacom.com>
    Reviewed-by: Jason Gerecke <jason.gerecke@wacom.com>
    Signed-off-by: Jiri Kosina <jkosina@suse.cz>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 44ecea8be480c2e01633feee5f95dd00357050b2
Author: Danit Goldberg <danitg@mellanox.com>
Date:   Fri Jul 5 19:21:57 2019 +0300

    IB/mlx5: Report correctly tag matching rendezvous capability

    commit 89705e92700170888236555fe91b45e4c1bb0985 upstream.

    Userspace expects the IB_TM_CAP_RC bit to indicate that the device
    supports RC transport tag matching with rendezvous offload. However the
    firmware splits this into two capabilities for eager and rendezvous tag
    matching.

    Only if the FW supports both modes should userspace be told the tag
    matching capability is available.

    Cc: <stable@vger.kernel.org> # 4.13
    Fixes: eb761894351d ("IB/mlx5: Fill XRQ capabilities")
    Signed-off-by: Danit Goldberg <danitg@mellanox.com>
    Reviewed-by: Yishai Hadas <yishaih@mellanox.com>
    Reviewed-by: Artemy Kovalyov <artemyko@mellanox.com>
    Signed-off-by: Leon Romanovsky <leonro@mellanox.com>
    Signed-off-by: Jason Gunthorpe <jgg@mellanox.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit b6a0d03dd4ea4733aaaf0a7e5c4beaaf8ac10306
Author: Filipe Manana <fdmanana@suse.com>
Date:   Wed Jun 19 13:05:50 2019 +0100

    Btrfs: add missing inode version, ctime and mtime updates when punching hole

    commit 179006688a7e888cbff39577189f2e034786d06a upstream.

    If the range for which we are punching a hole covers only part of a page,
    we end up updating the inode item but we skip the update of the inode's
    iversion, mtime and ctime. Fix that by ensuring we update those properties
    of the inode.

    A patch for fstests test case generic/059 that tests this as been sent
    along with this fix.

    Fixes: 2aaa66558172b0 ("Btrfs: add hole punching")
    Fixes: e8c1c76e804b18 ("Btrfs: add missing inode update when punching hole")
    CC: stable@vger.kernel.org # 4.4+
    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 24ae804edf6d76bb7c6c7d64c5a331c8c482f7c4
Author: Filipe Manana <fdmanana@suse.com>
Date:   Wed Jun 19 13:05:39 2019 +0100

    Btrfs: fix fsync not persisting dentry deletions due to inode evictions

    commit 803f0f64d17769071d7287d9e3e3b79a3e1ae937 upstream.

    In order to avoid searches on a log tree when unlinking an inode, we check
    if the inode being unlinked was logged in the current transaction, as well
    as the inode of its parent directory. When any of the inodes are logged,
    we proceed to delete directory items and inode reference items from the
    log, to ensure that if a subsequent fsync of only the inode being unlinked
    or only of the parent directory when the other is not fsync'ed as well,
    does not result in the entry still existing after a power failure.

    That check however is not reliable when one of the inodes involved (the
    one being unlinked or its parent directory's inode) is evicted, since the
    logged_trans field is transient, that is, it is not stored on disk, so it
    is lost when the inode is evicted and loaded into memory again (which is
    set to zero on load). As a consequence the checks currently being done by
    btrfs_del_dir_entries_in_log() and btrfs_del_inode_ref_in_log() always
    return true if the inode was evicted before, regardless of the inode
    having been logged or not before (and in the current transaction), this
    results in the dentry being unlinked still existing after a log replay
    if after the unlink operation only one of the inodes involved is fsync'ed.

    Example:

      $ mkfs.btrfs -f /dev/sdb
      $ mount /dev/sdb /mnt

      $ mkdir /mnt/dir
      $ touch /mnt/dir/foo
      $ xfs_io -c fsync /mnt/dir/foo

      # Keep an open file descriptor on our directory while we evict inodes.
      # We just want to evict the file's inode, the directory's inode must not
      # be evicted.
      $ ( cd /mnt/dir; while true; do :; done ) &
      $ pid=$!
```

```
       # Wait a bit to give time to background process to chdir to our test
       # directory.
       $ sleep 0.5

       # Trigger eviction of the file's inode.
       $ echo 2 > /proc/sys/vm/drop_caches

       # Unlink our file and fsync the parent directory. After a power failure
       # we don't expect to see the file anymore, since we fsync'ed the parent
       # directory.
       $ rm -f $SCRATCH_MNT/dir/foo
       $ xfs_io -c fsync /mnt/dir

       <power failure>

       $ mount /dev/sdb /mnt
       $ ls /mnt/dir
       foo
       $
        --> file still there, unlink not persisted despite explicit fsync on dir
```

Fix this by checking if the inode has the full_sync bit set in its runtime flags as well, since that bit is set everytime an inode is loaded from disk, or for other less common cases such as after a shrinking truncate or failure to allocate extent maps for holes, and gets cleared after the first fsync. Also consider the inode as possibly logged only if it was last modified in the current transaction (besides having the full_fsync flag set).

Fixes: 3a5f1d458ad161 ("Btrfs: Optimize btree walking while logging inodes")
CC: stable@vger.kernel.org # 4.4+
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 12e58c1b445e48f8b21740407242bbe422be42ee
Author: Filipe Manana <fdmanana@suse.com>
Date: Fri Jun 7 11:25:24 2019 +0100

Btrfs: fix data loss after inode eviction, renaming it, and fsync it

commit d1d832a0b51dd9570429bb4b81b2a6c1759e681a upstream.

When we log an inode, regardless of logging it completely or only that it exists, we always update it as logged (logged_trans and last_log_commit fields of the inode are updated). This is generally fine and avoids future attempts to log it from having to do repeated work that brings no value.

However, if we write data to a file, then evict its inode after all the dealloc was flushed (and ordered extents completed), rename the file and fsync it, we end up not logging the new extents, since the rename may result in logging that the inode exists in case the parent directory was logged before. The following reproducer shows and explains how this can happen:

```
       $ mkfs.btrfs -f /dev/sdb
       $ mount /dev/sdb /mnt

       $ mkdir /mnt/dir
       $ touch /mnt/dir/foo
       $ touch /mnt/dir/bar

       # Do a direct IO write instead of a buffered write because with a
       # buffered write we would need to make sure dealloc gets flushed and
```

```
       # complete before we do the inode eviction later, and we can not do that
       # from user space with call to things such as sync(2) since that results
       # in a transaction commit as well.
       $ xfs_io -d -c "pwrite -S 0xd3 0 4K" /mnt/dir/bar

       # Keep the directory dir in use while we evict inodes. We want our file
       # bar's inode to be evicted but we don't want our directory's inode to
       # be evicted (if it were evicted too, we would not be able to reproduce
       # the issue since the first fsync below, of file foo, would result in a
       # transaction commit.
       $ ( cd /mnt/dir; while true; do :; done ) &
       $ pid=$!

       # Wait a bit to give time for the background process to chdir.
       $ sleep 0.1

       # Evict all inodes, except the inode for the directory dir because it is
       # currently in use by our background process.
       $ echo 2 > /proc/sys/vm/drop_caches

       # fsync file foo, which ends up persisting information about the parent
       # directory because it is a new inode.
       $ xfs_io -c fsync /mnt/dir/foo

       # Rename bar, this results in logging that this inode exists (inode item,
       # names, xattrs) because the parent directory is in the log.
       $ mv /mnt/dir/bar /mnt/dir/baz

       # Now fsync baz, which ends up doing absolutely nothing because of the
       # rename operation which logged that the inode exists only.
       $ xfs_io -c fsync /mnt/dir/baz

       <power failure>

       $ mount /dev/sdb /mnt
       $ od -t x1 -A d /mnt/dir/baz
       0000000

         --> Empty file, data we wrote is missing.

    Fix this by not updating last_sub_trans of an inode when we are logging
    only that it exists and the inode was not yet logged since it was loaded
    from disk (full_sync bit set), this is enough to make btrfs_inode_in_log()
    return false for this scenario and make us log the inode. The logged_trans
    of the inode is still always setsince that alone is used to track if names
    need to be deleted as part of unlink operations.

    Fixes: 257c62e1bce03e ("Btrfs: avoid tree log commit when there are no changes")
    CC: stable@vger.kernel.org # 4.4+
    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 ced88769e67d95e5e7748135ec5ff19df721921d
Author: Johannes Thumshirn <jthumshirn@suse.de>
Date:   Thu Jun 6 12:07:15 2019 +0200

    btrfs: correctly validate compression type

    commit aa53e3bfac7205fb3a8815ac1c937fd6ed01b41e upstream.

    Nikolay reported the following KASAN splat when running btrfs/048:

    [ 1843.470920] ==================================================================
    [ 1843.471971] BUG: KASAN: slab-out-of-bounds in strncmp+0x66/0xb0
```

```
    [ 1843.472775] Read of size 1 at addr ffff888111e369e2 by task btrfs/3979

    [ 1843.473904] CPU: 3 PID: 3979 Comm: btrfs Not tainted 5.2.0-rc3-default #536
    [ 1843.475009] Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS 1.10.2-
1ubuntu1 04/01/2014
    [ 1843.476322] Call Trace:
    [ 1843.476674]  dump_stack+0x7c/0xbb
    [ 1843.477132]  ? strncmp+0x66/0xb0
    [ 1843.477587]  print_address_description+0x114/0x320
    [ 1843.478256]  ? strncmp+0x66/0xb0
    [ 1843.478740]  ? strncmp+0x66/0xb0
    [ 1843.479185]  __kasan_report+0x14e/0x192
    [ 1843.479759]  ? strncmp+0x66/0xb0
    [ 1843.480209]  kasan_report+0xe/0x20
    [ 1843.480679]  strncmp+0x66/0xb0
    [ 1843.481105]  prop_compression_validate+0x24/0x70
    [ 1843.481798]  btrfs_xattr_handler_set_prop+0x65/0x160
    [ 1843.482509]  __vfs_setxattr+0x71/0x90
    [ 1843.483012]  __vfs_setxattr_noperm+0x84/0x130
    [ 1843.483606]  vfs_setxattr+0xac/0xb0
    [ 1843.484085]  setxattr+0x18c/0x230
    [ 1843.484546]  ? vfs_setxattr+0xb0/0xb0
    [ 1843.485048]  ? __mod_node_page_state+0x1f/0xa0
    [ 1843.485672]  ? _raw_spin_unlock+0x24/0x40
    [ 1843.486233]  ? __handle_mm_fault+0x988/0x1290
    [ 1843.486823]  ? lock_acquire+0xb4/0x1e0
    [ 1843.487330]  ? lock_acquire+0xb4/0x1e0
    [ 1843.487842]  ? mnt_want_write_file+0x3c/0x80
    [ 1843.488442]  ? debug_lockdep_rcu_enabled+0x22/0x40
    [ 1843.489089]  ? rcu_sync_lockdep_assert+0xe/0x70
    [ 1843.489707]  ? __sb_start_write+0x158/0x200
    [ 1843.490278]  ? mnt_want_write_file+0x3c/0x80
    [ 1843.490855]  ? __mnt_want_write+0x98/0xe0
    [ 1843.491397]  __x64_sys_fsetxattr+0xba/0xe0
    [ 1843.492201]  ? trace_hardirqs_off_thunk+0x1a/0x1c
    [ 1843.493201]  do_syscall_64+0x6c/0x230
    [ 1843.493988]  entry_SYSCALL_64_after_hwframe+0x49/0xbe
    [ 1843.495041] RIP: 0033:0x7fa7a8a7707a
    [ 1843.495819] Code: 48 8b 0d 21 de 2b 00 f7 d8 64 89 01 48 83 c8 ff c3 66 2e 0f 1f 84
00 00 00 00 00 0f 1f 44 00 00 49 89 ca b8 be 00 00 00 0f 05 <48> 3d 01 f0 ff ff 73 01 c3 48
8b 0d ee dd 2b 00 f7 d8 64 89 01 48
    [ 1843.499203] RSP: 002b:00007ffcb73bca38 EFLAGS: 00000202 ORIG_RAX: 00000000000000be
    [ 1843.500210] RAX: ffffffffffffffda RBX: 00007ffcb73bda9d RCX: 00007fa7a8a7707a
    [ 1843.501170] RDX: 00007ffcb73bda9d RSI: 00000000006dc050 RDI: 0000000000000003
    [ 1843.502152] RBP: 00000000006dc050 R08: 0000000000000000 R09: 0000000000000000
    [ 1843.503109] R10: 0000000000000002 R11: 0000000000000202 R12: 00007ffcb73bda91
    [ 1843.504055] R13: 0000000000000003 R14: 00007ffcb73bda82 R15: ffffffffffffffff

    [ 1843.505268] Allocated by task 3979:
    [ 1843.505771]  save_stack+0x19/0x80
    [ 1843.506211]  __kasan_kmalloc.constprop.5+0xa0/0xd0
    [ 1843.506836]  setxattr+0xeb/0x230
    [ 1843.507264]  __x64_sys_fsetxattr+0xba/0xe0
    [ 1843.507886]  do_syscall_64+0x6c/0x230
    [ 1843.508429]  entry_SYSCALL_64_after_hwframe+0x49/0xbe

    [ 1843.509558] Freed by task 0:
    [ 1843.510188] (stack is not available)

    [ 1843.511309] The buggy address belongs to the object at ffff888111e369e0
                    which belongs to the cache kmalloc-8 of size 8
    [ 1843.514095] The buggy address is located 2 bytes inside of
                    8-byte region [ffff888111e369e0, ffff888111e369e8)
    [ 1843.516524] The buggy address belongs to the page:
    [ 1843.517561] page:ffff88813f478d80 refcount:1 mapcount:0 mapping:ffff88811940c300
```

```
  index:0xffff888111e373b8 compound_mapcount: 0
      [ 1843.519993] flags: 0x4404000010200(slab|head)
      [ 1843.520951] raw: 0004404000010200 ffff88813f48b008 ffff888119403d50 ffff88811940c300
      [ 1843.522616] raw: ffff888111e373b8 000000000016000f 00000001ffffffff 0000000000000000
      [ 1843.524281] page dumped because: kasan: bad access detected

      [ 1843.525936] Memory state around the buggy address:
      [ 1843.526975]  ffff888111e36880: fc fc fc fc fc fc fc fc fc fc fc fc fc fc fc fc
      [ 1843.528479]  ffff888111e36900: fc fc fc fc fc fc fc fc fc fc fc fc fc fc fc fc
      [ 1843.530138] >ffff888111e36980: fc fc fc fc fc fc fc fc fc fc fc fc 02 fc fc fc
      [ 1843.531877]                                                             ^
      [ 1843.533287]  ffff888111e36a00: fc fc fc fc fc fc fc fc fc fc fc fc fc fc fc fc
      [ 1843.534874]  ffff888111e36a80: fc fc fc fc fc fc fc fc fc fc fc fc fc fc fc fc
      [ 1843.536468] ==================================================================

      This is caused by supplying a too short compression value ('lz') in the
      test-case and comparing it to 'lzo' with strncmp() and a length of 3.
      strncmp() read past the 'lz' when looking for the 'o' and thus caused an
      out-of-bounds read.

      Introduce a new check 'btrfs_compress_is_valid_type()' which not only
      checks the user-supplied value against known compression types, but also
      employs checks for too short values.

      Reported-by: Nikolay Borisov <nborisov@suse.com>
      Fixes: 272e5326c783 ("btrfs: prop: fix vanished compression property after failed set")
      CC: stable@vger.kernel.org # 5.1+
      Reviewed-by: Nikolay Borisov <nborisov@suse.com>
      Signed-off-by: Johannes Thumshirn <jthumshirn@suse.de>
      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 af2efb64664f8d02864a198719ee8bd44622d733
  Author: Niklas Cassel <niklas.cassel@linaro.org>
  Date:   Wed May 29 11:43:52 2019 +0200

      PCI: qcom: Ensure that PERST is asserted for at least 100 ms

      commit 64adde31c8e996a6db6f7a1a4131180e363aa9f2 upstream.

      Currently, there is only a 1 ms sleep after asserting PERST.

      Reading the datasheets for different endpoints, some require PERST to be
      asserted for 10 ms in order for the endpoint to perform a reset, others
      require it to be asserted for 50 ms.

      Several SoCs using this driver uses PCIe Mini Card, where we don't know
      what endpoint will be plugged in.

      The PCI Express Card Electromechanical Specification r2.0, section
      2.2, "PERST# Signal" specifies:

      "On power up, the deassertion of PERST# is delayed 100 ms (TPVPERL) from
      the power rails achieving specified operating limits."

      Add a sleep of 100 ms before deasserting PERST, in order to ensure that
      we are compliant with the spec.

      Fixes: 82a823833f4e ("PCI: qcom: Add Qualcomm PCIe controller driver")
      Signed-off-by: Niklas Cassel <niklas.cassel@linaro.org>
      Signed-off-by: Lorenzo Pieralisi <lorenzo.pieralisi@arm.com>
      Acked-by: Stanimir Varbanov <svarbanov@mm-sol.com>
      Cc: stable@vger.kernel.org # 4.5+
      Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit 3530d286c105e0da0986f2ae236a64f83df67eda
Author: Mika Westerberg <mika.westerberg@linux.intel.com>
Date:   Wed Jun 12 13:57:39 2019 +0300

    PCI: Do not poll for PME if the device is in D3cold

    commit 000dd5316e1c756a1c028f22e01d06a38249dd4d upstream.

    PME polling does not take into account that a device that is directly
    connected to the host bridge may go into D3cold as well. This leads to a
    situation where the PME poll thread reads from a config space of a
    device that is in D3cold and gets incorrect information because the
    config space is not accessible.

    Here is an example from Intel Ice Lake system where two PCIe root ports
    are in D3cold (I've instrumented the kernel to log the PMCSR register
    contents):

      [   62.971442] pcieport 0000:00:07.1: Check PME status, PMCSR=0xffff
      [   62.971504] pcieport 0000:00:07.0: Check PME status, PMCSR=0xffff

    Since 0xffff is interpreted so that PME is pending, the root ports will
    be runtime resumed. This repeats over and over again essentially
    blocking all runtime power management.

    Prevent this from happening by checking whether the device is in D3cold
    before its PME status is read.

    Fixes: 71a83bd727cc ("PCI/PM: add runtime PM support to PCIe port")
    Signed-off-by: Mika Westerberg <mika.westerberg@linux.intel.com>
    Reviewed-by: Lukas Wunner <lukas@wunner.de>
    Cc: 3.6+ <stable@vger.kernel.org> # v3.6+
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 45c91514e61c63c9ebcd91b0d5fbc60c01dc2a54
Author: Dexuan Cui <decui@microsoft.com>
Date:   Fri Jun 21 23:45:23 2019 +0000

    PCI: hv: Fix a use-after-free bug in hv_eject_device_work()

    commit 4df591b20b80cb77920953812d894db259d85bd7 upstream.

    Fix a use-after-free in hv_eject_device_work().

    Fixes: 05f151a73ec2 ("PCI: hv: Fix a memory leak in hv_eject_device_work()")
    Signed-off-by: Dexuan Cui <decui@microsoft.com>
    Signed-off-by: Lorenzo Pieralisi <lorenzo.pieralisi@arm.com>
    Reviewed-by: Michael Kelley <mikelley@microsoft.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 48f48d3e692229cf7f6542cb8a186079c6228998
Author: Alexander Shishkin <alexander.shishkin@linux.intel.com>
Date:   Fri Jun 21 19:19:30 2019 +0300

    intel_th: pci: Add Ice Lake NNPI support

    commit 4aa5aed2b6f267592705a526f57518a5d715b769 upstream.

    This adds Ice Lake NNPI support to the Intel(R) Trace Hub.

    Signed-off-by: Alexander Shishkin <alexander.shishkin@linux.intel.com>
    Reviewed-by: Andy Shevchenko <andriy.shevchenko@linux.intel.com>
```

```
    Cc: stable <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20190621161930.60785-5-
alexander.shishkin@linux.intel.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 45d2dde9580621043be2e747f254967768a76425
Author: Jason Gunthorpe <jgg@mellanox.com>
Date:   Tue Jun 11 13:09:51 2019 -0300

    RDMA/odp: Fix missed unlock in non-blocking invalidate_start

    commit 7608bf40cf2480057ec0da31456cc428791c32ef upstream.

    If invalidate_start returns with EAGAIN then the umem_rwsem needs to be
    unlocked as no invalidate_end will be called.

    Cc: <stable@vger.kernel.org>
    Fixes: ca748c39ea3f ("RDMA/umem: Get rid of per_mm->notifier_count")
    Signed-off-by: Jason Gunthorpe <jgg@mellanox.com>
    Reviewed-by: Leon Romanovsky <leonro@mellanox.com>
    Signed-off-by: Doug Ledford <dledford@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 6b0c68827b72b67e1435ce45fa12cc056cf18ffd
Author: Bart Van Assche <bvanassche@acm.org>
Date:   Wed May 29 09:38:31 2019 -0700

    RDMA/srp: Accept again source addresses that do not have a port number

    commit bcef5b7215681250c4bf8961dfe15e9e4fef97d0 upstream.

    The function srp_parse_in() is used both for parsing source address
    specifications and for target address specifications. Target addresses
    must have a port number. Having to specify a port number for source
    addresses is inconvenient. Make sure that srp_parse_in() supports again
    parsing addresses with no port number.

    Cc: <stable@vger.kernel.org>
    Fixes: c62adb7def71 ("IB/srp: Fix IPv6 address parsing")
    Signed-off-by: Bart Van Assche <bvanassche@acm.org>
    Signed-off-by: Jason Gunthorpe <jgg@mellanox.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b80e370c0cbd49de7b391afd51c4cc335b0aa689
Author: Damien Le Moal <damien.lemoal@wdc.com>
Date:   Wed Jul 10 13:53:10 2019 +0900

    block: Fix potential overflow in blk_report_zones()

    commit 113ab72ed4794c193509a97d7c6d32a6886e1682 upstream.

    For large values of the number of zones reported and/or large zone
    sizes, the sector increment calculated with

    blk_queue_zone_sectors(q) * n

    in blk_report_zones() loop can overflow the unsigned int type used for
    the calculation as both "n" and blk_queue_zone_sectors() value are
    unsigned int. E.g. for a device with 256 MB zones (524288 sectors),
    overflow happens with 8192 or more zones reported.

    Changing the return type of blk_queue_zone_sectors() to sector_t, fixes
    this problem and avoids overflow problem for all other callers of this
    helper too. The same change is also applied to the bdev_zone_sectors()
    helper.
```

```
    Fixes: e76239a3748c ("block: add a report_zones method")
    Cc: stable@vger.kernel.org
    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 72ac05ce265155709de803683f326407f4ae046b
Author: Damien Le Moal <damien.lemoal@wdc.com>
Date:   Mon Jul 1 14:09:15 2019 +0900

    block: Allow mapping of vmalloc-ed buffers

    commit b4c5875d36178e8df409bdce232f270cac89fafe upstream.

    To allow the SCSI subsystem scsi_execute_req() function to issue
    requests using large buffers that are better allocated with vmalloc()
    rather than kmalloc(), modify bio_map_kern() to allow passing a buffer
    allocated with vmalloc().

    To do so, detect vmalloc-ed buffers using is_vmalloc_addr(). For
    vmalloc-ed buffers, flush the buffer using flush_kernel_vmap_range(),
    use vmalloc_to_page() instead of virt_to_page() to obtain the pages of
    the buffer, and invalidate the buffer addresses with
    invalidate_kernel_vmap_range() on completion of read BIOs. This last
    point is executed using the function bio_invalidate_vmalloc_pages()
    which is defined only if the architecture defines
    ARCH_HAS_FLUSH_KERNEL_DCACHE_PAGE, that is, if the architecture
    actually needs the invalidation done.

    Fixes: 515ce6061312 ("scsi: sd_zbc: Fix sd_zbc_report_zones() buffer allocation")
    Fixes: e76239a3748c ("block: add a report_zones method")
    Cc: stable@vger.kernel.org
    Reviewed-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Damien Le Moal <damien.lemoal@wdc.com>
    Reviewed-by: Christoph Hellwig <hch@lst.de>
    Reviewed-by: Chaitanya Kulkarni <chaitanya.kulkarni@wdc.com>
    Reviewed-by: Ming Lei <ming.lei@redhat.com>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 0a71cb0fb4ef792705eab23ad2e8e4c1a740edf1
Author: Andres Rodriguez <andresx7@gmail.com>
Date:   Wed Jun 19 14:09:01 2019 -0400

    drm/edid: parse CEA blocks embedded in DisplayID

    commit e28ad544f462231d3fd081a7316339359efbb481 upstream.

    DisplayID blocks allow embedding of CEA blocks. The payloads are
    identical to traditional top level CEA extension blocks, but the header
    is slightly different.

    This change allows the CEA parser to find a CEA block inside a DisplayID
    block. Additionally, it adds support for parsing the embedded CTA
    header. No further changes are necessary due to payload parity.

    This change fixes audio support for the Valve Index HMD.

    Signed-off-by: Andres Rodriguez <andresx7@gmail.com>
    Reviewed-by: Dave Airlie <airlied@redhat.com>
    Cc: Jani Nikula <jani.nikula@linux.intel.com>
    Cc: <stable@vger.kernel.org> # v4.15
    Signed-off-by: Dave Airlie <airlied@redhat.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20190619180901.17901-1-
```

```
andresx7@gmail.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 4f216eb095ac5e116d002bd0c81f832773541edf
Author: Eiichi Tsukata <devel@etsukata.com>
Date:   Thu Jul 11 11:35:01 2019 +0900

    x86/stacktrace: Prevent infinite loop in arch_stack_walk_user()

    commit cbf5b73d162b22e044fe0b7d51dcaa33be065253 upstream.

    arch_stack_walk_user() checks `if (fp == frame.next_fp)` to prevent a
    infinite loop by self reference but it's not enogh for circular reference.

    Once a lack of return address is found, there is no point to continue the
    loop, so break out.

    Fixes: 02b67518e2b1 ("tracing: add support for userspace stacktraces in
tracing/iter_ctrl")
    Signed-off-by: Eiichi Tsukata <devel@etsukata.com>
    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Acked-by: Linus Torvalds <torvalds@linux-foundation.org>
    Link: https://lkml.kernel.org/r/20190711023501.963-1-devel@etsukata.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e00f7c63433a64322cd2cbcb0517c3ed0809d0e2
Author: Kim Phillips <kim.phillips@amd.com>
Date:   Fri Jun 28 21:59:33 2019 +0000

    perf/x86/amd/uncore: Set the thread mask for F17h L3 PMCs

    commit 2f217d58a8a086d3399fecce39fb358848e799c4 upstream.

    Fill in the L3 performance event select register ThreadMask
    bitfield, to enable per hardware thread accounting.

    Signed-off-by: Kim Phillips <kim.phillips@amd.com>
    Signed-off-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Cc: <stable@vger.kernel.org>
    Cc: Alexander Shishkin <alexander.shishkin@linux.intel.com>
    Cc: Arnaldo Carvalho de Melo <acme@redhat.com>
    Cc: Borislav Petkov <bp@alien8.de>
    Cc: Gary Hook <Gary.Hook@amd.com>
    Cc: H. Peter Anvin <hpa@zytor.com>
    Cc: Janakarajan Natarajan <Janakarajan.Natarajan@amd.com>
    Cc: Jiri Olsa <jolsa@redhat.com>
    Cc: Linus Torvalds <torvalds@linux-foundation.org>
    Cc: Martin Liska <mliska@suse.cz>
    Cc: Namhyung Kim <namhyung@kernel.org>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Cc: Pu Wen <puwen@hygon.cn>
    Cc: Stephane Eranian <eranian@google.com>
    Cc: Suravee Suthikulpanit <Suravee.Suthikulpanit@amd.com>
    Cc: Thomas Gleixner <tglx@linutronix.de>
    Cc: Vince Weaver <vincent.weaver@maine.edu>
    Link: https://lkml.kernel.org/r/20190628215906.4276-2-kim.phillips@amd.com
    Signed-off-by: Ingo Molnar <mingo@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b86794662315a908655569acb4a32e2ce5c16907
Author: Kim Phillips <kim.phillips@amd.com>
Date:   Fri Jun 28 21:59:20 2019 +0000

    perf/x86/amd/uncore: Do not set 'ThreadMask' and 'SliceMask' for non-L3 PMCs
```

```
    commit 16f4641166b10e199f0d7b68c2c5f004fef0bda3 upstream.

    The following commit:

      d7cbbe49a930 ("perf/x86/amd/uncore: Set ThreadMask and SliceMask for L3 Cache perf
events")

    enables L3 PMC events for all threads and slices by writing 1's in
    'ChL3PmcCfg' (L3 PMC PERF_CTL) register fields.

    Those bitfields overlap with high order event select bits in the Data
    Fabric PMC control register, however.

    So when a user requests raw Data Fabric events (-e amd_df/event=0xYYY/),
    the two highest order bits get inadvertently set, changing the counter
    select to events that don't exist, and for which no counts are read.

    This patch changes the logic to write the L3 masks only when dealing
    with L3 PMC counters.

    AMD Family 16h and below Northbridge (NB) counters were not affected.

    Signed-off-by: Kim Phillips <kim.phillips@amd.com>
    Signed-off-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Cc: <stable@vger.kernel.org>
    Cc: Alexander Shishkin <alexander.shishkin@linux.intel.com>
    Cc: Arnaldo Carvalho de Melo <acme@redhat.com>
    Cc: Borislav Petkov <bp@alien8.de>
    Cc: Gary Hook <Gary.Hook@amd.com>
    Cc: H. Peter Anvin <hpa@zytor.com>
    Cc: Janakarajan Natarajan <Janakarajan.Natarajan@amd.com>
    Cc: Jiri Olsa <jolsa@redhat.com>
    Cc: Linus Torvalds <torvalds@linux-foundation.org>
    Cc: Martin Liska <mliska@suse.cz>
    Cc: Namhyung Kim <namhyung@kernel.org>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Cc: Pu Wen <puwen@hygon.cn>
    Cc: Stephane Eranian <eranian@google.com>
    Cc: Suravee Suthikulpanit <Suravee.Suthikulpanit@amd.com>
    Cc: Thomas Gleixner <tglx@linutronix.de>
    Cc: Vince Weaver <vincent.weaver@maine.edu>
    Fixes: d7cbbe49a930 ("perf/x86/amd/uncore: Set ThreadMask and SliceMask for L3 Cache
perf events")
    Link: https://lkml.kernel.org/r/20190628215906.4276-1-kim.phillips@amd.com
    Signed-off-by: Ingo Molnar <mingo@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e078b94f21818c8f8181cc426bd26cc542bc7481
Author: Kan Liang <kan.liang@linux.intel.com>
Date:   Tue Jun 25 07:21:35 2019 -0700

    perf/x86/intel: Fix spurious NMI on fixed counter

    commit e4557c1a46b0d32746bd309e1941914b5a6912b4 upstream.

    If a user first sample a PEBS event on a fixed counter, then sample a
    non-PEBS event on the same fixed counter on Icelake, it will trigger
    spurious NMI. For example:

      perf record -e 'cycles:p' -a
      perf record -e 'cycles' -a

    The error message for spurious NMI:

      [June 21 15:38] Uhhuh. NMI received for unknown reason 30 on CPU 2.
```

```
       [    +0.000000] Do you have a strange power saving mode enabled?
       [    +0.000000] Dazed and confused, but trying to continue

    The bug was introduced by the following commit:

      commit 6f55967ad9d9 ("perf/x86/intel: Fix race in intel_pmu_disable_event()")

    The commit moves the intel_pmu_pebs_disable() after intel_pmu_disable_fixed(),
    which returns immediately.  The related bit of PEBS_ENABLE MSR will never be
    cleared for the fixed counter. Then a non-PEBS event runs on the fixed counter,
    but the bit on PEBS_ENABLE is still set, which triggers spurious NMIs.

    Check and disable PEBS for fixed counters after intel_pmu_disable_fixed().

    Reported-by: Yi, Ammy <ammy.yi@intel.com>
    Signed-off-by: Kan Liang <kan.liang@linux.intel.com>
    Signed-off-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Acked-by: Jiri Olsa <jolsa@kernel.org>
    Cc: <stable@vger.kernel.org>
    Cc: Alexander Shishkin <alexander.shishkin@linux.intel.com>
    Cc: Arnaldo Carvalho de Melo <acme@redhat.com>
    Cc: Jiri Olsa <jolsa@redhat.com>
    Cc: Linus Torvalds <torvalds@linux-foundation.org>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Cc: Stephane Eranian <eranian@google.com>
    Cc: Thomas Gleixner <tglx@linutronix.de>
    Cc: Vince Weaver <vincent.weaver@maine.edu>
    Fixes: 6f55967ad9d9 ("perf/x86/intel: Fix race in intel_pmu_disable_event()")
    Link: https://lkml.kernel.org/r/20190625142135.22112-1-kan.liang@linux.intel.com
    Signed-off-by: Ingo Molnar <mingo@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 9a0d1e1fc893ffd5399d39f2ac043de0286ad6ac
Author: David Rientjes <rientjes@google.com>
Date:   Tue Jul 9 19:44:03 2019 -0700

    x86/boot: Fix memory leak in default_get_smp_config()

    commit e74bd96989dd42a51a73eddb4a5510a6f5e42ac3 upstream.

    When default_get_smp_config() is called with early == 1 and mpf->feature1
    is non-zero, mpf is leaked because the return path does not do
    early_memunmap().

    Fix this and share a common exit routine.

    Fixes: 5997efb96756 ("x86/boot: Use memremap() to map the MPF and MPC data")
    Reported-by: Cfir Cohen <cfir@google.com>
    Signed-off-by: David Rientjes <rientjes@google.com>
    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Cc: stable@vger.kernel.org
    Link:
https://lkml.kernel.org/r/alpine.DEB.2.21.1907091942570.28240@chino.kir.corp.google.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a717a5bdf41d05a59e66cbf7bd94cbeb6700d217
Author: Dexuan Cui <decui@microsoft.com>
Date:   Fri Jul 19 03:22:35 2019 +0000

    x86/hyper-v: Zero out the VP ASSIST PAGE on allocation

    commit e320ab3cec7dd8b1606964d81ae1e14391ff8e96 upstream.

    The VP ASSIST PAGE is an "overlay" page (see Hyper-V TLFS's Section
    5.2.1 "GPA Overlay Pages" for the details) and here is an excerpt:
```

```
    "The hypervisor defines several special pages that "overlay" the guest's
     Guest Physical Addresses (GPA) space. Overlays are addressed GPA but are
     not included in the normal GPA map maintained internally by the hypervisor.
     Conceptually, they exist in a separate map that overlays the GPA map.

     If a page within the GPA space is overlaid, any SPA page mapped to the
     GPA page is effectively "obscured" and generally unreachable by the
     virtual processor through processor memory accesses.

     If an overlay page is disabled, the underlying GPA page is "uncovered",
     and an existing mapping becomes accessible to the guest."

    SPA = System Physical Address = the final real physical address.

    When a CPU (e.g. CPU1) is onlined, hv_cpu_init() allocates the VP ASSIST
    PAGE and enables the EOI optimization for this CPU by writing the MSR
    HV_X64_MSR_VP_ASSIST_PAGE. From now on, hvp->apic_assist belongs to the
    special SPA page, and this CPU *always* uses hvp->apic_assist (which is
    shared with the hypervisor) to decide if it needs to write the EOI MSR.

    When a CPU is offlined then on the outgoing CPU:
    1. hv_cpu_die() disables the EOI optimizaton for this CPU, and from
       now on hvp->apic_assist belongs to the original "normal" SPA page;
    2. the remaining work of stopping this CPU is done
    3. this CPU is completely stopped.

    Between 1 and 3, this CPU can still receive interrupts (e.g. reschedule
    IPIs from CPU0, and Local APIC timer interrupts), and this CPU *must* write
    the EOI MSR for every interrupt received, otherwise the hypervisor may not
    deliver further interrupts, which may be needed to completely stop the CPU.

    So, after the EOI optimization is disabled in hv_cpu_die(), it's required
    that the hvp->apic_assist's bit0 is zero, which is not guaranteed by the
    current allocation mode because it lacks __GFP_ZERO. As a consequence the
    bit might be set and interrupt handling would not write the EOI MSR causing
    interrupt delivery to become stuck.

    Add the missing __GFP_ZERO to the allocation.

    Note 1: after the "normal" SPA page is allocted and zeroed out, neither the
    hypervisor nor the guest writes into the page, so the page remains with
    zeros.

    Note 2: see Section 10.3.5 "EOI Assist" for the details of the EOI
    optimization. When the optimization is enabled, the guest can still write
    the EOI MSR register irrespective of the "No EOI required" value, but
    that's slower than the optimized assist based variant.

    Fixes: ba696429d290 ("x86/hyper-v: Implement EOI assist")
    Signed-off-by: Dexuan Cui <decui@microsoft.com>
    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Cc: stable@vger.kernel.org
    Link: https://lkml.kernel.org/r/
<PU1P153MB0169B716A637FABF07433C04BFCB0@PU1P153MB0169.APCP153.PROD.OUTLOOK.COM
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 26f111d5b4c1e0bbc66f116f24d528e6a2c6bed5
Author: Soeren Moch <smoch@web.de>
Date:   Mon Jul 1 12:53:13 2019 +0200

    rt2x00usb: fix rx queue hang

    commit 41a531ffa4c5aeb062f892227c00fabb3b4a9c91 upstream.
```

```
    Since commit ed194d136769 ("usb: core: remove local_irq_save() around
     ->complete() handler") the handler rt2x00usb_interrupt_rxdone() is
    not running with interrupts disabled anymore. So this completion handler
    is not guaranteed to run completely before workqueue processing starts
    for the same queue entry.
    Be sure to set all other flags in the entry correctly before marking
    this entry ready for workqueue processing. This way we cannot miss error
    conditions that need to be signalled from the completion handler to the
    worker thread.
    Note that rt2x00usb_work_rxdone() processes all available entries, not
    only such for which queue_work() was called.

    This patch is similar to what commit df71c9cfceea ("rt2x00: fix order
    of entry flags modification") did for TX processing.

    This fixes a regression on a RT5370 based wifi stick in AP mode, which
    suddenly stopped data transmission after some period of heavy load. Also
    stopping the hanging hostapd resulted in the error message "ieee80211
    phy0: rt2x00queue_flush_queue: Warning - Queue 14 failed to flush".
    Other operation modes are probably affected as well, this just was
    the used testcase.

    Fixes: ed194d136769 ("usb: core: remove local_irq_save() around ->complete() handler")
    Cc: stable@vger.kernel.org # 4.20+
    Signed-off-by: Soeren Moch <smoch@web.de>
    Acked-by: Stanislaw Gruszka <sgruszka@redhat.com>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1e635a53f3988d5d43fb5c7f113c370b9b12e441
Author: YueHaibing <yuehaibing@huawei.com>
Date:   Tue Apr 30 19:59:42 2019 +0800

    9p/virtio: Add cleanup path in p9_virtio_init

    commit d4548543fc4ece56c6f04b8586f435fb4fd84c20 upstream.

    KASAN report this:

    BUG: unable to handle kernel paging request at ffffffffa0097000
    PGD 3870067 P4D 3870067 PUD 3871063 PMD 2326e2067 PTE 0
    Oops: 0000 [#1
    CPU: 0 PID: 5340 Comm: modprobe Not tainted 5.1.0-rc7+ #25
    Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS rel-1.9.3-0-ge2fc41e-
prebuilt.qemu-project.org 04/01/2014
    RIP: 0010:__list_add_valid+0x10/0x70
    Code: c3 48 8b 06 55 48 89 e5 5d 48 39 07 0f 94 c0 0f b6 c0 c3 90 90 90 90 90 90 90 55
48 89 d0 48 8b 52 08 48 89 e5 48 39 f2 75 19 <48> 8b 32 48 39 f0 75 3a

    RSP: 0018:ffffc90000e23c68 EFLAGS: 00010246
    RAX: ffffffffa00ad000 RBX: ffffffffa009d000 RCX: 0000000000000000
    RDX: ffffffffa0097000 RSI: ffffffffa0097000 RDI: ffffffffa009d000
    RBP: ffffc90000e23c68 R08: 0000000000000001 R09: 0000000000000000
    R10: 0000000000000000 R11: 0000000000000000 R12: ffffffffa0097000
    R13: ffff888231797180 R14: 0000000000000000 R15: ffffc90000e23e78
    FS:  00007fb215285540(0000) GS:ffff888237a00000(0000) knlGS:0000000000000000
    CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    CR2: ffffffffa0097000 CR3: 000000022f144000 CR4: 00000000000006f0
    Call Trace:
     v9fs_register_trans+0x2f/0x60 [9pnet
     ? 0xffffffffa0087000
     p9_virtio_init+0x25/0x1000 [9pnet_virtio
     do_one_initcall+0x6c/0x3cc
     ? kmem_cache_alloc_trace+0x248/0x3b0
     do_init_module+0x5b/0x1f1
```

```
      load_module+0x1db1/0x2690
      ? m_show+0x1d0/0x1d0
      __do_sys_finit_module+0xc5/0xd0
      __x64_sys_finit_module+0x15/0x20
      do_syscall_64+0x6b/0x1d0
      entry_SYSCALL_64_after_hwframe+0x49/0xbe
     RIP: 0033:0x7fb214d8e839
     Code: 00 f3 c3 66 2e 0f 1f 84 00 00 00 00 00 0f 1f 40 00 48 89 f8 48 89 f7 48 89 d6 48
 89 ca 4d 89 c2 4d 89 c8 4c 8b 4c 24 08 0f 05 <48> 3d 01 f0 ff ff 73 01

     RSP: 002b:00007ffc96554278 EFLAGS: 00000246 ORIG_RAX: 0000000000000139
     RAX: ffffffffffffffda RBX: 000055e67eed2aa0 RCX: 00007fb214d8e839
     RDX: 0000000000000000 RSI: 000055e67ce95c2e RDI: 0000000000000003
     RBP: 000055e67ce95c2e R08: 0000000000000000 R09: 000055e67eed2aa0
     R10: 0000000000000003 R11: 0000000000000246 R12: 0000000000000000
     R13: 000055e67eeda500 R14: 0000000000040000 R15: 000055e67eed2aa0
     Modules linked in: 9pnet_virtio(+) 9pnet gre rfkill vmw_vsock_virtio_transport_common
 vsock [last unloaded: 9pnet_virtio
     CR2: ffffffffa0097000
     ---[ end trace 4a52bb13ff07b761

     If register_virtio_driver() fails in p9_virtio_init,
     we should call v9fs_unregister_trans() to do cleanup.

     Link: http://lkml.kernel.org/r/20190430115942.41840-1-yuehaibing@huawei.com
     Cc: stable@vger.kernel.org
     Reported-by: Hulk Robot <hulkci@huawei.com>
     Fixes: b530cc794024 ("9p: add virtio transport")
     Signed-off-by: YueHaibing <yuehaibing@huawei.com>
     Signed-off-by: Dominique Martinet <dominique.martinet@cea.fr>
     Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

 commit 13e6ba7257bb453e9f4678ee7046ce5ce6511a71
 Author: YueHaibing <yuehaibing@huawei.com>
 Date:   Tue Apr 30 22:39:33 2019 +0800

     9p/xen: Add cleanup path in p9_trans_xen_init

     commit 80a316ff16276b36d0392a8f8b2f63259857ae98 upstream.

     If xenbus_register_frontend() fails in p9_trans_xen_init,
     we should call v9fs_unregister_trans() to do cleanup.

     Link: http://lkml.kernel.org/r/20190430143933.19368-1-yuehaibing@huawei.com
     Cc: stable@vger.kernel.org
     Fixes: 868eb122739a ("xen/9pfs: introduce Xen 9pfs transport driver")
     Signed-off-by: YueHaibing <yuehaibing@huawei.com>
     Signed-off-by: Dominique Martinet <dominique.martinet@cea.fr>
     Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

 commit bd98e000f632f4275f3c29710646c23a363704d3
 Author: Juergen Gross <jgross@suse.com>
 Date:   Fri Jun 21 20:47:03 2019 +0200

     xen/events: fix binding user event channels to cpus

     commit bce5963bcb4f9934faa52be323994511d59fd13c upstream.

     When binding an interdomain event channel to a vcpu via
     IOCTL_EVTCHN_BIND_INTERDOMAIN not only the event channel needs to be
     bound, but the affinity of the associated IRQi must be changed, too.
     Otherwise the IRQ and the event channel won't be moved to another vcpu
     in case the original vcpu they were bound to is going offline.

     Cc: <stable@vger.kernel.org> # 4.13
```

```
    Fixes: c48f64ab472389df ("xen-evtchn: Bind dyn evtchn:qemu-dm interrupt to next online
VCPU")
    Signed-off-by: Juergen Gross <jgross@suse.com>
    Reviewed-by: Boris Ostrovsky <boris.ostrovsky@oracle.com>
    Signed-off-by: Juergen Gross <jgross@suse.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 8ef6682c1ad6e59357dcddd507244b227ff775e0
Author: Damien Le Moal <damien.lemoal@wdc.com>
Date:   Tue Jul 16 14:39:34 2019 +0900

    dm zoned: fix zone state management race

    commit 3b8cafdd5436f9298b3bf6eb831df5eef5ee82b6 upstream.

    dm-zoned uses the zone flag DMZ_ACTIVE to indicate that a zone of the
    backend device is being actively read or written and so cannot be
    reclaimed. This flag is set as long as the zone atomic reference
    counter is not 0. When this atomic is decremented and reaches 0 (e.g.
    on BIO completion), the active flag is cleared and set again whenever
    the zone is reused and BIO issued with the atomic counter incremented.
    These 2 operations (atomic inc/dec and flag set/clear) are however not
    always executed atomically under the target metadata mutex lock and
    this causes the warning:

    WARN_ON(!test_bit(DMZ_ACTIVE, &zone->flags));

    in dmz_deactivate_zone() to be displayed. This problem is regularly
    triggered with xfstests generic/209, generic/300, generic/451 and
    xfs/077 with XFS being used as the file system on the dm-zoned target
    device. Similarly, xfstests ext4/303, ext4/304, generic/209 and
    generic/300 trigger the warning with ext4 use.

    This problem can be easily fixed by simply removing the DMZ_ACTIVE flag
    and managing the "ACTIVE" state by directly looking at the reference
    counter value. To do so, the functions dmz_activate_zone() and
    dmz_deactivate_zone() are changed to inline functions respectively
    calling atomic_inc() and atomic_dec(), while the dmz_is_active() macro
    is changed to an inline function calling atomic_read().

    Fixes: 3b1a94c88b79 ("dm zoned: drive-managed zoned block device target")
    Cc: stable@vger.kernel.org
    Reported-by: Masato Suzuki <masato.suzuki@wdc.com>
    Signed-off-by: Damien Le Moal <damien.lemoal@wdc.com>
    Signed-off-by: Mike Snitzer <snitzer@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 90b50060fe158af6ef85e7c9bb7d4a9911553492
Author: Daniel Jordan <daniel.m.jordan@oracle.com>
Date:   Tue Jul 16 12:32:53 2019 -0400

    padata: use smp_mb in padata_reorder to avoid orphaned padata jobs

    commit cf144f81a99d1a3928f90b0936accfd3f45c9a0a upstream.

    Testing padata with the tcrypt module on a 5.2 kernel...

        # modprobe tcrypt alg="pcrypt(rfc4106(gcm(aes)))" type=3
        # modprobe tcrypt mode=211 sec=1

    ...produces this splat:

        INFO: task modprobe:10075 blocked for more than 120 seconds.
              Not tainted 5.2.0-base+ #16
        modprobe        D    0 10075  10064 0x80004080
```

```
     Call Trace:
      ? __schedule+0x4dd/0x610
      ? ring_buffer_unlock_commit+0x23/0x100
      schedule+0x6c/0x90
      schedule_timeout+0x3b/0x320
      ? trace_buffer_unlock_commit_regs+0x4f/0x1f0
      wait_for_common+0x160/0x1a0
      ? wake_up_q+0x80/0x80
      { crypto_wait_req }             # entries in braces added by hand
      { do_one_aead_op }
      { test_aead_jiffies }
      test_aead_speed.constprop.17+0x681/0xf30 [tcrypt]
      do_test+0x4053/0x6a2b [tcrypt]
      ? 0xffffffffa00f4000
      tcrypt_mod_init+0x50/0x1000 [tcrypt]
      ...
```

The second modprobe command never finishes because in padata_reorder, CPU0's load of reorder_objects is executed before the unlocking store in spin_unlock_bh(pd->lock), causing CPU0 to miss CPU1's increment:

```
CPU0                                  CPU1

padata_reorder                        padata_do_serial
  LOAD reorder_objects  // 0
                                        INC reorder_objects  // 1
                                        padata_reorder
                                          TRYLOCK pd->lock   // failed
  UNLOCK pd->lock
```

CPU0 deletes the timer before returning from padata_reorder and since no other job is submitted to padata, modprobe waits indefinitely.

Add a pair of full barriers to guarantee proper ordering:

```
CPU0                                  CPU1

padata_reorder                        padata_do_serial
  UNLOCK pd->lock
  smp_mb()
  LOAD reorder_objects
                                        INC reorder_objects
                                        smp_mb__after_atomic()
                                        padata_reorder
                                          TRYLOCK pd->lock
```

smp_mb__after_atomic is needed so the read part of the trylock operation comes after the INC, as Andrea points out.   Thanks also to Andrea for help with writing a litmus test.

Fixes: 16295bec6398 ("padata: Generic parallelization/serialization interface")
Signed-off-by: Daniel Jordan <daniel.m.jordan@oracle.com>
Cc: <stable@vger.kernel.org>
Cc: Andrea Parri <andrea.parri@amarulasolutions.com>
Cc: Boqun Feng <boqun.feng@gmail.com>
Cc: Herbert Xu <herbert@gondor.apana.org.au>
Cc: Paul E. McKenney <paulmck@linux.ibm.com>
Cc: Peter Zijlstra <peterz@infradead.org>
Cc: Steffen Klassert <steffen.klassert@secunet.com>
Cc: linux-arch@vger.kernel.org
Cc: linux-crypto@vger.kernel.org
Cc: linux-kernel@vger.kernel.org
Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

```
commit 082d7e3acce2ef5f50f54f2a11da6fdee3747dd3
Author: Lyude Paul <lyude@redhat.com>
Date:   Wed Jun 26 14:10:27 2019 -0400

    drm/nouveau/i2c: Enable i2c pads & busses during preinit

    commit 7cb95eeea6706c790571042a06782e378b2561ea upstream.

    It turns out that while disabling i2c bus access from software when the
    GPU is suspended was a step in the right direction with:

    commit 342406e4fbba ("drm/nouveau/i2c: Disable i2c bus access after
    ->fini()")

    We also ended up accidentally breaking the vbios init scripts on some
    older Tesla GPUs, as apparently said scripts can actually use the i2c
    bus. Since these scripts are executed before initializing any
    subdevices, we end up failing to acquire access to the i2c bus which has
    left a number of cards with their fan controllers uninitialized. Luckily
    this doesn't break hardware - it just means the fan gets stuck at 100%.

    This also means that we've always been using our i2c busses before
    initializing them during the init scripts for older GPUs, we just didn't
    notice it until we started preventing them from being used until init.
    It's pretty impressive this never caused us any issues before!

    So, fix this by initializing our i2c pad and busses during subdev
    pre-init. We skip initializing aux busses during pre-init, as those are
    guaranteed to only ever be used by nouveau for DP aux transactions.

    Signed-off-by: Lyude Paul <lyude@redhat.com>
    Tested-by: Marc Meledandri <m.meledandri@gmail.com>
    Fixes: 342406e4fbba ("drm/nouveau/i2c: Disable i2c bus access after ->fini()")
    Cc: stable@vger.kernel.org
    Signed-off-by: Ben Skeggs <bskeggs@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 237997e7ad8c2db3adeb47ba1b3e16efbc62aa69
Author: Linus Walleij <linus.walleij@linaro.org>
Date:   Mon Jul 15 22:21:01 2019 +0200

    ARM: dts: gemini: Set DIR-685 SPI CS as active low

    commit f90b8fda3a9d72a9422ea80ae95843697f94ea4a upstream.

    The SPI to the display on the DIR-685 is active low, we were
    just saved by the SPI library enforcing active low on everything
    before, so set it as active low to avoid ambiguity.

    Link: https://lore.kernel.org/r/20190715202101.16060-1-linus.walleij@linaro.org
    Cc: stable@vger.kernel.org
    Signed-off-by: Linus Walleij <linus.walleij@linaro.org>
    Signed-off-by: Olof Johansson <olof@lixom.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 76e1c730e6d4fb22b6b73e95aa766d880f6cb363
Author: Vitor Soares <Vitor.Soares@synopsys.com>
Date:   Wed Jun 19 20:36:31 2019 +0200

    i3c: fix i2c and i3c scl rate by bus mode

    commit ecc8fb54bd443bf69996d9d5ddb8d90a50f14936 upstream.

    Currently the I3C framework limits SCL frequency to FM speed when
    dealing with a mixed slow bus, even if all I2C devices are FM+ capable.
```

```
    The core was also not accounting for I3C speed limitations when
    operating in mixed slow mode and was erroneously using FM+ speed as the
    max I2C speed when operating in mixed fast mode.

    Fixes: 3a379bbcea0a ("i3c: Add core I3C infrastructure")
    Signed-off-by: Vitor Soares <vitor.soares@synopsys.com>
    Cc: Boris Brezillon <bbrezillon@kernel.org>
    Cc: <stable@vger.kernel.org>
    Cc: <linux-kernel@vger.kernel.org>
    Signed-off-by: Boris Brezillon <boris.brezillon@collabora.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 14cc90952cef94bfa89a6b4a2f55fd9a70f50a16
Author: Radoslaw Burny <rburny@google.com>
Date:   Tue Jul 16 16:26:51 2019 -0700

    fs/proc/proc_sysctl.c: fix the default values of i_uid/i_gid on /proc/sys inodes.

    commit 5ec27ec735ba0477d48c80561cc5e856f0c5dfaf upstream.

    Normally, the inode's i_uid/i_gid are translated relative to s_user_ns,
    but this is not a correct behavior for proc.  Since sysctl permission
    check in test_perm is done against GLOBAL_ROOT_[UG]ID, it makes more
    sense to use these values in u_[ug]id of proc inodes.  In other words:
    although uid/gid in the inode is not read during test_perm, the inode
    logically belongs to the root of the namespace.  I have confirmed this
    with Eric Biederman at LPC and in this thread:
      https://lore.kernel.org/lkml/87k1kzjdff.fsf@xmission.com

    Consequences
    ============

    Since the i_[ug]id values of proc nodes are not used for permissions
    checks, this change usually makes no functional difference.  However, it
    causes an issue in a setup where:

     * a namespace container is created without root user in container -
       hence the i_[ug]id of proc nodes are set to INVALID_[UG]ID

     * container creator tries to configure it by writing /proc/sys files,
       e.g. writing /proc/sys/kernel/shmmax to configure shared memory limit

    Kernel does not allow to open an inode for writing if its i_[ug]id are
    invalid, making it impossible to write shmmax and thus - configure the
    container.

    Using a container with no root mapping is apparently rare, but we do use
    this configuration at Google.  Also, we use a generic tool to configure
    the container limits, and the inability to write any of them causes a
    failure.

    History
    =======

    The invalid uids/gids in inodes first appeared due to 81754357770e (fs:
    Update i_[ug]id_(read|write) to translate relative to s_user_ns).
    However, AFAIK, this did not immediately cause any issues.  The
    inability to write to these "invalid" inodes was only caused by a later
    commit 0bd23d09b874 (vfs: Don't modify inodes with a uid or gid unknown
    to the vfs).

    Tested: Used a repro program that creates a user namespace without any
    mapping and stat'ed /proc/$PID/root/proc/sys/kernel/shmmax from outside.
    Before the change, it shows the overflow uid, with the change it's 0.
```

```
    The overflow uid indicates that the uid in the inode is not correct and
    thus it is not possible to open the file for writing.

    Link: http://lkml.kernel.org/r/20190708115130.250149-1-rburny@google.com
    Fixes: 0bd23d09b874 ("vfs: Don't modify inodes with a uid or gid unknown to the vfs")
    Signed-off-by: Radoslaw Burny <rburny@google.com>
    Acked-by: Luis Chamberlain <mcgrof@kernel.org>
    Cc: Kees Cook <keescook@chromium.org>
    Cc: "Eric W . Biederman" <ebiederm@xmission.com>
    Cc: Seth Forshee <seth.forshee@canonical.com>
    Cc: John Sperbeck <jsperbeck@google.com>
    Cc: Alexey Dobriyan <adobriyan@gmail.com>
    Cc: <stable@vger.kernel.org>    [4.8+]
    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 2f0e9eed11ca778d0ec5507a739c28472fd4ca20
Author: Eric W. Biederman <ebiederm@xmission.com>
Date:   Wed May 15 22:54:56 2019 -0500

    signal: Correct namespace fixups of si_pid and si_uid

    commit 7a0cf094944e2540758b7f957eb6846d5126f535 upstream.

    The function send_signal was split from __send_signal so that it would
    be possible to bypass the namespace logic based upon current[1].  As it
    turns out the si_pid and the si_uid fixup are both inappropriate in
    the case of kill_pid_usb_asyncio so move that logic into send_signal.

    It is difficult to arrange but possible for a signal with an si_code
    of SI_TIMER or SI_SIGIO to be sent across namespace boundaries.  In
    which case tests for when it is ok to change si_pid and si_uid based
    on SI_FROMUSER are incorrect.  Replace the use of SI_FROMUSER with a
    new test has_si_pid_and_used based on siginfo_layout.

    Now that the uid fixup is no longer present after expanding
    SEND_SIG_NOINFO properly calculate the si_uid that the target
    task needs to read.

    [1] 7978b567d315 ("signals: add from_ancestor_ns parameter to send_signal()")
    Cc: stable@vger.kernel.org
    Fixes: 6588c1e3ff01 ("signals: SI_USER: Masquerade si_pid when crossing pid ns
boundary")
    Fixes: 6b550f949594 ("user namespace: make signal.c respect user namespaces")
    Signed-off-by: "Eric W. Biederman" <ebiederm@xmission.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1852befc685e413a7f86a4fc2b8666e7533b6660
Author: Eric W. Biederman <ebiederm@xmission.com>
Date:   Thu Feb 7 19:44:12 2019 -0600

    signal/usb: Replace kill_pid_info_as_cred with kill_pid_usb_asyncio

    commit 70f1b0d34bdf03065fe869e93cc17cad1ea20c4a upstream.

    The usb support for asyncio encoded one of it's values in the wrong
    field.  It should have used si_value but instead used si_addr which is
    not present in the _rt union member of struct siginfo.

    The practical result of this is that on a 64bit big endian kernel
    when delivering a signal to a 32bit process the si_addr field
    is set to NULL, instead of the expected pointer value.

    This issue can not be fixed in copy_siginfo_to_user32 as the usb
```

usage of the the _sigfault (aka si_addr) member of the siginfo
union when SI_ASYNCIO is set is incompatible with the POSIX and
glibc usage of the _rt member of the siginfo union.

Therefore replace kill_pid_info_as_cred with kill_pid_usb_asyncio a
dedicated function for this one specific case. There are no other
users of kill_pid_info_as_cred so this specialization should have no
impact on the amount of code in the kernel. Have kill_pid_usb_asyncio
take instead of a siginfo_t which is difficult and error prone, 3
arguments, a signal number, an errno value, and an address enconded as
a sigval_t. The encoding of the address as a sigval_t allows the
code that reads the userspace request for a signal to handle this
compat issue along with all of the other compat issues.

Add BUILD_BUG_ONs in kernel/signal.c to ensure that we can now place
the pointer value at the in si_pid (instead of si_addr). That is the
code now verifies that si_pid and si_addr always occur at the same
location. Further the code veries that for native structures a value
placed in si_pid and spilling into si_uid will appear in userspace in
si_addr (on a byte by byte copy of siginfo or a field by field copy of
siginfo). The code also verifies that for a 64bit kernel and a 32bit
userspace the 32bit pointer will fit in si_pid.

I have used the usbsig.c program below written by Alan Stern and
slightly tweaked by me to run on a big endian machine to verify the
issue exists (on sparc64) and to confirm the patch below fixes the issue.

```
 /* usbsig.c -- test USB async signal delivery */

 #define _GNU_SOURCE
 #include <stdio.h>
 #include <fcntl.h>
 #include <signal.h>
 #include <string.h>
 #include <sys/ioctl.h>
 #include <unistd.h>
 #include <endian.h>
 #include <linux/usb/ch9.h>
 #include <linux/usbdevice_fs.h>

 static struct usbdevfs_urb urb;
 static struct usbdevfs_disconnectsignal ds;
 static volatile sig_atomic_t done = 0;

 void urb_handler(int sig, siginfo_t *info , void *ucontext)
 {
        printf("Got signal %d, signo %d errno %d code %d addr: %p urb: %p\n",
               sig, info->si_signo, info->si_errno, info->si_code,
               info->si_addr, &urb);

        printf("%s\n", (info->si_addr == &urb) ? "Good" : "Bad");
 }

 void ds_handler(int sig, siginfo_t *info , void *ucontext)
 {
        printf("Got signal %d, signo %d errno %d code %d addr: %p ds: %p\n",
               sig, info->si_signo, info->si_errno, info->si_code,
               info->si_addr, &ds);

        printf("%s\n", (info->si_addr == &ds) ? "Good" : "Bad");
        done = 1;
 }

 int main(int argc, char **argv)
 {
```

```
char *devfilename;
int fd;
int rc;
struct sigaction act;
struct usb_ctrlrequest *req;
void *ptr;
char buf[80];

if (argc != 2) {
        fprintf(stderr, "Usage: usbsig device-file-name\n");
        return 1;
}

devfilename = argv[1];
fd = open(devfilename, O_RDWR);
if (fd == -1) {
        perror("Error opening device file");
        return 1;
}

act.sa_sigaction = urb_handler;
sigemptyset(&act.sa_mask);
act.sa_flags = SA_SIGINFO;

rc = sigaction(SIGUSR1, &act, NULL);
if (rc == -1) {
        perror("Error in sigaction");
        return 1;
}

act.sa_sigaction = ds_handler;
sigemptyset(&act.sa_mask);
act.sa_flags = SA_SIGINFO;

rc = sigaction(SIGUSR2, &act, NULL);
if (rc == -1) {
        perror("Error in sigaction");
        return 1;
}

memset(&urb, 0, sizeof(urb));
urb.type = USBDEVFS_URB_TYPE_CONTROL;
urb.endpoint = USB_DIR_IN | 0;
urb.buffer = buf;
urb.buffer_length = sizeof(buf);
urb.signr = SIGUSR1;

req = (struct usb_ctrlrequest *) buf;
req->bRequestType = USB_DIR_IN | USB_TYPE_STANDARD | USB_RECIP_DEVICE;
req->bRequest = USB_REQ_GET_DESCRIPTOR;
req->wValue = htole16(USB_DT_DEVICE << 8);
req->wIndex = htole16(0);
req->wLength = htole16(sizeof(buf) - sizeof(*req));

rc = ioctl(fd, USBDEVFS_SUBMITURB, &urb);
if (rc == -1) {
        perror("Error in SUBMITURB ioctl");
        return 1;
}

rc = ioctl(fd, USBDEVFS_REAPURB, &ptr);
if (rc == -1) {
        perror("Error in REAPURB ioctl");
        return 1;
}
```

```
             memset(&ds, 0, sizeof(ds));
             ds.signr = SIGUSR2;
             ds.context = &ds;
             rc = ioctl(fd, USBDEVFS_DISCSIGNAL, &ds);
             if (rc == -1) {
                     perror("Error in DISCSIGNAL ioctl");
                     return 1;
             }

             printf("Waiting for usb disconnect\n");
             while (!done) {
                     sleep(1);
             }

             close(fd);
             return 0;
      }
```

```
    Cc: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Cc: linux-usb@vger.kernel.org
    Cc: Alan Stern <stern@rowland.harvard.edu>
    Cc: Oliver Neukum <oneukum@suse.com>
    Fixes: v2.3.39
    Cc: stable@vger.kernel.org
    Acked-by: Alan Stern <stern@rowland.harvard.edu>
    Signed-off-by: "Eric W. Biederman" <ebiederm@xmission.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e7afbfb30988a4cb62b95f1f1869c5c095915a77
Author: Shaokun Zhang <zhangshaokun@hisilicon.com>
Date:   Fri Jun 21 19:19:27 2019 +0300

    intel_th: msu: Fix unused variable warning on arm64 platform

    commit b96fb368b08f1637cbf780a6b83e36c2c5ed4ff5 upstream.

    Commit ba39bd8306057 ("intel_th: msu: Switch over to scatterlist")
    introduced the following warnings on non-x86 architectures, as a result
    of reordering the multi mode buffer allocation sequence:

    > drivers/hwtracing/intel_th/msu.c: In function ‘msc_buffer_win_alloc’:
    > drivers/hwtracing/intel_th/msu.c:783:21: warning: unused variable ‘i’
    > [-Wunused-variable]
    > int ret = -ENOMEM, i;
    >                     ^
    > drivers/hwtracing/intel_th/msu.c: In function ‘msc_buffer_win_free’:
    > drivers/hwtracing/intel_th/msu.c:863:6: warning: unused variable ‘i’
    > [-Wunused-variable]
    > int i;
    >     ^

    Fix this compiler warning by factoring out set_memory sequences and making
    them x86-only.

    Suggested-by: Alexander Shishkin <alexander.shishkin@linux.intel.com>
    Signed-off-by: Shaokun Zhang <zhangshaokun@hisilicon.com>
    Fixes: ba39bd8306057 ("intel_th: msu: Switch over to scatterlist")
    Reviewed-by: Andy Shevchenko <andriy.shevchenko@linux.intel.com>
    Signed-off-by: Alexander Shishkin <alexander.shishkin@linux.intel.com>
    Cc: stable <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20190621161930.60785-2-
alexander.shishkin@linux.intel.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit 96495465fd059d5e4606f2afe17bd474e901083b
Author: Julien Thierry <julien.thierry@arm.com>
Date:   Tue Jun 11 10:38:10 2019 +0100

    arm64: Fix incorrect irqflag restore for priority masking

    commit bd82d4bd21880b7c4d5f5756be435095d6ae07b5 upstream.

    When using IRQ priority masking to disable interrupts, in order to deal
    with the PSR.I state, local_irq_save() would convert the I bit into a
    PMR value (GIC_PRIO_IRQOFF). This resulted in local_irq_restore()
    potentially modifying the value of PMR in undesired location due to the
    state of PSR.I upon flag saving [1].

    In an attempt to solve this issue in a less hackish manner, introduce
    a bit (GIC_PRIO_IGNORE_PMR) for the PMR values that can represent
    whether PSR.I is being used to disable interrupts, in which case it
    takes precedence of the status of interrupt masking via PMR.

    GIC_PRIO_PSR_I_SET is chosen such that (<pmr_value> |
    GIC_PRIO_PSR_I_SET) does not mask more interrupts than <pmr_value> as
    some sections (e.g. arch_cpu_idle(), interrupt acknowledge path)
    requires PMR not to mask interrupts that could be signaled to the
    CPU when using only PSR.I.

    [1] https://www.spinics.net/lists/arm-kernel/msg716956.html

    Fixes: 4a503217ce37 ("arm64: irqflags: Use ICC_PMR_EL1 for interrupt masking")
    Cc: <stable@vger.kernel.org> # 5.1.x-
    Reported-by: Zenghui Yu <yuzenghui@huawei.com>
    Cc: Steven Rostedt <rostedt@goodmis.org>
    Cc: Wei Li <liwei391@huawei.com>
    Cc: Will Deacon <will.deacon@arm.com>
    Cc: Christoffer Dall <christoffer.dall@arm.com>
    Cc: James Morse <james.morse@arm.com>
    Cc: Suzuki K Pouloze <suzuki.poulose@arm.com>
    Cc: Oleg Nesterov <oleg@redhat.com>
    Reviewed-by: Marc Zyngier <marc.zyngier@arm.com>
    Signed-off-by: Julien Thierry <julien.thierry@arm.com>
    Signed-off-by: Catalin Marinas <catalin.marinas@arm.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ea37736199adc234187c54e00ff67cf17f32d7f5
Author: Julien Thierry <julien.thierry@arm.com>
Date:   Tue Jun 11 10:38:08 2019 +0100

    arm64: irqflags: Add condition flags to inline asm clobber list

    commit f57065782f245ca96f1472209a485073bbc11247 upstream.

    Some of the inline assembly instruction use the condition flags and need
    to include "cc" in the clobber list.

    Fixes: 4a503217ce37 ("arm64: irqflags: Use ICC_PMR_EL1 for interrupt masking")
    Cc: <stable@vger.kernel.org> # 5.1.x-
    Suggested-by: Marc Zyngier <marc.zyngier@arm.com>
    Cc: Will Deacon <will.deacon@arm.com>
    Reviewed-by: Marc Zyngier <marc.zyngier@arm.com>
    Acked-by: Mark Rutland <mark.rutland@arm.com>
    Signed-off-by: Julien Thierry <julien.thierry@arm.com>
    Signed-off-by: Catalin Marinas <catalin.marinas@arm.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f9458d66de53f13f0c33dddb1163f7c6d2499c0f
Author: Jon Hunter <jonathanh@nvidia.com>
```

```
Date:   Thu Jun 20 09:17:00 2019 +0100

    arm64: tegra: Fix AGIC register range

    commit ba24eee6686f6ed3738602b54d959253316a9541 upstream.

    The Tegra AGIC interrupt controller is an ARM GIC400 interrupt
    controller. Per the ARM GIC device-tree binding, the first address
    region is for the GIC distributor registers and the second address
    region is for the GIC CPU interface registers. The address space for
    the distributor registers is 4kB, but currently this is incorrectly
    defined as 8kB for the Tegra AGIC and overlaps with the CPU interface
    registers. Correct the address space for the distributor to be 4kB.

    Cc: stable@vger.kernel.org
    Signed-off-by: Jon Hunter <jonathanh@nvidia.com>
    Fixes: bcdbde433542 ("arm64: tegra: Add AGIC node for Tegra210")
    Signed-off-by: Thierry Reding <treding@nvidia.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 9f062aef7356f41e37bdfb513c09e3b96fa49ec8
Author: Like Xu <like.xu@linux.intel.com>
Date:   Thu Jul 18 13:35:14 2019 +0800

    KVM: x86/vPMU: refine kvm_pmu err msg when event creation failed

    commit 6fc3977ccc5d3c22e851f2dce2d3ce2a0a843842 upstream.

    If a perf_event creation fails due to any reason of the host perf
    subsystem, it has no chance to log the corresponding event for guest
    which may cause abnormal sampling data in guest result. In debug mode,
    this message helps to understand the state of vPMC and we may not
    limit the number of occurrences but not in a spamming style.

    Suggested-by: Joe Perches <joe@perches.com>
    Signed-off-by: Like Xu <like.xu@linux.intel.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Paolo Bonzini <pbonzini@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e00821d97dede291e84a76790e28632a708000da
Author: Michael Neuling <mikey@neuling.org>
Date:   Thu Jun 20 16:00:40 2019 +1000

    KVM: PPC: Book3S HV: Fix CR0 setting in TM emulation

    commit 3fefd1cd95df04da67c83c1cb93b663f04b3324f upstream.

    When emulating tsr, treclaim and trechkpt, we incorrectly set CR0. The
    code currently sets:
        CR0 <- 00 || MSR[TS]
    but according to the ISA it should be:
        CR0 <-  0 || MSR[TS] || 0

    This fixes the bit shift to put the bits in the correct location.

    This is a data integrity issue as CR0 is corrupted.

    Fixes: 4bb3c7a0208f ("KVM: PPC: Book3S HV: Work around transactional memory bugs in
POWER9")
    Cc: stable@vger.kernel.org # v4.17+
    Tested-by: Suraj Jitindar Singh <sjitindarsingh@gmail.com>
    Signed-off-by: Michael Neuling <mikey@neuling.org>
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit 0b9293c22671538259198e050f1236df2ad780d3
Author: Suraj Jitindar Singh <sjitindarsingh@gmail.com>
Date:   Thu Jun 20 11:46:51 2019 +1000

    KVM: PPC: Book3S HV: Clear pending decrementer exceptions on nested guest entry

    commit 3c25ab35fbc8526ac0c9b298e8a78e7ad7a55479 upstream.

    If we enter an L1 guest with a pending decrementer exception then this
    is cleared on guest exit if the guest has writtien a positive value
    into the decrementer (indicating that it handled the decrementer
    exception) since there is no other way to detect that the guest has
    handled the pending exception and that it should be dequeued. In the
    event that the L1 guest tries to run a nested (L2) guest immediately
    after this and the L2 guest decrementer is negative (which is loaded
    by L1 before making the H_ENTER_NESTED hcall), then the pending
    decrementer exception isn't cleared and the L2 entry is blocked since
    L1 has a pending exception, even though L1 may have already handled
    the exception and written a positive value for it's decrementer. This
    results in a loop of L1 trying to enter the L2 guest and L0 blocking
    the entry since L1 has an interrupt pending with the outcome being
    that L2 never gets to run and hangs.

    Fix this by clearing any pending decrementer exceptions when L1 makes
    the H_ENTER_NESTED hcall since it won't do this if it's decrementer
    has gone negative, and anyway it's decrementer has been communicated
    to L0 in the hdec_expires field and L0 will return control to L1 when
    this goes negative by delivering an H_DECREMENTER exception.

    Fixes: 95a6432ce903 ("KVM: PPC: Book3S HV: Streamlined guest entry/exit path on P9 for
radix guests")
    Cc: stable@vger.kernel.org # v4.20+
    Signed-off-by: Suraj Jitindar Singh <sjitindarsingh@gmail.com>
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c86cc6eb69829ecddcb5078aeeff5a2bbee4edb9
Author: Suraj Jitindar Singh <sjitindarsingh@gmail.com>
Date:   Thu Jun 20 11:46:50 2019 +1000

    KVM: PPC: Book3S HV: Signed extend decrementer value if not using large decrementer

    commit 869537709ebf1dc865e75c3fc97b23f8acf37c16 upstream.

    On POWER9 the decrementer can operate in large decrementer mode where
    the decrementer is 56 bits and signed extended to 64 bits. When not
    operating in this mode the decrementer behaves as a 32 bit decrementer
    which is NOT signed extended (as on POWER8).

    Currently when reading a guest decrementer value we don't take into
    account whether the large decrementer is enabled or not, and this
    means the value will be incorrect when the guest is not using the
    large decrementer. Fix this by sign extending the value read when the
    guest isn't using the large decrementer.

    Fixes: 95a6432ce903 ("KVM: PPC: Book3S HV: Streamlined guest entry/exit path on P9 for
radix guests")
    Cc: stable@vger.kernel.org # v4.20+
    Signed-off-by: Suraj Jitindar Singh <sjitindarsingh@gmail.com>
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3ac6994561a29bff4bdf65833d6a88c0abab1691
Author: KarimAllah Ahmed <karahmed@amazon.de>
```

```
Date:   Wed Jul 10 11:13:13 2019 +0200

    KVM: Properly check if "page" is valid in kvm_vcpu_unmap

    commit b614c6027896ff9ad6757122e84760d938cab15e upstream.

    The field "page" is initialized to KVM_UNMAPPED_PAGE when it is not used
    (i.e. when the memory lives outside kernel control). So this check will
    always end up using kunmap even for memremap regions.

    Fixes: e45adf665a53 ("KVM: Introduce a new guest mapping API")
    Cc: stable@vger.kernel.org
    Signed-off-by: KarimAllah Ahmed <karahmed@amazon.de>
    Signed-off-by: Paolo Bonzini <pbonzini@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 700ec4462661df192effa94b9546a6a0d63d0a2f
Author: Wanpeng Li <wanpengli@tencent.com>
Date:   Thu Jun 20 17:00:02 2019 +0800

    KVM: VMX: check CPUID before allowing read/write of IA32_XSS

    commit 4d763b168e9c5c366b05812c7bba7662e5ea3669 upstream.

    Raise #GP when guest read/write IA32_XSS, but the CPUID bits
    say that it shouldn't exist.

    Fixes: 203000993de5 (kvm: vmx: add MSR logic for XSAVES)
    Reported-by: Xiaoyao Li <xiaoyao.li@linux.intel.com>
    Reported-by: Tao Xu <tao3.xu@intel.com>
    Cc: Paolo Bonzini <pbonzini@redhat.com>
    Cc: Radim Krčmář <rkrcmar@redhat.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Wanpeng Li <wanpengli@tencent.com>
    Signed-off-by: Paolo Bonzini <pbonzini@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e4472749aad6ffde1a080a37c160ceef4feefbaf
Author: Sean Christopherson <sean.j.christopherson@intel.com>
Date:   Fri Apr 19 22:50:55 2019 -0700

    KVM: VMX: Fix handling of #MC that occurs during VM-Entry

    commit beb8d93b3e423043e079ef3dda19dad7b28467a8 upstream.

    A previous fix to prevent KVM from consuming stale VMCS state after a
    failed VM-Entry inadvertantly blocked KVM's handling of machine checks
    that occur during VM-Entry.

    Per Intel's SDM, a #MC during VM-Entry is handled in one of three ways,
    depending on when the #MC is recognoized.  As it pertains to this bug
    fix, the third case explicitly states EXIT_REASON_MCE_DURING_VMENTRY
    is handled like any other VM-Exit during VM-Entry, i.e. sets bit 31 to
    indicate the VM-Entry failed.

    If a machine-check event occurs during a VM entry, one of the following occurs:
     - The machine-check event is handled as if it occurred before the VM entry:
           ...
     - The machine-check event is handled after VM entry completes:
           ...
     - A VM-entry failure occurs as described in Section 26.7. The basic
       exit reason is 41, for "VM-entry failure due to machine-check event".

    Explicitly handle EXIT_REASON_MCE_DURING_VMENTRY as a one-off case in
    vmx_vcpu_run() instead of binning it into vmx_complete_atomic_exit().
```

```
    Doing so allows vmx_vcpu_run() to handle VMX_EXIT_REASONS_FAILED_VMENTRY
    in a sane fashion and also simplifies vmx_complete_atomic_exit() since
    VMCS.VM_EXIT_INTR_INFO is guaranteed to be fresh.

    Fixes: b060ca3b2e9e7 ("kvm: vmx: Handle VMLAUNCH/VMRESUME failure properly")
    Cc: stable@vger.kernel.org
    Signed-off-by: Sean Christopherson <sean.j.christopherson@intel.com>
    Reviewed-by: Jim Mattson <jmattson@google.com>
    Signed-off-by: Paolo Bonzini <pbonzini@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 6e91f6d4113e577e51a930a3a8129269c146c295
Author: Sean Christopherson <sean.j.christopherson@intel.com>
Date:   Tue May 7 09:06:28 2019 -0700

    KVM: nVMX: Always sync GUEST_BNDCFGS when it comes from vmcs01

    commit 3b013a2972d5bc344d6eaa8f24fdfe268211e45f upstream.

    If L1 does not set VM_ENTRY_LOAD_BNDCFGS, then L1's BNDCFGS value must
    be propagated to vmcs02 since KVM always runs with VM_ENTRY_LOAD_BNDCFGS
    when MPX is supported.  Because the value effectively comes from vmcs01,
    vmcs02 must be updated even if vmcs12 is clean.

    Fixes: 62cf9bd8118c4 ("KVM: nVMX: Fix emulation of VM_ENTRY_LOAD_BNDCFGS")
    Cc: stable@vger.kernel.org
    Cc: Liran Alon <liran.alon@oracle.com>
    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 f4da0a8823dbb9ff9cf467093650b4b41f9cc17a
Author: Sean Christopherson <sean.j.christopherson@intel.com>
Date:   Tue May 7 09:06:27 2019 -0700

    KVM: VMX: Always signal #GP on WRMSR to MSR_IA32_CR_PAT with bad value

    commit d28f4290b53a157191ed9991ad05dffe9e8c0c89 upstream.

    The behavior of WRMSR is in no way dependent on whether or not KVM
    consumes the value.

    Fixes: 4566654bb9be9 ("KVM: vmx: Inject #GP on invalid PAT CR")
    Cc: stable@vger.kernel.org
    Cc: Nadav Amit <nadav.amit@gmail.com>
    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 1d83b4652e023a0bb86e1689882e46fc28f2184a
Author: Sean Christopherson <sean.j.christopherson@intel.com>
Date:   Tue May 7 09:06:26 2019 -0700

    KVM: nVMX: Don't dump VMCS if virtual APIC page can't be mapped

    commit 73cb85568433feadb79e963bf2efba9b3e9ae3df upstream.

    ... as a malicious userspace can run a toy guest to generate invalid
    virtual-APIC page addresses in L1, i.e. flood the kernel log with error
    messages.

    Fixes: 690908104e39d ("KVM: nVMX: allow tests to use bad virtual-APIC page address")
    Cc: stable@vger.kernel.org
    Cc: Paolo Bonzini <pbonzini@redhat.com>
    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 91fc7faf95ceabf343c4b01e7fd0e42f6bfc4a99
Author: Sakari Ailus <sakari.ailus@linux.intel.com>
Date:   Wed Dec 12 07:44:14 2018 -0500

    media: videobuf2-dma-sg: Prevent size from overflowing

    commit 14f28f5cea9e3998442de87846d1907a531b6748 upstream.

    buf->size is an unsigned long; casting that to int will lead to an
    overflow if buf->size exceeds INT_MAX.

    Fix this by changing the type to unsigned long instead. This is possible
    as the buf->size is always aligned to PAGE_SIZE, and therefore the size
    will never have values lesser than 0.

    Note on backporting to stable: the file used to be under
    drivers/media/v4l2-core, it was moved to the current location after 4.14.

    Signed-off-by: Sakari Ailus <sakari.ailus@linux.intel.com>
    Cc: stable@vger.kernel.org
    Reviewed-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 d2b1e734f49803ee4dcf299f504734c10d80c4b2
Author: Sakari Ailus <sakari.ailus@linux.intel.com>
Date:   Wed Dec 12 07:27:10 2018 -0500

    media: videobuf2-core: Prevent size alignment wrapping buffer size to 0

    commit defcdc5d89ced780fb45196d539d6570ec5b1ba5 upstream.

    PAGE_ALIGN() may wrap the buffer size around to 0. Prevent this by
    checking that the aligned value is not smaller than the unaligned one.

    Note on backporting to stable: the file used to be under
    drivers/media/v4l2-core, it was moved to the current location after 4.14.

    Signed-off-by: Sakari Ailus <sakari.ailus@linux.intel.com>
    Cc: stable@vger.kernel.org
    Reviewed-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 33397c9d5e62dd4b142dcd1f38d15ee1bc5229d3
Author: Ezequiel Garcia <ezequiel@collabora.com>
Date:   Thu May 2 18:00:43 2019 -0400

    media: coda: Remove unbalanced and unneeded mutex unlock

    commit 766b9b168f6c75c350dd87c3e0bc6a9b322f0013 upstream.

    The mutex unlock in the threaded interrupt handler is not paired
    with any mutex lock. Remove it.

    This bug has been here for a really long time, so it applies
    to any stable repo.

    Reviewed-by: Philipp Zabel <p.zabel@pengutronix.de>
    Signed-off-by: Ezequiel Garcia <ezequiel@collabora.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Cc: stable@vger.kernel.org
```

```
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 0d37d2442289fdf876fc5080def1052440f0f735
Author: Boris Brezillon <boris.brezillon@collabora.com>
Date:   Wed Jun 19 05:21:33 2019 -0400

    media: v4l2: Test type instead of cfg->type in v4l2_ctrl_new_custom()

    commit 07d89227a983df957a6a7c56f7c040cde9ac571f upstream.

    cfg->type can be overridden by v4l2_ctrl_fill() and the new value is
    stored in the local type var. Fix the tests to use this local var.

    Fixes: 0996517cf8ea ("V4L/DVB: v4l2: Add new control handling framework")
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Boris Brezillon <boris.brezillon@collabora.com>
    [hverkuil-cisco@xs4all.nl: change to !qmenu and !qmenu_int (checkpatch)]
    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 31d15efb4c4d87c5c80f0a89648977d6520be6d8
Author: Yan, Zheng <zyan@redhat.com>
Date:   Sun Jun 2 09:45:38 2019 +0800

    ceph: use ceph_evict_inode to cleanup inode's resource

    commit 87bc5b895d94a0f40fe170d4cf5771c8e8f85d15 upstream.

    remove_session_caps() relies on __wait_on_freeing_inode(), to wait for
    freeing inode to remove its caps. But VFS wakes freeing inode waiters
    before calling destroy_inode().

    Cc: stable@vger.kernel.org
    Link: https://tracker.ceph.com/issues/40102
    Signed-off-by: "Yan, Zheng" <zyan@redhat.com>
    Reviewed-by: Jeff Layton <jlayton@redhat.com>
    Signed-off-by: Ilya Dryomov <idryomov@gmail.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 82715656bb1aa4959816eb3dbaa023998196c861
Author: Luis Henriques <lhenriques@suse.com>
Date:   Mon Jul 1 18:16:34 2019 +0100

    ceph: fix end offset in truncate_inode_pages_range call

    commit d31d07b97a5e76f41e00eb81dcca740e84aa7782 upstream.

    Commit e450f4d1a5d6 ("ceph: pass inclusive lend parameter to
    filemap_write_and_wait_range()") fixed the end offset parameter used to
    call filemap_write_and_wait_range and invalidate_inode_pages2_range.
    Unfortunately it missed truncate_inode_pages_range, introducing a
    regression that is easily detected by xfstest generic/130.

    The problem is that when doing direct IO it is possible that an extra page
    is truncated from the page cache when the end offset is page aligned.
    This can cause data loss if that page hasn't been sync'ed to the OSDs.

    While there, change code to use PAGE_ALIGN macro instead.

    Cc: stable@vger.kernel.org
    Fixes: e450f4d1a5d6 ("ceph: pass inclusive lend parameter to
filemap_write_and_wait_range()")
    Signed-off-by: Luis Henriques <lhenriques@suse.com>
```

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

commit bfaf0f9ad25e90208b13c92cc51f86f00a6edc0f
Author: Takashi Iwai <tiwai@suse.de>
Date:   Mon Jul 15 23:14:53 2019 +0200

    ALSA: hda/hdmi - Fix i915 reverse port/pin mapping

    commit 3140aafb22edeab0cc41f15f53b12a118c0ac215 upstream.

    The recent fix for Icelake HDMI codec introduced the mapping from pin
    NID to the i915 gfx port number.  However, it forgot the reverse
    mapping from the port number to the pin NID that is used in the ELD
    notifier callback.  As a result, it's processed to a wrong widget and
    gives a warning like
      snd_hda_codec_hdmi hdaudioC0D2: HDMI: pin nid 5 not registered

    This patch corrects it with a proper reverse mapping function.

    Bugzilla: https://bugzilla.kernel.org/show_bug.cgi?id=204133
    Fixes: b0d8bc50b9f2 ("ALSA: hda: hdmi - add Icelake support")
    Reviewed-by: Kai Vehmanen <kai.vehmanen@linux.intel.com>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 21fcbec8929d89f1a9d5417531528cb01481165c
Author: Takashi Iwai <tiwai@suse.de>
Date:   Mon Jul 15 23:12:13 2019 +0200

    ALSA: hda/hdmi - Remove duplicated define

    commit eb4177116bf568a413c544eca3f4446cb4064be9 upstream.

    INTEL_GET_VENDOR_VERB is defined twice identically.
    Let's remove a superfluous line.

    Fixes: b0d8bc50b9f2 ("ALSA: hda: hdmi - add Icelake support")
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 89f7d875979cfd5923e21639ee59edb161619e4d
Author: Hui Wang <hui.wang@canonical.com>
Date:   Tue Jul 16 15:21:34 2019 +0800

    ALSA: hda/realtek: apply ALC891 headset fixup to one Dell machine

    commit 4b4e0e32e4b09274dbc9d173016c1a026f44608c upstream.

    Without this patch, the headset-mic and headphone-mic don't work.

    Cc: <stable@vger.kernel.org>
    Signed-off-by: Hui Wang <hui.wang@canonical.com>
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ed1729b3bd2ca99a8c360c6699124dd43b1fbc93
Author: Kailang Yang <kailang@realtek.com>
Date:   Mon Jul 15 10:41:50 2019 +0800

    ALSA: hda/realtek - Fixed Headphone Mic can't record on Dell platform
```

```
    commit fbc571290d9f7bfe089c50f4ac4028dd98ebfe98 upstream.

    It assigned to wrong model. So, The headphone Mic can't work.

    Fixes: 3f640970a414 ("ALSA: hda - Fix headset mic detection problem for several Dell
laptops")
    Signed-off-by: Kailang Yang <kailang@realtek.com>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ebb9258e65d2f038469bb76f5e7e6309b5c8e113
Author: Takashi Iwai <tiwai@suse.de>
Date:   Tue Jul 16 08:56:51 2019 +0200

    ALSA: hda - Don't resume forcibly i915 HDMI/DP codec

    commit 4914da2fb0c89205790503f20dfdde854f3afdd8 upstream.

    We apply the codec resume forcibly at system resume callback for
    updating and syncing the jack detection state that may have changed
    during sleeping.  This is, however, superfluous for the codec like
    Intel HDMI/DP, where the jack detection is managed via the audio
    component notification; i.e. the jack state change shall be reported
    sooner or later from the graphics side at mode change.

    This patch changes the codec resume callback to avoid the forcible
    resume conditionally with a new flag, codec->relaxed_resume, for
    reducing the resume time.  The flag is set in the codec probe.

    Although this doesn't fix the entire bug mentioned in the bugzilla
    entry below, it's still a good optimization and some improvements are
    seen.

    Bugzilla: https://bugzilla.kernel.org/show_bug.cgi?id=201901
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a9865cf13a6edfedac84cc0d53023ca0a9267cfe
Author: Takashi Iwai <tiwai@suse.de>
Date:   Mon Jul 15 22:50:27 2019 +0200

    ALSA: seq: Break too long mutex context in the write loop

    commit ede34f397ddb063b145b9e7d79c6026f819ded13 upstream.

    The fix for the racy writes and ioctls to sequencer widened the
    application of client->ioctl_mutex to the whole write loop.  Although
    it does unlock/relock for the lengthy operation like the event dup,
    the loop keeps the ioctl_mutex for the whole time in other
    situations.  This may take quite long time if the user-space would
    give a huge buffer, and this is a likely cause of some weird behavior
    spotted by syzcaller fuzzer.

    This patch puts a simple workaround, just adding a mutex break in the
    loop when a large number of events have been processed.  This
    shouldn't hit any performance drop because the threshold is set high
    enough for usual operations.

    Fixes: 7bd800915677 ("ALSA: seq: More protection for concurrent write and ioctl races")
    Reported-by: syzbot+97aae04ce27e39cbfca9@syzkaller.appspotmail.com
    Reported-by: syzbot+4c595632b98bb8ffcc66@syzkaller.appspotmail.com
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
```

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

commit c7b5dbbacb0dace989c57a726aa68be72e746441
Author: Masahiro Yamada <yamada.masahiro@socionext.com>
Date:   Fri Jul 12 15:07:09 2019 +0900

    kconfig: fix missing choice values in auto.conf

    commit 8e2442a5f86e1f77b86401fce274a7f622740bc4 upstream.

    Since commit 00c864f8903d ("kconfig: allow all config targets to write
    auto.conf if missing"), Kconfig creates include/config/auto.conf in the
    defconfig stage when it is missing.

    Joonas Kylmälä reported incorrect auto.conf generation under some
    circumstances.

    To reproduce it, apply the following diff:

    |  --- a/arch/arm/configs/imx_v6_v7_defconfig
    |  +++ b/arch/arm/configs/imx_v6_v7_defconfig
    |  @@ -345,14 +345,7 @@ CONFIG_USB_CONFIGFS_F_MIDI=y
    |   CONFIG_USB_CONFIGFS_F_HID=y
    |   CONFIG_USB_CONFIGFS_F_UVC=y
    |   CONFIG_USB_CONFIGFS_F_PRINTER=y
    |  -CONFIG_USB_ZERO=m
    |  -CONFIG_USB_AUDIO=m
    |  -CONFIG_USB_ETH=m
    |  -CONFIG_USB_G_NCM=m
    |  -CONFIG_USB_GADGETFS=m
    |  -CONFIG_USB_FUNCTIONFS=m
    |  -CONFIG_USB_MASS_STORAGE=m
    |  -CONFIG_USB_G_SERIAL=m
    |  +CONFIG_USB_FUNCTIONFS=y
    |   CONFIG_MMC=y
    |   CONFIG_MMC_SDHCI=y
    |   CONFIG_MMC_SDHCI_PLTFM=y

    And then, run:

    $ make ARCH=arm mrproper imx_v6_v7_defconfig

    You will see CONFIG_USB_FUNCTIONFS=y is correctly contained in the
    .config, but not in the auto.conf.

    Please note drivers/usb/gadget/legacy/Kconfig is included from a choice
    block in drivers/usb/gadget/Kconfig. So USB_FUNCTIONFS is a choice value.

    This is probably a similar situation described in commit beaaddb62540
    ("kconfig: tests: test defconfig when two choices interact").

    When sym_calc_choice() is called, the choice symbol forgets the
    SYMBOL_DEF_USER unless all of its choice values are explicitly set by
    the user.

    The choice symbol is given just one chance to recall it because
    set_all_choice_values() is called if SYMBOL_NEED_SET_CHOICE_VALUES
    is set.

    When sym_calc_choice() is called again, the choice symbol forgets it
    forever, since SYMBOL_NEED_SET_CHOICE_VALUES is a one-time aid.
    Hence, we cannot call sym_clear_all_valid() again and again.

    It is crazy to repeat set and unset of internal flags. However, we
    cannot simply get rid of "sym->flags &= flags | ~SYMBOL_DEF_USER;"
```

```
    Doing so would re-introduce the problem solved by commit 5d09598d488f
    ("kconfig: fix new choices being skipped upon config update").

    To work around the issue, conf_write_autoconf() stopped calling
    sym_clear_all_valid().

    conf_write() must be changed accordingly. Currently, it clears
    SYMBOL_WRITE after the symbol is written into the .config file. This
    is needed to prevent it from writing the same symbol multiple times in
    case the symbol is declared in two or more locations. I added the new
    flag SYMBOL_WRITTEN, to track the symbols that have been written.

    Anyway, this is a cheesy workaround in order to suppress the issue
    as far as defconfig is concerned.

    Handling of choices is totally broken. sym_clear_all_valid() is called
    every time a user touches a symbol from the GUI interface. To reproduce
    it, just add a new symbol drivers/usb/gadget/legacy/Kconfig, then touch
    around unrelated symbols from menuconfig. USB_FUNCTIONFS will disappear
    from the .config file.

    I added the Fixes tag since it is more fatal than before. But, this
    has been broken since long long time before, and still it is.
    We should take a closer look to fix this correctly somehow.

    Fixes: 00c864f8903d ("kconfig: allow all config targets to write auto.conf if missing")
    Cc: linux-stable <stable@vger.kernel.org> # 4.19+
    Reported-by: Joonas Kylmälä <joonas.kylmala@iki.fi>
    Signed-off-by: Masahiro Yamada <yamada.masahiro@socionext.com>
    Tested-by: Joonas Kylmälä <joonas.kylmala@iki.fi>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f4ca6021b566c6356d5d2472672251c47799c31a
Author: Xiao Ni <xni@redhat.com>
Date:   Fri Jun 14 15:41:05 2019 -0700

    raid5-cache: Need to do start() part job after adding journal device

    commit d9771f5ec46c282d518b453c793635dbdc3a2a94 upstream.

    commit d5d885fd514f ("md: introduce new personality funciton start()")
    splits the init job to two parts. The first part run() does the jobs that
    do not require the md threads. The second part start() does the jobs that
    require the md threads.

    Now it just does run() in adding new journal device. It needs to do the
    second part start() too.

    Fixes: d5d885fd514f ("md: introduce new personality funciton start()")
    Cc: stable@vger.kernel.org #v4.9+
    Reported-by: Michal Soltys <soltys@ziu.info>
    Signed-off-by: Xiao Ni <xni@redhat.com>
    Signed-off-by: Song Liu <songliubraving@fb.com>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 906c14d095998ea7b173ebcc9b02a1b75a013b65
Author: Mark Brown <broonie@kernel.org>
Date:   Fri Jun 21 12:33:56 2019 +0100

    ASoC: core: Adapt for debugfs API change

    commit c2c928c93173f220955030e8440517b87ec7df92 upstream.

    Back in ff9fb72bc07705c (debugfs: return error values, not NULL) the
```

```
    debugfs APIs were changed to return error pointers rather than NULL
    pointers on error, breaking the error checking in ASoC. Update the
    code to use IS_ERR() and log the codes that are returned as part of
    the error messages.

    Fixes: ff9fb72bc07705c (debugfs: return error values, not NULL)
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Cc: stable@vger.kernel.org
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3a25c3e66f7c0e0538319169af0034b8255a6c36
Author: Mark Brown <broonie@kernel.org>
Date:   Fri Jun 21 12:33:57 2019 +0100

    ASoC: dapm: Adapt for debugfs API change

    commit ceaea851b9ea75f9ea2bbefb53ff0d4b27cd5a6e upstream.

    Back in ff9fb72bc07705c (debugfs: return error values, not NULL) the
    debugfs APIs were changed to return error pointers rather than NULL
    pointers on error, breaking the error checking in ASoC. Update the
    code to use IS_ERR() and log the codes that are returned as part of
    the error messages.

    Fixes: ff9fb72bc07705c (debugfs: return error values, not NULL)
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Cc: stable@vger.kernel.org
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 8035ba716c3e00610d28a51f4a397886e7e42129
Author: Christophe Leroy <christophe.leroy@c-s.fr>
Date:   Mon Jun 24 07:20:14 2019 +0000

    lib/scatterlist: Fix mapping iterator when sg->offset is greater than PAGE_SIZE

    commit aeb87246537a83c2aff482f3f34a2e0991e02cbc upstream.

    All mapping iterator logic is based on the assumption that sg->offset
    is always lower than PAGE_SIZE.

    But there are situations where sg->offset is such that the SG item
    is on the second page. In that case sg_copy_to_buffer() fails
    properly copying the data into the buffer. One of the reason is
    that the data will be outside the kmapped area used to access that
    data.

    This patch fixes the issue by adjusting the mapping iterator
    offset and pgoffset fields such that offset is always lower than
    PAGE_SIZE.

    Signed-off-by: Christophe Leroy <christophe.leroy@c-s.fr>
    Fixes: 4225fc8555a9 ("lib/scatterlist: use page iterator in the mapping iterator")
    Cc: stable@vger.kernel.org
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3afd8b1e1568a9357ad268fed8d340552c35a81f
Author: Trond Myklebust <trond.myklebust@hammerspace.com>
Date:   Wed Jul 17 21:22:38 2019 -0400

    SUNRPC: Ensure the bvecs are reset when we re-encode the RPC request

    commit 75369089820473eac45e9ddd970081901a373c08 upstream.
```

```
    The bvec tracks the list of pages, so if the number of pages changes
    due to a re-encode, we need to reset the bvec as well.

    Fixes: 277e4ab7d530 ("SUNRPC: Simplify TCP receive code by switching...")
    Signed-off-by: Trond Myklebust <trond.myklebust@hammerspace.com>
    Cc: stable@vger.kernel.org # v4.20+
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a09d3e3c3092a97d936a8e90e8f247fafcc7b951
Author: Trond Myklebust <trond.myklebust@hammerspace.com>
Date:   Thu Jul 18 15:33:42 2019 -0400

    pnfs: Fix a problem where we gratuitously start doing I/O through the MDS

    commit 58bbeab425c6c5e318f5b6ae31d351331ddfb34b upstream.

    If the client has to stop in pnfs_update_layout() to wait for another
    layoutget to complete, it currently exits and defaults to I/O through
    the MDS if the layoutget was successful.

    Fixes: d03360aaf5cc ("pNFS: Ensure we return the error if someone kills...")
    Signed-off-by: Trond Myklebust <trond.myklebust@hammerspace.com>
    Cc: stable@vger.kernel.org # v4.20+
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 9cd087dacd8145c832fbb7304435c5f5c3614dc2
Author: Trond Myklebust <trond.myklebust@hammerspace.com>
Date:   Wed Jul 17 13:57:44 2019 -0400

    pnfs/flexfiles: Fix PTR_ERR() dereferences in ff_layout_track_ds_error

    commit 8e04fdfadda75a849c649f7e50fe7d97772e1fcb upstream.

    mirror->mirror_ds can be NULL if uninitialised, but can contain
    a PTR_ERR() if call to GETDEVICEINFO failed.

    Fixes: 65990d1afbd2 ("pNFS/flexfiles: Fix a deadlock on LAYOUTGET")
    Signed-off-by: Trond Myklebust <trond.myklebust@hammerspace.com>
    Cc: stable@vger.kernel.org # 4.10+
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3744eb0954fd7486e2387db42efd013198d04153
Author: Max Kellermann <mk@cm4all.com>
Date:   Fri Jul 12 16:18:06 2019 +0200

    Revert "NFS: readdirplus optimization by cache mechanism" (memleak)

    commit db531db951f950b86d274cc8ed7b21b9e2240036 upstream.

    This reverts commit be4c2d4723a4a637f0d1b4f7c66447141a4b3564.

    That commit caused a severe memory leak in nfs_readdir_make_qstr().

    When listing a directory with more than 100 files (this is how many
    struct nfs_cache_array_entry elements fit in one 4kB page), all
    allocated file name strings past those 100 leak.

    The root of the leakage is that those string pointers are managed in
    pages which are never linked into the page cache.

    fs/nfs/dir.c puts pages into the page cache by calling
    read_cache_page(); the callback function nfs_readdir_filler() will
    then fill the given page struct which was passed to it, which is
    already linked in the page cache (by do_read_cache_page() calling
```

```
    add_to_page_cache_lru()).

    Commit be4c2d4723a4 added another (local) array of allocated pages, to
    be filled with more data, instead of discarding excess items received
    from the NFS server.  Those additional pages can be used by the next
    nfs_readdir_filler() call (from within the same nfs_readdir() call).

    The leak happens when some of those additional pages are never used
    (copied to the page cache using copy_highpage()).  The pages will be
    freed by nfs_readdir_free_pages(), but their contents will not.  The
    commit did not invoke nfs_readdir_clear_array() (and doing so would
    have been dangerous, because it did not track which of those pages
    were already copied to the page cache, risking double free bugs).

    How to reproduce the leak:

    - Use a kernel with CONFIG_SLUB_DEBUG_ON.

    - Create a directory on a NFS mount with more than 100 files with
      names long enough to use the "kmalloc-32" slab (so we can easily
      look up the allocation counts):

      for i in `seq 110`; do touch ${i}_0123456789abcdef; done

    - Drop all caches:

      echo 3 >/proc/sys/vm/drop_caches

    - Check the allocation counter:

      grep nfs_readdir /sys/kernel/slab/kmalloc-32/alloc_calls
      30564391 nfs_readdir_add_to_array+0x73/0xd0 age=534558/4791307/6540952 pid=370-1048386
cpus=0-47 nodes=0-1

    - Request a directory listing and check the allocation counters again:

      ls
      [...]
      grep nfs_readdir /sys/kernel/slab/kmalloc-32/alloc_calls
      30564511 nfs_readdir_add_to_array+0x73/0xd0 age=207/4792999/6542663 pid=370-1048386
cpus=0-47 nodes=0-1

    There are now 120 new allocations.

    - Drop all caches and check the counters again:

      echo 3 >/proc/sys/vm/drop_caches
      grep nfs_readdir /sys/kernel/slab/kmalloc-32/alloc_calls
      30564401 nfs_readdir_add_to_array+0x73/0xd0 age=735/4793524/6543176 pid=370-1048386
cpus=0-47 nodes=0-1

    110 allocations are gone, but 10 have leaked and will never be freed.

    Unhelpfully, those allocations are explicitly excluded from KMEMLEAK,
    that's why my initial attempts with KMEMLEAK were not successful:

            /*
             * Avoid a kmemleak false positive. The pointer to the name is stored
             * in a page cache page which kmemleak does not scan.
             */
            kmemleak_not_leak(string->name);

    It would be possible to solve this bug without reverting the whole
    commit:
```

```
      - keep track of which pages were not used, and call
        nfs_readdir_clear_array() on them, or
      - manually link those pages into the page cache

      But for now I have decided to just revert the commit, because the real
      fix would require complex considerations, risking more dangerous
      (crash) bugs, which may seem unsuitable for the stable branches.

      Signed-off-by: Max Kellermann <mk@cm4all.com>
      Cc: stable@vger.kernel.org # v5.1+
      Signed-off-by: Trond Myklebust <trond.myklebust@hammerspace.com>
      Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

  commit eb7a7cfcf021f662748478dff8bff13d7ebe5fc2
  Author: Trond Myklebust <trond.myklebust@hammerspace.com>
  Date:   Thu Jun 27 06:41:45 2019 -0400

      NFSv4: Handle the special Linux file open access mode

      commit 44942b4e457beda00981f616402a1a791e8c616e upstream.

      According to the open() manpage, Linux reserves the access mode 3
      to mean "check for read and write permission on the file and return
      a file descriptor that can't be used for reading or writing."

      Currently, the NFSv4 code will ask the server to open the file,
      and will use an incorrect share access mode of 0. Since it has
      an incorrect share access mode, the client later forgets to send
      a corresponding close, meaning it can leak stateids on the server.

      Fixes: ce4ef7c0a8a05 ("NFS: Split out NFS v4 file operations")
      Cc: stable@vger.kernel.org # 3.6+
      Signed-off-by: Trond Myklebust <trond.myklebust@hammerspace.com>
      Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

  commit 09df5c0ef24c1b50c8700401d2dcf3487df8a85d
  Author: Eiichi Tsukata <devel@etsukata.com>
  Date:   Sun Jun 30 17:54:38 2019 +0900

      tracing: Fix user stack trace "??" output

      commit 6d54ceb539aacc3df65c89500e8b045924f3ef81 upstream.

      Commit c5c27a0a5838 ("x86/stacktrace: Remove the pointless ULONG_MAX
      marker") removes ULONG_MAX marker from user stack trace entries but
      trace_user_stack_print() still uses the marker and it outputs unnecessary
      "??".

      For example:

                  less-1911  [001] d..2    34.758944: <user stack trace>
         =>  <00007f16f2295910>
         => ??
         => ??
         => ??
         => ??
         => ??
         => ??
         => ??

      The user stack trace code zeroes the storage before saving the stack, so if
      the trace is shorter than the maximum number of entries it can terminate
      the print loop if a zero entry is detected.

      Link: http://lkml.kernel.org/r/20190630085438.25545-1-devel@etsukata.com
```

```
    Cc: stable@vger.kernel.org
    Fixes: 4285f2fcef80 ("tracing: Remove the ULONG_MAX stack trace hackery")
    Signed-off-by: Eiichi Tsukata <devel@etsukata.com>
    Signed-off-by: Steven Rostedt (VMware) <rostedt@goodmis.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit afa1d4c43c4c3461b6f646b29c5056ab9a593e85
Author: Julien Thierry <julien.thierry@arm.com>
Date:   Tue Jun 11 10:38:09 2019 +0100

    arm64: Fix interrupt tracing in the presence of NMIs

    commit 17ce302f3117e9518395847a3120c8a108b587b8 upstream.

    In the presence of any form of instrumentation, nmi_enter() should be
    done before calling any traceable code and any instrumentation code.

    Currently, nmi_enter() is done in handle_domain_nmi(), which is much
    too late as instrumentation code might get called before. Move the
    nmi_enter/exit() calls to the arch IRQ vector handler.

    On arm64, it is not possible to know if the IRQ vector handler was
    called because of an NMI before acknowledging the interrupt. However, It
    is possible to know whether normal interrupts could be taken in the
    interrupted context (i.e. if taking an NMI in that context could
    introduce a potential race condition).

    When interrupting a context with IRQs disabled, call nmi_enter() as soon
    as possible. In contexts with IRQs enabled, defer this to the interrupt
    controller, which is in a better position to know if an interrupt taken
    is an NMI.

    Fixes: bc3c03ccb464 ("arm64: Enable the support of pseudo-NMIs")
    Cc: <stable@vger.kernel.org> # 5.1.x-
    Cc: Will Deacon <will.deacon@arm.com>
    Cc: Thomas Gleixner <tglx@linutronix.de>
    Cc: Jason Cooper <jason@lakedaemon.net>
    Cc: Mark Rutland <mark.rutland@arm.com>
    Reviewed-by: Marc Zyngier <marc.zyngier@arm.com>
    Signed-off-by: Julien Thierry <julien.thierry@arm.com>
    Signed-off-by: Catalin Marinas <catalin.marinas@arm.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 28a8baf3c627f3be8aa16890ec68ea5bba94f505
Author: Dmitry Osipenko <digetx@gmail.com>
Date:   Sun Jun 23 20:50:53 2019 +0300

    opp: Don't use IS_ERR on invalid supplies

    commit 560d1bcad715c215e7ffe5d7cffe045974b623d0 upstream.

    _set_opp_custom() receives a set of OPP supplies as its arguments and
    the caller of it passes NULL when the supplies are not valid. But
    _set_opp_custom(), by mistake, checks for error by performing
    IS_ERR(old_supply) on it which will always evaluate to false.

    The problem was spotted during of testing of upcoming update for the
    NVIDIA Tegra CPUFreq driver.

    Cc: stable <stable@vger.kernel.org>
    Fixes: 7e535993fa4f ("OPP: Separate out custom OPP handler specific code")
    Reported-by: Marc Dietrich <marvin24@gmx.de>
    Signed-off-by: Dmitry Osipenko <digetx@gmail.com>
    [ Viresh: Massaged changelog ]
```

```
    Signed-off-by: Viresh Kumar <viresh.kumar@linaro.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 02fbcac747e80ef506080646aecd19996f30a4dc
Author: Emmanuel Grumbach <emmanuel.grumbach@intel.com>
Date:   Wed May 22 12:22:35 2019 +0300

    iwlwifi: mvm: clear rfkill_safe_init_done when we start the firmware

    commit 940225628652b340b2bfe99f42f3d2db9fd9ce6c upstream.

    Otherwise it'll stay set forever which is clearly buggy.

    Cc: stable@vger.kernel.org
    Signed-off-by: Emmanuel Grumbach <emmanuel.grumbach@intel.com>
    Signed-off-by: Luca Coelho <luciano.coelho@intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit d490dd53109ac2879f38da533cde0987ab121491
Author: Johannes Berg <johannes.berg@intel.com>
Date:   Mon May 20 14:49:56 2019 +0200

    iwlwifi: mvm: delay GTK setting in FW in AP mode

    commit c56e00a3feaee2b46b7d33875fb7f52efd30241f upstream.

    In AP (and IBSS) mode, we can only set GTKs to firmware after we have
    sent down the multicast station, but this we can only do after we've
    enabled beaconing, etc.

    However, during rfkill exit, hostapd will configure the keys before
    starting the AP, and cfg80211/mac80211 accept it happily.

    On earlier devices, this didn't bother us as GTK TX wasn't really
    handled in firmware, we just put the key material into the TX cmd
    and thus it only mattered when we actually transmitted a frame.

    On newer devices, however, the firmware needs to track all of this
    and that doesn't work if we add the key before the (multicast) sta
    it belongs to.

    To fix this, keep a list of keys to add during AP enable, and call
    the function there.

    Cc: stable@vger.kernel.org
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Luca Coelho <luciano.coelho@intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit cbc0bf51c970425ea3019a642c61a2b6a699e764
Author: Emmanuel Grumbach <emmanuel.grumbach@intel.com>
Date:   Mon May 20 15:18:24 2019 +0300

    iwlwifi: fix RF-Kill interrupt while FW load for gen2 devices

    commit ed3e4c6d3cd8f093a3636cb05492429fe2af228d upstream.

    Newest devices have a new firmware load mechanism. This
    mechanism is called the context info. It means that the
    driver doesn't need to load the sections of the firmware.
    The driver rather prepares a place in DRAM, with pointers
    to the relevant sections of the firmware, and the firmware
    loads itself.
    At the end of the process, the firmware sends the ALIVE
    interrupt. This is different from the previous scheme in
```

```
      which the driver expected the FH_TX interrupt after each
      section being transferred over the DMA.

      In order to support this new flow, we enabled all the
      interrupts. This broke the assumption that we have in the
      code that the RF-Kill interrupt can't interrupt the firmware
      load flow.

      Change the context info flow to enable only the ALIVE
      interrupt, and re-enable all the other interrupts only
      after the firmware is alive. Then, we won't see the RF-Kill
      interrupt until then. Getting the RF-Kill interrupt while
      loading the firmware made us kill the firmware while it is
      loading and we ended up dumping garbage instead of the firmware
      state.

      Re-enable the ALIVE | RX interrupts from the ISR when we
      get the ALIVE interrupt to be able to get the RX interrupt
      that comes immediately afterwards for the ALIVE
      notification. This is needed for non MSI-X only.

      Cc: stable@vger.kernel.org
      Signed-off-by: Emmanuel Grumbach <emmanuel.grumbach@intel.com>
      Signed-off-by: Luca Coelho <luciano.coelho@intel.com>
      Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

  commit ce0b3a3fbb9d612ff3ffd8ee246fd7565838d4cd
  Author: Emmanuel Grumbach <emmanuel.grumbach@intel.com>
  Date:   Wed May 22 12:17:09 2019 +0300

      iwlwifi: don't WARN when calling iwl_get_shared_mem_conf with RF-Kill

      commit 0d53cfd0cca3c729a089c39eef0e7d8ae7662974 upstream.

      iwl_mvm_send_cmd returns 0 when the command won't be sent
      because RF-Kill is asserted. Do the same when we call
      iwl_get_shared_mem_conf since it is not sent through
      iwl_mvm_send_cmd but directly calls the transport layer.

      Cc: stable@vger.kernel.org
      Signed-off-by: Emmanuel Grumbach <emmanuel.grumbach@intel.com>
      Signed-off-by: Luca Coelho <luciano.coelho@intel.com>
      Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

  commit 4c4359b4e264fd9dba5d57361c9e42d5d363b9ef
  Author: Emmanuel Grumbach <emmanuel.grumbach@intel.com>
  Date:   Tue May 21 15:03:21 2019 +0300

      iwlwifi: pcie: fix ALIVE interrupt handling for gen2 devices w/o MSI-X

      commit ec46ae30245ecb41d73f8254613db07c653fb498 upstream.

      We added code to restock the buffer upon ALIVE interrupt
      when MSI-X is disabled. This was added as part of the context
      info code. This code was added only if the ISR debug level
      is set which is very unlikely to be related.
      Move this code to run even when the ISR debug level is not
      set.

      Note that gen2 devices work with MSI-X in most cases so that
      this path is seldom used.

      Cc: stable@vger.kernel.org
      Signed-off-by: Emmanuel Grumbach <emmanuel.grumbach@intel.com>
      Signed-off-by: Luca Coelho <luciano.coelho@intel.com>
```

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

commit d31f08534842675c6feae8c85df7ecc4c641de89
Author: Emmanuel Grumbach <emmanuel.grumbach@intel.com>
Date:   Tue May 21 15:10:38 2019 +0300

    iwlwifi: pcie: don't service an interrupt that was masked

    commit 3b57a10ca14c619707398dc58fe5ece18c95b20b upstream.

    Sometimes the register status can include interrupts that
    were masked. We can, for example, get the RF-Kill bit set
    in the interrupt status register although this interrupt
    was masked. Then if we get the ALIVE interrupt (for example)
    that was not masked, we need to *not* service the RF-Kill
    interrupt.
    Fix this in the MSI-X interrupt handler.

    Cc: stable@vger.kernel.org
    Signed-off-by: Emmanuel Grumbach <emmanuel.grumbach@intel.com>
    Signed-off-by: Luca Coelho <luciano.coelho@intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b0dd5c6a62c80e80f9d4a9cbce5f660a4082f321
Author: Oren Givon <oren.givon@intel.com>
Date:   Thu Jun 20 11:46:23 2019 +0300

    iwlwifi: add support for hr1 RF ID

    commit 498d3eb5bfbb2e05e40005152976a7b9eadfb59c upstream.

    The 22000 series FW that was meant to be used with hr is
    also the FW that is used for hr1 and has a different RF ID.
    Add support to load the hr FW when hr1 RF ID is detected.

    Cc: stable@vger.kernel.org # 5.1+
    Signed-off-by: Oren Givon <oren.givon@intel.com>
    Signed-off-by: Luciano Coelho <luciano.coelho@intel.com>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 5ba8c02c299dbf452e8de07a866cc5d78130c5ea
Author: Jon Hunter <jonathanh@nvidia.com>
Date:   Thu Jun 20 09:17:02 2019 +0100

    arm64: tegra: Fix Jetson Nano GPU regulator

    commit 434e8aedeaec595933811c2af191db9f11d3ce3b upstream.

    There are a few issues with the GPU regulator defined for Jetson Nano
    which are:

    1. The GPU regulator is a PWM based regulator and not a fixed voltage
       regulator.
    2. The output voltages for the GPU regulator are not correct.
    3. The regulator enable ramp delay is too short for the regulator and
       needs to be increased. 2ms should be sufficient.
    4. This is the same regulator used on Jetson TX1 and so make the ramp
       delay and settling time the same as Jetson TX1.

    Cc: stable@vger.kernel.org
    Signed-off-by: Jon Hunter <jonathanh@nvidia.com>
    Fixes: 6772cd0eacc8 ("arm64: tegra: Add NVIDIA Jetson Nano Developer Kit support")
    Signed-off-by: Thierry Reding <treding@nvidia.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit 57a9365708306734a4f2ea087a7f1c4cee9ece56
Author: Jon Hunter <jonathanh@nvidia.com>
Date:   Thu Jun 20 09:17:01 2019 +0100

    arm64: tegra: Update Jetson TX1 GPU regulator timings

    commit ece6031ece2dd64d63708cfe1088016cee5b10c0 upstream.

    The GPU regulator enable ramp delay for Jetson TX1 is set to 1ms which
    not sufficient because the enable ramp delay has been measured to be
    greater than 1ms. Furthermore, the downstream kernels released by NVIDIA
    for Jetson TX1 are using a enable ramp delay 2ms and a settling delay of
    160us. Update the GPU regulator enable ramp delay for Jetson TX1 to be
    2ms and add a settling delay of 160us.

    Cc: stable@vger.kernel.org
    Signed-off-by: Jon Hunter <jonathanh@nvidia.com>
    Fixes: 5e6b9a89afce ("arm64: tegra: Add VDD_GPU regulator to Jetson TX1")
    Signed-off-by: Thierry Reding <treding@nvidia.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 32380e7b912ba79469d2d0d8d081daa5e0522706
Author: Krzysztof Kozlowski <krzk@kernel.org>
Date:   Sat Jun 29 13:44:45 2019 +0200

    regulator: s2mps11: Fix buck7 and buck8 wrong voltages

    commit 16da0eb5ab6ef2dd1d33431199126e63db9997cc upstream.

    On S2MPS11 device, the buck7 and buck8 regulator voltages start at 750
    mV, not 600 mV.  Using wrong minimal value caused shifting of these
    regulator values by 150 mV (e.g. buck7 usually configured to v1.35 V was
    reported as 1.2 V).

    On most of the boards these regulators are left in default state so this
    was only affecting reported voltage.  However if any driver wanted to
    change them, then effectively it would set voltage 150 mV higher than
    intended.

    Cc: <stable@vger.kernel.org>
    Fixes: cb74685ecb39 ("regulator: s2mps11: Add samsung s2mps11 regulator driver")
    Signed-off-by: Krzysztof Kozlowski <krzk@kernel.org>
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3d85a83ac59695eb9c09e56929a8e0a7571ca4b4
Author: Krzysztof Kozlowski <krzk@kernel.org>
Date:   Wed Jun 19 14:42:39 2019 +0200

    regulator: s2mps11: Fix ERR_PTR dereference on GPIO lookup failure

    commit 70ca117b02f3b1c8830fe95e4e3dea2937038e11 upstream.

    If devm_gpiod_get_from_of_node() call returns ERR_PTR, it is assigned
    into an array of GPIO descriptors and used later because such error is
    not treated as critical thus it is not propagated back to the probe
    function.

    All code later expects that such GPIO descriptor is either a NULL or
    proper value.  This later might lead to dereference of ERR_PTR.

    Only devices with S2MPS14 flavor are affected (other do not control
    regulators with GPIOs).
```

```
    Fixes: 1c984942f0a4 ("regulator: s2mps11: Pass descriptor instead of GPIO number")
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Krzysztof Kozlowski <krzk@kernel.org>
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e0190131305dc5c0a2af892b69caf82223202c63
Author: Hui Wang <hui.wang@canonical.com>
Date:   Fri Jul 19 12:38:58 2019 +0300

    Input: alps - fix a mismatch between a condition check and its comment

    commit 771a081e44a9baa1991ef011cc453ef425591740 upstream.

    In the function alps_is_cs19_trackpoint(), we check if the param[1] is
    in the 0x20~0x2f range, but the code we wrote for this checking is not
    correct:
    (param[1] & 0x20) does not mean param[1] is in the range of 0x20~0x2f,
    it also means the param[1] is in the range of 0x30~0x3f, 0x60~0x6f...

    Now fix it with a new condition checking ((param[1] & 0xf0) == 0x20).

    Fixes: 7e4935ccc323 ("Input: alps - don't handle ALPS cs19 trackpoint-only device")
    Cc: stable@vger.kernel.org
    Signed-off-by: Hui Wang <hui.wang@canonical.com>
    Signed-off-by: Dmitry Torokhov <dmitry.torokhov@gmail.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit bba47b85d2f8fe847e70d44a4131789a4aeadb66
Author: Nick Black <dankamongmen@gmail.com>
Date:   Thu Jul 11 23:42:03 2019 -0700

    Input: synaptics - whitelist Lenovo T580 SMBus intertouch

    commit 1976d7d200c5a32e72293a2ada36b7b7c9d6dd6e upstream.

    Adds the Lenovo T580 to the SMBus intertouch list for Synaptics
    touchpads. I've tested with this for a week now, and it seems a great
    improvement. It's also nice to have the complaint gone from dmesg.

    Signed-off-by: Nick Black <dankamongmen@gmail.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Dmitry Torokhov <dmitry.torokhov@gmail.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 33871123fa3a2dff9c3fc3f2c1a6cc3114e12964
Author: Hui Wang <hui.wang@canonical.com>
Date:   Mon Jul 15 10:00:58 2019 -0700

    Input: alps - don't handle ALPS cs19 trackpoint-only device

    commit 7e4935ccc3236751e5fe4bd6846f86e46bb2e427 upstream.

    On a latest Lenovo laptop, the trackpoint and 3 buttons below it
    don't work at all, when we move the trackpoint or press those 3
    buttons, the kernel will print out:
    "Rejected trackstick packet from non DualPoint device"

    This device is identified as an alps touchpad but the packet has
    trackpoint format, so the alps.c drops the packet and prints out
    the message above.

    According to XiaoXiao's explanation, this device is named cs19 and
    is trackpoint-only device, its firmware is only for trackpoint, it
    is independent of touchpad and is a device completely different from
```

```
    DualPoint ones.

    To drive this device with mininal changes to the existing driver, we
    just let the alps driver not handle this device, then the trackpoint.c
    will be the driver of this device if the trackpoint driver is enabled.
    (if not, this device will fallback to a bare PS/2 device)

    With the trackpoint.c, this trackpoint and 3 buttons all work well,
    they have all features that the trackpoint should have, like
    scrolling-screen, drag-and-drop and frame-selection.

    Signed-off-by: XiaoXiao Liu <sliuuxiaonxiao@gmail.com>
    Signed-off-by: Hui Wang <hui.wang@canonical.com>
    Reviewed-by: Pali Rohár <pali.rohar@gmail.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Dmitry Torokhov <dmitry.torokhov@gmail.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 63fabf4287b23da069986b7a7fdc6ad0b202f00a
Author: Grant Hernandez <granthernandez@google.com>
Date:   Sat Jul 13 01:00:12 2019 -0700

    Input: gtco - bounds check collection indent level

    commit 2a017fd82c5402b3c8df5e3d6e5165d9e6147dc1 upstream.

    The GTCO tablet input driver configures itself from an HID report sent
    via USB during the initial enumeration process. Some debugging messages
    are generated during the parsing. A debugging message indentation
    counter is not bounds checked, leading to the ability for a specially
    crafted HID report to cause '-' and null bytes be written past the end
    of the indentation array. As long as the kernel has CONFIG_DYNAMIC_DEBUG
    enabled, this code will not be optimized out.  This was discovered
    during code review after a previous syzkaller bug was found in this
    driver.

    Signed-off-by: Grant Hernandez <granthernandez@google.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Dmitry Torokhov <dmitry.torokhov@gmail.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 26b8adf883f43db0c4bba7af860a0ff6ebdb59ba
Author: Coly Li <colyli@suse.de>
Date:   Fri Jun 28 19:59:44 2019 +0800

    bcache: destroy dc->writeback_write_wq if failed to create dc->writeback_thread

    commit f54d801dda14942dbefa00541d10603015b7859c upstream.

    Commit 9baf30972b55 ("bcache: fix for gc and write-back race") added a
    new work queue dc->writeback_write_wq, but forgot to destroy it in the
    error condition when creating dc->writeback_thread failed.

    This patch destroys dc->writeback_write_wq if kthread_create() returns
    error pointer to dc->writeback_thread, then a memory leak is avoided.

    Fixes: 9baf30972b55 ("bcache: fix for gc and write-back race")
    Signed-off-by: Coly Li <colyli@suse.de>
    Cc: stable@vger.kernel.org
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 8b52937e010fceba8be11242e0eaea0af6143080
Author: Coly Li <colyli@suse.de>
Date:   Fri Jun 28 19:59:43 2019 +0800
```

```
    bcache: fix mistaken sysfs entry for io_error counter

    commit 5461999848e0462c14f306a62923d22de820a59c upstream.

    In bch_cached_dev_files[] from driver/md/bcache/sysfs.c, sysfs_errors is
    incorrectly inserted in. The correct entry should be sysfs_io_errors.

    This patch fixes the problem and now I/O errors of cached device can be
    read from /sys/block/bcache<N>/bcache/io_errors.

    Fixes: c7b7bd07404c5 ("bcache: add io_disable to struct cached_dev")
    Signed-off-by: Coly Li <colyli@suse.de>
    Cc: stable@vger.kernel.org
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 735c54c3cecf40a7d81adcf6d9a8613925b6491d
Author: Coly Li <colyli@suse.de>
Date:   Fri Jun 28 19:59:29 2019 +0800

    bcache: ignore read-ahead request failure on backing device

    commit 578df99b1b0531d19af956530fe4da63d01a1604 upstream.

    When md raid device (e.g. raid456) is used as backing device, read-ahead
    requests on a degrading and recovering md raid device might be failured
    immediately by md raid code, but indeed this md raid array can still be
    read or write for normal I/O requests. Therefore such failed read-ahead
    request are not real hardware failure. Further more, after degrading and
    recovering accomplished, read-ahead requests will be handled by md raid
    array again.

    For such condition, I/O failures of read-ahead requests don't indicate
    real health status (because normal I/O still be served), they should not
    be counted into I/O error counter dc->io_errors.

    Since there is no simple way to detect whether the backing divice is a
    md raid device, this patch simply ignores I/O failures for read-ahead
    bios on backing device, to avoid bogus backing device failure on a
    degrading md raid array.

    Suggested-and-tested-by: Thorsten Knabe <linux@thorsten-knabe.de>
    Signed-off-by: Coly Li <colyli@suse.de>
    Cc: stable@vger.kernel.org
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e294fa562fadad697cd49e2d7883b1b95a8aa230
Author: Coly Li <colyli@suse.de>
Date:   Fri Jun 28 19:59:53 2019 +0800

    bcache: Revert "bcache: free heap cache_set->flush_btree in bch_journal_free"

    commit ba82c1ac1667d6efb91a268edb13fc9cdaecec9b upstream.

    This reverts commit 6268dc2c4703aabfb0b35681be709acf4c2826c6.

    This patch depends on commit c4dc2497d50d ("bcache: fix high CPU
    occupancy during journal") which is reverted in previous patch. So
    revert this one too.

    Fixes: 6268dc2c4703 ("bcache: free heap cache_set->flush_btree in bch_journal_free")
    Signed-off-by: Coly Li <colyli@suse.de>
    Cc: stable@vger.kernel.org
```

```
    Cc: Shenghui Wang <shhuiw@foxmail.com>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit fe4a2764fd85185f9014beecc68510d1f9bdc958
Author: Coly Li <colyli@suse.de>
Date:   Fri Jun 28 19:59:54 2019 +0800

    bcache: Revert "bcache: fix high CPU occupancy during journal"

    commit 249a5f6da57c28a903c75d81505d58ec8c10030d upstream.

    This reverts commit c4dc2497d50d9c6fb16aa0d07b6a14f3b2adb1e0.

    This patch enlarges a race between normal btree flush code path and
    flush_btree_write(), which causes deadlock when journal space is
    exhausted. Reverts this patch makes the race window from 128 btree
    nodes to only 1 btree nodes.

    Fixes: c4dc2497d50d ("bcache: fix high CPU occupancy during journal")
    Signed-off-by: Coly Li <colyli@suse.de>
    Cc: stable@vger.kernel.org
    Cc: Tang Junhui <tang.junhui.linux@gmail.com>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3244e913f1061b3e0b412ea8b664706eb6a7591a
Author: Coly Li <colyli@suse.de>
Date:   Fri Jun 28 19:59:27 2019 +0800

    Revert "bcache: set CACHE_SET_IO_DISABLE in bch_cached_dev_error()"

    commit 695277f16b3a102fcc22c97fdf2de77c7b19f0b3 upstream.

    This reverts commit 6147305c73e4511ca1a975b766b97a779d442567.

    Although this patch helps the failed bcache device to stop faster when
    too many I/O errors detected on corresponding cached device, setting
    CACHE_SET_IO_DISABLE bit to cache set c->flags was not a good idea. This
    operation will disable all I/Os on cache set, which means other attached
    bcache devices won't work neither.

    Without this patch, the failed bcache device can also be stopped
    eventually if internal I/O accomplished (e.g. writeback). Therefore here
    I revert it.

    Fixes: 6147305c73e4 ("bcache: set CACHE_SET_IO_DISABLE in bch_cached_dev_error()")
    Reported-by: Yong Li <mr.liyong@qq.com>
    Signed-off-by: Coly Li <colyli@suse.de>
    Cc: stable@vger.kernel.org
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3ed98f12f8443234a7229250520c0230f716a8b7
Author: Aurelien Aptel <aaptel@suse.com>
Date:   Wed Jul 17 12:46:28 2019 +0200

    CIFS: fix deadlock in cached root handling

    commit 7e5a70ad88b1e6f6d9b934b2efb41afff496820f upstream.

    Prevent deadlock between open_shroot() and
    cifs_mark_open_files_invalid() by releasing the lock before entering
    SMB2_open, taking it again after and checking if we still need to use
    the result.
```

```
    Link: https://lore.kernel.org/linux-cifs/684ed01c-cbca-2716-bc28-b0a59a0f8521@prodrive-
technologies.com/T/#u
    Fixes: 3d4ef9a15343 ("smb3: fix redundant opens on root")
    Signed-off-by: Aurelien Aptel <aaptel@suse.com>
    Reviewed-by: Pavel Shilovsky <pshilov@microsoft.com>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    CC: Stable <stable@vger.kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 8f85f239ad9e07d4fccd832e381801a8403f970d
Author: Ronnie Sahlberg <lsahlber@redhat.com>
Date:   Fri Jul 19 08:12:11 2019 +1000

    cifs: flush before set-info if we have writeable handles

    commit aa081859b10c5d8b19f5c525c78883a59d73c2b8 upstream.

    Servers can defer destaging any data and updating the mtime until close().
    This means that if we do a setinfo to modify the mtime while other handles
    are open for write the server may overwrite our setinfo timestamps when
    if flushes the file on close() of the writeable handle.

    To solve this we add an explicit flush when the mtime is about to
    be updated.

    This fixes "cp -p" to preserve mtime when copying a file onto an SMB2 share.

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

commit b14228f7e98e2638ced1207a324dcfccba624a78
Author: Paulo Alcantara (SUSE) <paulo@paulo.ac>
Date:   Tue Jun 18 16:16:02 2019 -0300

    cifs: Properly handle auto disabling of serverino option

    commit 29fbeb7a908a60a5ae8c50fbe171cb8fdcef1980 upstream.

    Fix mount options comparison when serverino option is turned off later
    in cifs_autodisable_serverino() and thus avoiding mismatch of new cifs
    mounts.

    Cc: stable@vger.kernel.org
    Signed-off-by: Paulo Alcantara (SUSE) <paulo@paulo.ac>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Reviewed-by: Pavel Shilovsky <pshilove@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 98c3db87aae1903f196dee195506150f0ab7edb0
Author: Ronnie Sahlberg <lsahlber@redhat.com>
Date:   Tue Jul 16 10:41:46 2019 +1000

    cifs: fix crash in smb2_compound_op()/smb2_set_next_command()

    commit 88a92c913cef09e70b1744a8877d177aa6cb2189 upstream.

    RHBZ: 1722704

    In low memory situations the various SMB2_*_init() functions can fail
    to allocate a request PDU and thus leave the request iovector as NULL.
```

```
    If we don't check the return code for failure we end up calling
    smb2_set_next_command() with a NULL iovector causing a crash when it tries
    to dereference it.

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

commit 68421c9014d818fb991e98b53f6b05242eab37ed
Author: Ronnie Sahlberg <lsahlber@redhat.com>
Date:   Sat Jul 6 06:43:08 2019 +1000

    cifs: always add credits back for unsolicited PDUs

    commit 3e2725796cbdfe4efc7eb7b27cacaeac2ddad1a5 upstream.

    not just if CONFIG_CIFS_DEBUG2 is enabled.

    Signed-off-by: Ronnie Sahlberg <lsahlber@redhat.com>
    Reviewed-by: Pavel Shilovsky <pshilov@microsoft.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 72cd0e9143cc8fb0cd9713eef2d2d2365a06b3f7
Author: Wen Yang <wen.yang99@zte.com.cn>
Date:   Mon Jul 8 14:19:03 2019 +0800

    crypto: crypto4xx - fix a potential double free in ppc4xx_trng_probe

    commit 95566aa75cd6b3b404502c06f66956b5481194b3 upstream.

    There is a possible double free issue in ppc4xx_trng_probe():

    85:     dev->trng_base = of_iomap(trng, 0);
    86:     of_node_put(trng);          ---> released here
    87:     if (!dev->trng_base)
    88:             goto err_out;
    ...
    110:    ierr_out:
    111:            of_node_put(trng);  ---> double released here
    ...

    This issue was detected by using the Coccinelle software.
    We fix it by removing the unnecessary of_node_put().

    Fixes: 5343e674f32f ("crypto4xx: integrate ppc4xx-rng into crypto4xx")
    Signed-off-by: Wen Yang <wen.yang99@zte.com.cn>
    Cc: <stable@vger.kernel.org>
    Cc: "David S. Miller" <davem@davemloft.net>
    Cc: Thomas Gleixner <tglx@linutronix.de>
    Cc: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Cc: Allison Randal <allison@lohutok.net>
    Cc: Armijn Hemel <armijn@tjaldur.nl>
    Cc: Julia Lawall <Julia.Lawall@lip6.fr>
    Cc: linux-crypto@vger.kernel.org
    Cc: linux-kernel@vger.kernel.org
    Acked-by: Julia Lawall <julia.lawall@lip6.fr>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b9796d1ce2a8ab32533f6a3c74c996dc1eef30c6
Author: Cfir Cohen <cfir@google.com>
Date:   Tue Jul 2 10:32:56 2019 -0700
```

```
    crypto: ccp/gcm - use const time tag comparison.

    commit 538a5a072e6ef04377b180ee9b3ce5bae0a85da4 upstream.

    Avoid leaking GCM tag through timing side channel.

    Fixes: 36cf515b9bbe ("crypto: ccp - Enable support for AES GCM on v5 CCPs")
    Cc: <stable@vger.kernel.org> # v4.12+
    Signed-off-by: Cfir Cohen <cfir@google.com>
    Acked-by: Gary R Hook <ghook@amd.com>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1ad6d2f0e22c39eca086c994f7d03739fd030f7d
Author: Hook, Gary <Gary.Hook@amd.com>
Date:   Wed Jul 10 00:09:22 2019 +0000

    crypto: ccp - memset structure fields to zero before reuse

    commit 20e833dc36355ed642d00067641a679c618303fa upstream.

    The AES GCM function reuses an 'op' data structure, which members
    contain values that must be cleared for each (re)use.

    This fix resolves a crypto self-test failure:
    alg: aead: gcm-aes-ccp encryption test failed (wrong result) on test vector 2, cfg="two
even aligned splits"

    Fixes: 36cf515b9bbe ("crypto: ccp - Enable support for AES GCM on v5 CCPs")
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Gary R Hook <gary.hook@amd.com>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f7ca2ff09361e140306abf9e0a16725cbb94285b
Author: Christian Lamparter <chunkeey@gmail.com>
Date:   Sat May 18 23:28:12 2019 +0200

    crypto: crypto4xx - block ciphers should only accept complete blocks

    commit 0f7a81374060828280fcfdfbaa162cb559017f9f upstream.

    The hardware automatically zero pads incomplete block ciphers
    blocks without raising any errors. This is a screw-up. This
    was noticed by CONFIG_CRYPTO_MANAGER_EXTRA_TESTS tests that
    sent a incomplete blocks and expect them to fail.

    This fixes:
    cbc-aes-ppc4xx encryption unexpectedly succeeded on test vector
    "random: len=2409 klen=32"; expected_error=-22, cfg="random:
    may_sleep use_digest src_divs=[96.90%@+2295, 2.34%@+4066,
    0.32%@alignmask+12, 0.34%@+4087, 0.9%@alignmask+1787, 0.1%@+3767]
    iv_offset=6"

    ecb-aes-ppc4xx encryption unexpectedly succeeded on test vector
    "random: len=1011 klen=32"; expected_error=-22, cfg="random:
    may_sleep use_digest src_divs=[100.0%@alignmask+20]
    dst_divs=[3.12%@+3001, 96.88%@+4070]"

    Cc: Eric Biggers <ebiggers@kernel.org>
    Cc: stable@vger.kernel.org [4.19, 5.0 and 5.1]
    Signed-off-by: Christian Lamparter <chunkeey@gmail.com>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit fe781373ae4c7a6cfaff33501209477fa0ecaeea
Author: Christian Lamparter <chunkeey@gmail.com>
Date:   Sat May 18 23:28:11 2019 +0200

    crypto: crypto4xx - fix blocksize for cfb and ofb

    commit 70c4997f34b6c6888b3ac157adec49e01d0df2d5 upstream.

    While the hardware consider them to be blockciphers, the
    reference implementation defines them as streamciphers.

    Do the right thing and set the blocksize to 1. This
    was found by CONFIG_CRYPTO_MANAGER_EXTRA_TESTS.

    This fixes the following issues:
    skcipher: blocksize for ofb-aes-ppc4xx (16) doesn't match generic impl (1)
    skcipher: blocksize for cfb-aes-ppc4xx (16) doesn't match generic impl (1)

    Cc: Eric Biggers <ebiggers@kernel.org>
    Cc: stable@vger.kernel.org
    Fixes: f2a13e7cba9e ("crypto: crypto4xx - enable AES RFC3686, ECB, CFB and OFB
offloads")
    Signed-off-by: Christian Lamparter <chunkeey@gmail.com>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 87bfbcff1e1a82540ccc875f99c456acf5d6a7df
Author: Christian Lamparter <chunkeey@gmail.com>
Date:   Fri May 17 23:15:57 2019 +0200

    crypto: crypto4xx - fix AES CTR blocksize value

    commit bfa2ba7d9e6b20aca82b99e6842fe18842ae3a0f upstream.

    This patch fixes a issue with crypto4xx's ctr(aes) that was
    discovered by libcapi's kcapi-enc-test.sh test.

    The some of the ctr(aes) encryptions test were failing on the
    non-power-of-two test:

    kcapi-enc - Error: encryption failed with error 0
    kcapi-enc - Error: decryption failed with error 0
    [FAILED: 32-bit - 5.1.0-rc1+] 15 bytes: STDIN / STDOUT enc test (128 bits):
    original file (1d100e..cc96184c) and generated file (e3b0c442..1b7852b855)
    [FAILED: 32-bit - 5.1.0-rc1+] 15 bytes: STDIN / STDOUT enc test (128 bits)
    (openssl generated CT): original file (e3b0..5) and generated file (3..8e)
    [PASSED: 32-bit - 5.1.0-rc1+] 15 bytes: STDIN / STDOUT enc test (128 bits)
    (openssl generated PT)
    [FAILED: 32-bit - 5.1.0-rc1+] 15 bytes: STDIN / STDOUT enc test (password):
    original file (1d1..84c) and generated file (e3b..852b855)

    But the 16, 32, 512, 65536 tests always worked.

    Thankfully, this isn't a hidden hardware problem like previously,
    instead this turned out to be a copy and paste issue.

    With this patch, all the tests are passing with and
    kcapi-enc-test.sh gives crypto4xx's a clean bill of health:
     "Number of failures: 0" :).

    Cc: stable@vger.kernel.org
    Fixes: 98e87e3d933b ("crypto: crypto4xx - add aes-ctr support")
    Fixes: f2a13e7cba9e ("crypto: crypto4xx - enable AES RFC3686, ECB, CFB and OFB
offloads")
```

```
    Signed-off-by: Christian Lamparter <chunkeey@gmail.com>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit baade060ea5ebc422b95e77d110d83e26dc77eed
Author: Eric Biggers <ebiggers@google.com>
Date:   Fri May 31 11:12:30 2019 -0700

    crypto: chacha20poly1305 - fix atomic sleep when using async algorithm

    commit 7545b6c2087f4ef0287c8c9b7eba6a728c67ff8e upstream.

    Clear the CRYPTO_TFM_REQ_MAY_SLEEP flag when the chacha20poly1305
    operation is being continued from an async completion callback, since
    sleeping may not be allowed in that context.

    This is basically the same bug that was recently fixed in the xts and
    lrw templates.  But, it's always been broken in chacha20poly1305 too.
    This was found using syzkaller in combination with the updated crypto
    self-tests which actually test the MAY_SLEEP flag now.

    Reproducer:

        python -c 'import socket; socket.socket(socket.AF_ALG, 5, 0).bind(
                   ("aead", "rfc7539(cryptd(chacha20-generic),poly1305-generic)"))'

    Kernel output:

        BUG: sleeping function called from invalid context at include/crypto/algapi.h:426
        in_atomic(): 1, irqs_disabled(): 0, pid: 1001, name: kworker/2:2
        [...]
        CPU: 2 PID: 1001 Comm: kworker/2:2 Not tainted 5.2.0-rc2 #5
        Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS 1.12.0-20181126_142135-
anatol 04/01/2014
        Workqueue: crypto cryptd_queue_worker
        Call Trace:
         __dump_stack lib/dump_stack.c:77 [inline]
         dump_stack+0x4d/0x6a lib/dump_stack.c:113
         ___might_sleep kernel/sched/core.c:6138 [inline]
         ___might_sleep.cold.19+0x8e/0x9f kernel/sched/core.c:6095
         crypto_yield include/crypto/algapi.h:426 [inline]
         crypto_hash_walk_done+0xd6/0x100 crypto/ahash.c:113
         shash_ahash_update+0x41/0x60 crypto/shash.c:251
         shash_async_update+0xd/0x10 crypto/shash.c:260
         crypto_ahash_update include/crypto/hash.h:539 [inline]
         poly_setkey+0xf6/0x130 crypto/chacha20poly1305.c:337
         poly_init+0x51/0x60 crypto/chacha20poly1305.c:364
         async_done_continue crypto/chacha20poly1305.c:78 [inline]
         poly_genkey_done+0x15/0x30 crypto/chacha20poly1305.c:369
         cryptd_skcipher_complete+0x29/0x70 crypto/cryptd.c:279
         cryptd_skcipher_decrypt+0xcd/0x110 crypto/cryptd.c:339
         cryptd_queue_worker+0x70/0xa0 crypto/cryptd.c:184
         process_one_work+0x1ed/0x420 kernel/workqueue.c:2269
         worker_thread+0x3e/0x3a0 kernel/workqueue.c:2415
         kthread+0x11f/0x140 kernel/kthread.c:255
         ret_from_fork+0x1f/0x30 arch/x86/entry/entry_64.S:352

    Fixes: 71ebc4d1b27d ("crypto: chacha20poly1305 - Add a ChaCha20-Poly1305 AEAD
construction, RFC7539")
    Cc: <stable@vger.kernel.org> # v4.2+
    Cc: Martin Willi <martin@strongswan.org>
    Signed-off-by: Eric Biggers <ebiggers@google.com>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit 7bbf1e0eaeafaa45e1cd84a482f387948d57f089
Author: Elena Petrova <lenaptr@google.com>
Date:   Tue May 28 15:35:06 2019 +0100

    crypto: arm64/sha2-ce - correct digest for empty data in finup

    commit 6bd934de1e393466b319d29c4427598fda096c57 upstream.

    The sha256-ce finup implementation for ARM64 produces wrong digest
    for empty input (len=0). Expected: the actual digest, result: initial
    value of SHA internal state. The error is in sha256_ce_finup:
    for empty data `finalize` will be 1, so the code is relying on
    sha2_ce_transform to make the final round. However, in
    sha256_base_do_update, the block function will not be called when
    len == 0.

    Fix it by setting finalize to 0 if data is empty.

    Fixes: 03802f6a80b3a ("crypto: arm64/sha2-ce - move SHA-224/256 ARMv8 implementation to
base layer")
    Cc: stable@vger.kernel.org
    Signed-off-by: Elena Petrova <lenaptr@google.com>
    Reviewed-by: Ard Biesheuvel <ard.biesheuvel@linaro.org>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 48100f1cc6fa4ba7334bc8db05d28b452b6db66a
Author: Elena Petrova <lenaptr@google.com>
Date:   Tue May 28 13:41:52 2019 +0100

    crypto: arm64/sha1-ce - correct digest for empty data in finup

    commit 1d4aaf16defa86d2665ae7db0259d6cb07e2091f upstream.

    The sha1-ce finup implementation for ARM64 produces wrong digest
    for empty input (len=0). Expected: da39a3ee..., result: 67452301...
    (initial value of SHA internal state). The error is in sha1_ce_finup:
    for empty data `finalize` will be 1, so the code is relying on
    sha1_ce_transform to make the final round. However, in
    sha1_base_do_update, the block function will not be called when
    len == 0.

    Fix it by setting finalize to 0 if data is empty.

    Fixes: 07eb54d306f4 ("crypto: arm64/sha1-ce - move SHA-1 ARMv8 implementation to base
layer")
    Cc: stable@vger.kernel.org
    Signed-off-by: Elena Petrova <lenaptr@google.com>
    Reviewed-by: Ard Biesheuvel <ard.biesheuvel@linaro.org>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 0c0de675977d99c7509837161a2dfb7d94a875d2
Author: Hook, Gary <Gary.Hook@amd.com>
Date:   Thu Jun 27 16:16:23 2019 +0000

    crypto: ccp - Validate the the error value used to index error messages

    commit 52393d617af7b554f03531e6756facf2ea687d2e upstream.

    The error code read from the queue status register is only 6 bits wide,
    but we need to verify its value is within range before indexing the error
    messages.

    Fixes: 81422badb3907 ("crypto: ccp - Make syslog errors human-readable")
```

```
    Cc: <stable@vger.kernel.org>
    Reported-by: Cfir Cohen <cfir@google.com>
    Signed-off-by: Gary R Hook <gary.hook@amd.com>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 629e1cb02b042ba5840d4f744a322ddfc55ac0c4
Author: Ard Biesheuvel <ard.biesheuvel@linaro.org>
Date:   Fri May 31 10:13:06 2019 +0200

    crypto: caam - limit output IV to CBC to work around CTR mode DMA issue

    commit ed527b13d800dd515a9e6c582f0a73eca65b2e1b upstream.

    The CAAM driver currently violates an undocumented and slightly
    controversial requirement imposed by the crypto stack that a buffer
    referred to by the request structure via its virtual address may not
    be modified while any scatterlists passed via the same request
    structure are mapped for inbound DMA.

    This may result in errors like

      alg: aead: decryption failed on test 1 for gcm_base(ctr-aes-caam,ghash-generic):
ret=74
      alg: aead: Failed to load transform for gcm(aes): -2

    on non-cache coherent systems, due to the fact that the GCM driver
    passes an IV buffer by virtual address which shares a cacheline with
    the auth_tag buffer passed via a scatterlist, resulting in corruption
    of the auth_tag when the IV is updated while the DMA mapping is live.

    Since the IV that is returned to the caller is only valid for CBC mode,
    and given that the in-kernel users of CBC (such as CTS) don't trigger the
    same issue as the GCM driver, let's just disable the output IV generation
    for all modes except CBC for the time being.

    Fixes: 854b06f76879 ("crypto: caam - properly set IV after {en,de}crypt")
    Cc: Horia Geanta <horia.geanta@nxp.com>
    Cc: Iuliana Prodan <iuliana.prodan@nxp.com>
    Reported-by: Sascha Hauer <s.hauer@pengutronix.de>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Ard Biesheuvel <ard.biesheuvel@linaro.org>
    Reviewed-by: Horia Geanta <horia.geanta@nxp.com>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f15795596275c0b31058696ba497e3cb569ad57d
Author: Eric Biggers <ebiggers@google.com>
Date:   Thu May 30 10:50:39 2019 -0700

    crypto: ghash - fix unaligned memory access in ghash_setkey()

    commit 5c6bc4dfa515738149998bb0db2481a4fdead979 upstream.

    Changing ghash_mod_init() to be subsys_initcall made it start running
    before the alignment fault handler has been installed on ARM.  In kernel
    builds where the keys in the ghash test vectors happened to be
    misaligned in the kernel image, this exposed the longstanding bug that
    ghash_setkey() is incorrectly casting the key buffer (which can have any
    alignment) to be128 for passing to gf128mul_init_4k_lle().

    Fix this by memcpy()ing the key to a temporary buffer.

    Don't fix it by setting an alignmask on the algorithm instead because
    that would unnecessarily force alignment of the data too.
```

```
    Fixes: 2cdc6899a88e ("crypto: ghash - Add GHASH digest algorithm for GCM")
    Reported-by: Peter Robinson <pbrobinson@gmail.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Eric Biggers <ebiggers@google.com>
    Tested-by: Peter Robinson <pbrobinson@gmail.com>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ef15dea6849becc29591f78e1cfa8ca345e5023c
Author: Finn Thain <fthain@telegraphics.com.au>
Date:   Sun Jun 9 11:19:11 2019 +1000

    scsi: mac_scsi: Fix pseudo DMA implementation, take 2

    commit 78ff751f8e6a9446e9fb26b2bff0b8d3f8974cbd upstream.

    A system bus error during a PDMA transfer can mess up the calculation of
    the transfer residual (the PDMA handshaking hardware lacks a byte
    counter). This results in data corruption.

    The algorithm in this patch anticipates a bus error by starting each
    transfer with a MOVE.B instruction. If a bus error is caught the transfer
    will be retried. If a bus error is caught later in the transfer (for a
    MOVE.W instruction) the transfer gets failed and subsequent requests for
    that target will use PIO instead of PDMA.

    This avoids the "!REQ and !ACK" error so the severity level of that message
    is reduced to KERN_DEBUG.

    Cc: Michael Schmitz <schmitzmic@gmail.com>
    Cc: Geert Uytterhoeven <geert@linux-m68k.org>
    Cc: stable@vger.kernel.org # v4.14+
    Fixes: 3a0f64bfa907 ("mac_scsi: Fix pseudo DMA implementation")
    Signed-off-by: Finn Thain <fthain@telegraphics.com.au>
    Reported-by: Chris Jones <chris@martin-jones.com>
    Tested-by: Stan Johnson <userm57@yahoo.com>
    Tested-by: Michael Schmitz <schmitzmic@gmail.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3df47088e7ad08dd41c3da0f17fc077e320251e1
Author: Finn Thain <fthain@telegraphics.com.au>
Date:   Sun Jun 9 11:19:11 2019 +1000

    scsi: mac_scsi: Increase PIO/PDMA transfer length threshold

    commit 7398cee4c3e6aea1ba07a6449e5533ecd0b92cdd upstream.

    Some targets introduce delays when handshaking the response to certain
    commands. For example, a disk may send a 96-byte response to an INQUIRY
    command (or a 24-byte response to a MODE SENSE command) too slowly.

    Apparently the first 12 or 14 bytes are handshaked okay but then the system
    bus error timeout is reached while transferring the next word.

    Since the scsi bus phase hasn't changed, the driver then sets the target
    borken flag to prevent further PDMA transfers. The driver also logs the
    warning, "switching to slow handshake".

    Raise the PDMA threshold to 512 bytes so that PIO transfers will be used
    for these commands. This default is sufficiently low that PDMA will still
    be used for READ and WRITE commands.

    The existing threshold (16 bytes) was chosen more or less at random.
```

```
    However, best performance requires the threshold to be as low as possible.
    Those systems that don't need the PIO workaround at all may benefit from
    mac_scsi.setup_use_pdma=1

    Cc: Michael Schmitz <schmitzmic@gmail.com>
    Cc: stable@vger.kernel.org # v4.14+
    Fixes: 3a0f64bfa907 ("mac_scsi: Fix pseudo DMA implementation")
    Signed-off-by: Finn Thain <fthain@telegraphics.com.au>
    Tested-by: Stan Johnson <userm57@yahoo.com>
    Tested-by: Michael Schmitz <schmitzmic@gmail.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 70644381cd5b6bbe2b5fc3d71d61a74c53755c09
Author: Shivasharan S <shivasharan.srikanteshwara@broadcom.com>
Date:   Fri Jun 28 18:02:12 2019 -0700

    scsi: megaraid_sas: Fix calculation of target ID

    commit c8f96df5b8e633056b7ebf5d52a9d6fb1b156ce3 upstream.

    In megasas_get_target_prop(), driver is incorrectly calculating the target
    ID for devices with channel 1 and 3.  Due to this, firmware will either
    fail the command (if there is no device with the target id sent from
    driver) or could return the properties for a target which was not
    intended.  Devices could end up with the wrong queue depth due to this.

    Fix target id calculation for channel 1 and 3.

    Fixes: 96188a89cc6d ("scsi: megaraid_sas: NVME interface target prop added")
    Cc: stable@vger.kernel.org
    Signed-off-by: Shivasharan S <shivasharan.srikanteshwara@broadcom.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ab5599830422c2a95cf665f7f4549277dfff33aa
Author: Benjamin Block <bblock@linux.ibm.com>
Date:   Tue Jul 2 23:02:01 2019 +0200

    scsi: zfcp: fix request object use-after-free in send path causing wrong traces

    commit 106d45f350c7cac876844dc685845cba4ffdb70b upstream.

    When tracing instances where we open and close WKA ports, we also pass the
    request-ID of the respective FSF command.

    But after successfully sending the FSF command we must not use the
    request-object anymore, as this might result in an use-after-free (see
    "zfcp: fix request object use-after-free in send path causing seqno
    errors" ).

    To fix this add a new variable that caches the request-ID before sending
    the request. This won't change during the hand-off to the FCP channel,
    and so it's safe to trace this cached request-ID later, instead of using
    the request object.

    Signed-off-by: Benjamin Block <bblock@linux.ibm.com>
    Fixes: d27a7cb91960 ("zfcp: trace on request for open and close of WKA port")
    Cc: <stable@vger.kernel.org> #2.6.38+
    Reviewed-by: Steffen Maier <maier@linux.ibm.com>
    Reviewed-by: Jens Remus <jremus@linux.ibm.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ef1f26bba1a16344c0829efe7c22627a98f25e0e
```

```
Author: Benjamin Block <bblock@linux.ibm.com>
Date:   Tue Jul 2 23:02:00 2019 +0200

    scsi: zfcp: fix request object use-after-free in send path causing seqno errors

    commit b76becde2b84137faa29bbc9a3b20953b5980e48 upstream.

    With a recent change to our send path for FSF commands we introduced a
    possible use-after-free of request-objects, that might further lead to
    zfcp crafting bad requests, which the FCP channel correctly complains
    about with an error (FSF_PROT_SEQ_NUMB_ERROR). This error is then handled
    by an adapter-wide recovery.

    The following sequence illustrates the possible use-after-free:

        Send Path:

            int zfcp_fsf_open_port(struct zfcp_erp_action *erp_action)
            {
                    struct zfcp_fsf_req *req;
                    ...
                    spin_lock_irq(&qdio->req_q_lock);
            //                    ^^^^^^^^^^^^^^^^
            //                    protects QDIO queue during sending
                    ...
                    req = zfcp_fsf_req_create(qdio,
                                              FSF_QTCB_OPEN_PORT_WITH_DID,
                                              SBAL_SFLAGS0_TYPE_READ,
                                              qdio->adapter->pool.erp_req);
            //            ^^^^^^^^^^^^^^^^^^^
            //            allocation of the request-object
                    ...
                    retval = zfcp_fsf_req_send(req);
                    ...
                    spin_unlock_irq(&qdio->req_q_lock);
                    return retval;
            }

            static int zfcp_fsf_req_send(struct zfcp_fsf_req *req)
            {
                    struct zfcp_adapter *adapter = req->adapter;
                    struct zfcp_qdio *qdio = adapter->qdio;
                    ...
                    zfcp_reqlist_add(adapter->req_list, req);
            //      ^^^^^^^^^^^^^^^^
            //      add request to our driver-internal hash-table for tracking
            //      (protected by separate lock req_list->lock)
                    ...
                    if (zfcp_qdio_send(qdio, &req->qdio_req)) {
            //          ^^^^^^^^^^^^^^
            //          hand-off the request to FCP channel;
            //          the request can complete at any point now
                            ...
                    }

                    /* Don't increase for unsolicited status */
                    if (!zfcp_fsf_req_is_status_read_buffer(req))
            //           ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
            //           possible use-after-free
                       adapter->fsf_req_seq_no++;
            //                     ^^^^^^^^^^^^^^^^
            //                     because of the use-after-free we might
            //                     miss this accounting, and as follow-up
            //                     this results in the FCP channel error
            //                     FSF_PROT_SEQ_NUMB_ERROR
```

```
                adapter->req_no++;

                return 0;
        }

        static inline bool
        zfcp_fsf_req_is_status_read_buffer(struct zfcp_fsf_req *req)
        {
                return req->qtcb == NULL;
        //             ^^^^^^^^^
        //             possible use-after-free
        }
```

Response Path:

```
        void zfcp_fsf_reqid_check(struct zfcp_qdio *qdio, int sbal_idx)
        {
                ...
                struct zfcp_fsf_req *fsf_req;
                ...
                for (idx = 0; idx < QDIO_MAX_ELEMENTS_PER_BUFFER; idx++) {
                        ...
                        fsf_req = zfcp_reqlist_find_rm(adapter->req_list,
                                                       req_id);
        //                       ^^^^^^^^^^^^^^^^^^^^
        //                       remove request from our driver-internal
        //                       hash-table (lock req_list->lock)
                        ...
                        zfcp_fsf_req_complete(fsf_req);
                }
        }

        static void zfcp_fsf_req_complete(struct zfcp_fsf_req *req)
        {
                ...
                if (likely(req->status & ZFCP_STATUS_FSFREQ_CLEANUP))
                        zfcp_fsf_req_free(req);
        //              ^^^^^^^^^^^^^^^^^
        //              free memory for request-object
                else
                        complete(&req->completion);
        //              ^^^^^^^^
        //              completion notification for code-paths that wait
        //              synchronous for the completion of the request; in
        //              those the memory is freed separately
        }
```

The result of the use-after-free only affects the send path, and can not lead to any data corruption. In case we miss the sequence-number accounting, because the memory was already re-purposed, the next FSF command will fail with said FCP channel error, and we will recover the whole adapter. This causes no additional errors, but it slows down traffic. There is a slight chance of the same thing happen again recursively after the adapter recovery, but so far this has not been seen.

This was seen under z/VM, where the send path might run on a virtual CPU that gets scheduled away by z/VM, while the return path might still run, and so create the necessary timing. Running with KASAN can also slow down the kernel sufficiently to run into this user-after-free, and then see the report by KASAN.

To fix this, simply pull the test for the sequence-number accounting in front of the hand-off to the FCP channel (this information doesn't change during hand-off), but leave the sequence-number accounting itself where it is.

```
    To make future regressions of the same kind less likely, add comments to
    all closely related code-paths.

    Signed-off-by: Benjamin Block <bblock@linux.ibm.com>
    Fixes: f9eca0227600 ("scsi: zfcp: drop duplicate fsf_command from zfcp_fsf_req which is
also in QTCB header")
    Cc: <stable@vger.kernel.org> #5.0+
    Reviewed-by: Steffen Maier <maier@linux.ibm.com>
    Reviewed-by: Jens Remus <jremus@linux.ibm.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ecc31dda6facb91ec484a7d368d0c48c6cb9e8f6
Author: Damien Le Moal <damien.lemoal@wdc.com>
Date:   Wed Jul 17 10:51:49 2019 +0900

    scsi: sd_zbc: Fix compilation warning

    commit 0cdc58580b37a160fac4b884266b8b7cb096f539 upstream.

    kbuild test robot gets the following compilation warning using gcc 7.4
    cross compilation for c6x (GCC_VERSION=7.4.0 make.cross ARCH=c6x).

       In file included from include/asm-generic/bug.h:18:0,
                        from arch/c6x/include/asm/bug.h:12,
                        from include/linux/bug.h:5,
                        from include/linux/thread_info.h:12,
                        from include/asm-generic/current.h:5,
                        from ./arch/c6x/include/generated/asm/current.h:1,
                        from include/linux/sched.h:12,
                        from include/linux/blkdev.h:5,
                        from drivers//scsi/sd_zbc.c:11:
       drivers//scsi/sd_zbc.c: In function 'sd_zbc_read_zones':
    >> include/linux/kernel.h:62:48: warning: 'zone_blocks' may be used
       uninitialized in this function [-Wmaybe-uninitialized]
        #define __round_mask(x, y) ((__typeof__(x))((y)-1))
                                                    ^
       drivers//scsi/sd_zbc.c:464:6: note: 'zone_blocks' was declared here
         u32 zone_blocks;
             ^~~~~~~~~~~

    This is a false-positive report. The variable zone_blocks is always
    initialized in sd_zbc_check_zones() before use. It is not initialized
    only and only if sd_zbc_check_zones() fails.

    Avoid this warning by initializing the zone_blocks variable to 0.

    Fixes: 5f832a395859 ("scsi: sd_zbc: Fix sd_zbc_check_zones() error checks")
    Cc: Stable <stable@vger.kernel.org>
    Signed-off-by: Damien Le Moal <damien.lemoal@wdc.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 80d4218201f4d6bcd875af30f3e48b4b974fd244
Author: Ming Lei <ming.lei@redhat.com>
Date:   Fri Jul 12 10:08:19 2019 +0800

    scsi: core: Fix race on creating sense cache

    commit f9b0530fa02e0c73f31a49ef743e8f44eb8e32cc upstream.

    When scsi_init_sense_cache(host) is called concurrently from different
    hosts, each code path may find that no cache has been created and
    allocate a new one. The lack of locking can lead to potentially
```

```
    overriding a cache allocated by a different host.

    Fix the issue by moving 'mutex_lock(&scsi_sense_cache_mutex)' before
    scsi_select_sense_cache().

    Fixes: 0a6ac4ee7c21 ("scsi: respect unchecked_isa_dma for blk-mq")
    Cc: Stable <stable@vger.kernel.org>
    Cc: Christoph Hellwig <hch@lst.de>
    Cc: Hannes Reinecke <hare@suse.com>
    Cc: Ewan D. Milne <emilne@redhat.com>
    Signed-off-by: Ming Lei <ming.lei@redhat.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b2f215376e3bdb437cc714cb568016f60dc896f6
Author: Finn Thain <fthain@telegraphics.com.au>
Date:   Sun Jun 9 11:19:11 2019 +1000

    Revert "scsi: ncr5380: Increase register polling limit"

    commit 25fcf94a2fa89dd3e73e965ebb0b38a2a4f72aa4 upstream.

    This reverts commit 4822827a69d7cd3bc5a07b7637484ebd2cf88db6.

    The purpose of that commit was to suppress a timeout warning message which
    appeared to be caused by target latency. But suppressing the warning is
    undesirable as the warning may indicate a messed up transfer count.

    Another problem with that commit is that 15 ms is too long to keep
    interrupts disabled as interrupt latency can cause system clock drift and
    other problems.

    Cc: Michael Schmitz <schmitzmic@gmail.com>
    Cc: stable@vger.kernel.org
    Fixes: 4822827a69d7 ("scsi: ncr5380: Increase register polling limit")
    Signed-off-by: Finn Thain <fthain@telegraphics.com.au>
    Tested-by: Stan Johnson <userm57@yahoo.com>
    Tested-by: Michael Schmitz <schmitzmic@gmail.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b9206a75c021cf0a206c12830ea62f6e515ffb97
Author: Finn Thain <fthain@telegraphics.com.au>
Date:   Sun Jun 9 11:19:11 2019 +1000

    scsi: NCR5380: Handle PDMA failure reliably

    commit f9dfed1c785734b95b08d67600e05d2092508ab0 upstream.

    A PDMA error is handled in the core driver by setting the device's 'borken'
    flag and aborting the command. Unfortunately, do_abort() is not
    dependable. Perform a SCSI bus reset instead, to make sure that the command
    fails and gets retried.

    Cc: Michael Schmitz <schmitzmic@gmail.com>
    Cc: stable@vger.kernel.org # v4.20+
    Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
    Signed-off-by: Finn Thain <fthain@telegraphics.com.au>
    Tested-by: Stan Johnson <userm57@yahoo.com>
    Tested-by: Michael Schmitz <schmitzmic@gmail.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit bd91d5ceb8ade3c85ca4f0ce2b9ff5cfdedeab43
Author: Finn Thain <fthain@telegraphics.com.au>
```

```
Date:   Sun Jun 9 11:19:11 2019 +1000

    scsi: NCR5380: Always re-enable reselection interrupt

    commit 57f31326518e98ee4cabf9a04efe00ed57c54147 upstream.

    The reselection interrupt gets disabled during selection and must be
    re-enabled when hostdata->connected becomes NULL. If it isn't re-enabled a
    disconnected command may time-out or the target may wedge the bus while
    trying to reselect the host. This can happen after a command is aborted.

    Fix this by enabling the reselection interrupt in NCR5380_main() after
    calls to NCR5380_select() and NCR5380_information_transfer() return.

    Cc: Michael Schmitz <schmitzmic@gmail.com>
    Cc: stable@vger.kernel.org # v4.9+
    Fixes: 8b00c3d5d40d ("ncr5380: Implement new eh_abort_handler")
    Signed-off-by: Finn Thain <fthain@telegraphics.com.au>
    Tested-by: Stan Johnson <userm57@yahoo.com>
    Tested-by: Michael Schmitz <schmitzmic@gmail.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 6ef36ab967c71690ebe7e5ef997a8be4da3bc844
Author: Juergen Gross <jgross@suse.com>
Date:   Wed Jun 19 11:00:56 2019 +0200

    xen: let alloc_xenballooned_pages() fail if not enough memory free

    commit a1078e821b605813b63bf6bca414a85f804d5c66 upstream.

    Instead of trying to allocate pages with GFP_USER in
    add_ballooned_pages() check the available free memory via
    si_mem_available(). GFP_USER is far less limiting memory exhaustion
    than the test via si_mem_available().

    This will avoid dom0 running out of memory due to excessive foreign
    page mappings especially on ARM and on x86 in PVH mode, as those don't
    have a pre-ballooned area which can be used for foreign mappings.

    As the normal ballooning suffers from the same problem don't balloon
    down more than si_mem_available() pages in one iteration. At the same
    time limit the default maximum number of retries.

    This is part of XSA-300.

    Signed-off-by: Juergen Gross <jgross@suse.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit d39c2e97277229970fe2ae56dcbf67a535e14873
Author: Denis Efremov <efremov@ispras.ru>
Date:   Fri Jul 12 21:55:23 2019 +0300

    floppy: fix out-of-bounds read in copy_buffer

    [ Upstream commit da99466ac243f15fbba65bd261bfc75ffa1532b6 ]

    This fixes a global out-of-bounds read access in the copy_buffer
    function of the floppy driver.

    The FDDEFPRM ioctl allows one to set the geometry of a disk.  The sect
    and head fields (unsigned int) of the floppy_drive structure are used to
    compute the max_sector (int) in the make_raw_rw_request function.  It is
    possible to overflow the max_sector.  Next, max_sector is passed to the
    copy_buffer function and used in one of the memcpy calls.
```

```
    An unprivileged user could trigger the bug if the device is accessible,
    but requires a floppy disk to be inserted.

    The patch adds the check for the .sect * .head multiplication for not
    overflowing in the set_geometry function.

    The bug was found by syzkaller.

    Signed-off-by: Denis Efremov <efremov@ispras.ru>
    Tested-by: Willy Tarreau <w@1wt.eu>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0f2e88b70018f3894a03edaf5f7d051eac92a074
Author: Denis Efremov <efremov@ispras.ru>
Date:   Fri Jul 12 21:55:22 2019 +0300

    floppy: fix invalid pointer dereference in drive_name

    [ Upstream commit 9b04609b784027968348796a18f601aed9db3789 ]

    This fixes the invalid pointer dereference in the drive_name function of
    the floppy driver.

    The native_format field of the struct floppy_drive_params is used as
    floppy_type array index in the drive_name function.  Thus, the field
    should be checked the same way as the autodetect field.

    To trigger the bug, one could use a value out of range and set the drive
    parameters with the FDSETDRVPRM ioctl.  Next, FDGETDRVTYP ioctl should
    be used to call the drive_name.  A floppy disk is not required to be
    inserted.

    CAP_SYS_ADMIN is required to call FDSETDRVPRM.

    The patch adds the check for a value of the native_format field to be in
    the '0 <= x < ARRAY_SIZE(floppy_type)' range of the floppy_type array
    indices.

    The bug was found by syzkaller.

    Signed-off-by: Denis Efremov <efremov@ispras.ru>
    Tested-by: Willy Tarreau <w@1wt.eu>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ad839534e9df9bf45c32a8a035d2b9277d56c2ce
Author: Denis Efremov <efremov@ispras.ru>
Date:   Fri Jul 12 21:55:21 2019 +0300

    floppy: fix out-of-bounds read in next_valid_format

    [ Upstream commit 5635f897ed83fd539df78e98ba69ee91592f9bb8 ]

    This fixes a global out-of-bounds read access in the next_valid_format
    function of the floppy driver.

    The values from autodetect field of the struct floppy_drive_params are
    used as indices for the floppy_type array in the next_valid_format
    function 'floppy_type[DP->autodetect[probed_format]].sect'.

    To trigger the bug, one could use a value out of range and set the drive
    parameters with the FDSETDRVPRM ioctl.  A floppy disk is not required to
    be inserted.
```

```
    CAP_SYS_ADMIN is required to call FDSETDRVPRM.

    The patch adds the check for values of the autodetect field to be in the
    '0 <= x < ARRAY_SIZE(floppy_type)' range of the floppy_type array indices.

    The bug was found by syzkaller.

    Signed-off-by: Denis Efremov <efremov@ispras.ru>
    Tested-by: Willy Tarreau <w@1wt.eu>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 697c0af7468a941522c1e26345aa5128fa2a4815
Author: Denis Efremov <efremov@ispras.ru>
Date:   Fri Jul 12 21:55:20 2019 +0300

    floppy: fix div-by-zero in setup_format_params

    [ Upstream commit f3554aeb991214cbfafd17d55e2bfddb50282e32 ]

    This fixes a divide by zero error in the setup_format_params function of
    the floppy driver.

    Two consecutive ioctls can trigger the bug: The first one should set the
    drive geometry with such .sect and .rate values for the F_SECT_PER_TRACK
    to become zero.  Next, the floppy format operation should be called.

    A floppy disk is not required to be inserted.  An unprivileged user
    could trigger the bug if the device is accessible.

    The patch checks F_SECT_PER_TRACK for a non-zero value in the
    set_geometry function.  The proper check should involve a reasonable
    upper limit for the .sect and .rate fields, but it could change the
    UAPI.

    The patch also checks F_SECT_PER_TRACK in the setup_format_params, and
    cancels the formatting operation in case of zero.

    The bug was found by syzkaller.

    Signed-off-by: Denis Efremov <efremov@ispras.ru>
    Tested-by: Willy Tarreau <w@1wt.eu>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d092fcf1d773950aa06253ebe933966fa04fc41a
Author: Andrii Nakryiko <andriin@fb.com>
Date:   Mon Jul 15 20:57:03 2019 -0700

    libbpf: fix another GCC8 warning for strncpy

    [ Upstream commit 763ff0e7d9c72e7094b31e7fb84a859be9325635 ]

    Similar issue was fixed in cdfc7f888c2a ("libbpf: fix GCC8 warning for
    strncpy") already. This one was missed. Fixing now.

    Cc: Magnus Karlsson <magnus.karlsson@intel.com>
    Signed-off-by: Andrii Nakryiko <andriin@fb.com>
    Acked-by: Magnus Karlsson <magnus.karlsson@intel.com>
    Signed-off-by: Alexei Starovoitov <ast@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dc2fc59d289c4563fad7ee6d37dd4ae768f41936
Author: Dennis Zhou <dennis@kernel.org>
```

```
Date:   Fri Jul 5 17:09:09 2019 -0400

    blk-iolatency: fix STS_AGAIN handling

    [ Upstream commit c9b3007feca018d3f7061f5d5a14cb00766ffe9b ]

    The iolatency controller is based on rq_qos. It increments on
    rq_qos_throttle() and decrements on either rq_qos_cleanup() or
    rq_qos_done_bio(). a3fb01ba5af0 fixes the double accounting issue where
    blk_mq_make_request() may call both rq_qos_cleanup() and
    rq_qos_done_bio() on REQ_NO_WAIT. So checking STS_AGAIN prevents the
    double decrement.

    The above works upstream as the only way we can get STS_AGAIN is from
    blk_mq_get_request() failing. The STS_AGAIN handling isn't a real
    problem as bio_endio() skipping only happens on reserved tag allocation
    failures which can only be caused by driver bugs and already triggers
    WARN.

    However, the fix creates a not so great dependency on how STS_AGAIN can
    be propagated. Internally, we (Facebook) carry a patch that kills read
    ahead if a cgroup is io congested or a fatal signal is pending. This
    combined with chained bios progagate their bi_status to the parent is
    not already set can can cause the parent bio to not clean up properly
    even though it was successful. This consequently leaks the inflight
    counter and can hang all IOs under that blkg.

    To nip the adverse interaction early, this removes the rq_qos_cleanup()
    callback in iolatency in favor of cleaning up always on the
    rq_qos_done_bio() path.

    Fixes: a3fb01ba5af0 ("blk-iolatency: only account submitted bios")
    Debugged-by: Tejun Heo <tj@kernel.org>
    Debugged-by: Josef Bacik <josef@toxicpanda.com>
    Signed-off-by: Dennis Zhou <dennis@kernel.org>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 524659fbd39f5e39fea9fd8ee0c141bd85bc89bb
Author: Colin Ian King <colin.king@canonical.com>
Date:   Wed Jun 19 15:30:44 2019 +0100

    iavf: fix dereference of null rx_buffer pointer

    [ Upstream commit 9fe06a51287b2d41baef7ece94df34b5abf19b90 ]

    A recent commit efa14c3985828d ("iavf: allow null RX descriptors") added
    a null pointer sanity check on rx_buffer, however, rx_buffer is being
    dereferenced before that check, which implies a null pointer dereference
    bug can potentially occur.  Fix this by only dereferencing rx_buffer
    until after the null pointer check.

    Addresses-Coverity: ("Dereference before null check")
    Signed-off-by: Colin Ian King <colin.king@canonical.com>
    Tested-by: Andrew Bowers <andrewx.bowers@intel.com>
    Signed-off-by: Jeff Kirsher <jeffrey.t.kirsher@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9c1de3cca1d4187e09f5c7a4143bad3c150c95c3
Author: Huazhong Tan <tanhuazhong@huawei.com>
Date:   Fri Jun 28 19:50:07 2019 +0800

    net: hns3: fix __QUEUE_STATE_STACK_XOFF not cleared issue

    [ Upstream commit f96315f2f17e7b2580d2fec7c4d6a706a131d904 ]
```

```
    When change MTU or other operations, which just calling .reset_notify
    to do HNAE3_DOWN_CLIENT and HNAE3_UP_CLIENT, then
    the netdev_tx_reset_queue() in the hns3_clear_all_ring() will be
    ignored. So the dev_watchdog() may misdiagnose a TX timeout.

    This patch separates netdev_tx_reset_queue() from
    hns3_clear_all_ring(), and unifies hns3_clear_all_ring() and
    hns3_force_clear_all_ring into one, since they are doing
    similar things.

    Fixes: 3a30964a2eef ("net: hns3: delay ring buffer clearing during reset")
    Signed-off-by: Huazhong Tan <tanhuazhong@huawei.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 34dd96bd7b5fa59629446704cbf444b787024846
Author: Josua Mayer <josua@solid-run.com>
Date:   Tue Jul 9 15:01:01 2019 +0200

    net: mvmdio: defer probe of orion-mdio if a clock is not ready

    [ Upstream commit 433a06d7d74e677c40b1148c70c48677ff62fb6b ]

    Defer probing of the orion-mdio interface when getting a clock returns
    EPROBE_DEFER. This avoids locking up the Armada 8k SoC when mdio is used
    before all clocks have been enabled.

    Signed-off-by: Josua Mayer <josua@solid-run.com>
    Reviewed-by: Andrew Lunn <andrew@lunn.ch>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8a090e3b73eaffe18e08ccc3fb5abecf6b0a9781
Author: Ilya Maximets <i.maximets@samsung.com>
Date:   Wed Jul 3 15:09:16 2019 +0300

    xdp: fix race on generic receive path

    [ Upstream commit bf0bdd1343efbbf65b4d53aef1fce14acbd79d50 ]

    Unlike driver mode, generic xdp receive could be triggered
    by different threads on different CPU cores at the same time
    leading to the fill and rx queue breakage. For example, this
    could happen while sending packets from two processes to the
    first interface of veth pair while the second part of it is
    open with AF_XDP socket.

    Need to take a lock for each generic receive to avoid race.

    Fixes: c497176cb2e4 ("xsk: add Rx receive functions and poll support")
    Signed-off-by: Ilya Maximets <i.maximets@samsung.com>
    Acked-by: Magnus Karlsson <magnus.karlsson@intel.com>
    Tested-by: William Tu <u9012063@gmail.com>
    Signed-off-by: Daniel Borkmann <daniel@iogearbox.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 549348f1db00a044658b2574657f590920305ac6
Author: Taehee Yoo <ap420073@gmail.com>
Date:   Wed Jul 3 00:23:42 2019 +0900

    gtp: fix use-after-free in gtp_newlink()

    [ Upstream commit a2bed90704c68d3763bf24decb1b781a45395de8 ]
```

```
    Current gtp_newlink() could be called after unregister_pernet_subsys().
    gtp_newlink() uses gtp_net but it can be destroyed by
    unregister_pernet_subsys().
    So unregister_pernet_subsys() should be called after
    rtnl_link_unregister().

    Test commands:
       #SHELL 1
       while :
       do
               for i in {1..5}
               do
                    ./gtp-link add gtp$i &
               done
               killall gtp-link
       done

       #SHELL 2
       while :
       do
            modprobe -rv gtp
       done

    Splat looks like:
    [  753.176631] BUG: KASAN: use-after-free in gtp_newlink+0x9b4/0xa5c [gtp]
    [  753.177722] Read of size 8 at addr ffff8880d48f2458 by task gtp-link/7126
    [  753.179082] CPU: 0 PID: 7126 Comm: gtp-link Tainted: G        W         5.2.0-rc6+
#50
    [  753.185801] Call Trace:
    [  753.186264]  dump_stack+0x7c/0xbb
    [  753.186863]  ? gtp_newlink+0x9b4/0xa5c [gtp]
    [  753.187583]  print_address_description+0xc7/0x240
    [  753.188382]  ? gtp_newlink+0x9b4/0xa5c [gtp]
    [  753.189097]  ? gtp_newlink+0x9b4/0xa5c [gtp]
    [  753.189846]  __kasan_report+0x12a/0x16f
    [  753.190542]  ? gtp_newlink+0x9b4/0xa5c [gtp]
    [  753.191298]  kasan_report+0xe/0x20
    [  753.191893]  gtp_newlink+0x9b4/0xa5c [gtp]
    [  753.192580]  ? __netlink_ns_capable+0xc3/0xf0
    [  753.193370]  __rtnl_newlink+0xb9f/0x11b0
    [ ... ]
    [  753.241201] Allocated by task 7186:
    [  753.241844]  save_stack+0x19/0x80
    [  753.242399]  __kasan_kmalloc.constprop.3+0xa0/0xd0
    [  753.243192]  __kmalloc+0x13e/0x300
    [  753.243764]  ops_init+0xd6/0x350
    [  753.244314]  register_pernet_operations+0x249/0x6f0
    [ ... ]
    [  753.251770] Freed by task 7178:
    [  753.252288]  save_stack+0x19/0x80
    [  753.252833]  __kasan_slab_free+0x111/0x150
    [  753.253962]  kfree+0xc7/0x280
    [  753.254509]  ops_free_list.part.11+0x1c4/0x2d0
    [  753.255241]  unregister_pernet_operations+0x262/0x390
    [ ... ]
    [  753.285883] list_add corruption. next->prev should be prev (ffff8880d48f2458), but
was ffff8880d497d878. (next.
    [  753.287241] ------------[ cut here ]------------
    [  753.287794] kernel BUG at lib/list_debug.c:25!
    [  753.288364] invalid opcode: 0000 [#1] SMP DEBUG_PAGEALLOC KASAN PTI
    [  753.289099] CPU: 0 PID: 7126 Comm: gtp-link Tainted: G    B   W         5.2.0-rc6+
#50
    [  753.291036] RIP: 0010:__list_add_valid+0x74/0xd0
    [  753.291589] Code: 48 39 da 75 27 48 39 f5 74 36 48 39 dd 74 31 48 83 c4 08 b8 01 00
00 00 5b 5d c3 48 89 d9 48b
```

```
     [  753.293779] RSP: 0018:ffff8880cae8f398 EFLAGS: 00010286
     [  753.294401] RAX: 0000000000000075 RBX: ffff8880d497d878 RCX: 0000000000000000
     [  753.296260] RDX: 0000000000000075 RSI: 0000000000000008 RDI: ffffed10195d1e69
     [  753.297070] RBP: ffff8880cd250ae0 R08: ffffed101b4bff21 R09: ffffed101b4bff21
     [  753.297899] R10: 0000000000000001 R11: ffffed101b4bff20 R12: ffff8880d497d878
     [  753.298703] R13: 0000000000000000 R14: ffff8880cd250ae0 R15: ffff8880d48f2458
     [  753.299564] FS:  00007f5f79805740(0000) GS:ffff8880da400000(0000)
 knlGS:0000000000000000
     [  753.300533] CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
     [  753.301231] CR2: 00007fe8c7ef4f10 CR3: 00000000b71a6006 CR4: 00000000000606f0
     [  753.302183] Call Trace:
     [  753.302530]  gtp_newlink+0x5f6/0xa5c [gtp]
     [  753.303037]  ? __netlink_ns_capable+0xc3/0xf0
     [  753.303576]  __rtnl_newlink+0xb9f/0x11b0
     [  753.304092]  ? rtnl_link_unregister+0x230/0x230

     Fixes: 459aa660eb1d ("gtp: add initial driver for datapath of GPRS Tunneling Protocol
 (GTP-U)")
     Signed-off-by: Taehee Yoo <ap420073@gmail.com>
     Signed-off-by: David S. Miller <davem@davemloft.net>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit 98446288d079418ea5a2afd0ea61d5cc025eae2e
 Author: Taehee Yoo <ap420073@gmail.com>
 Date:   Wed Jul 3 00:22:25 2019 +0900

     gtp: fix use-after-free in gtp_encap_destroy()

     [ Upstream commit 1788b8569f5de27da09087fa3f6580d2aa04cc75 ]

     gtp_encap_destroy() is called twice.
     1. When interface is deleted.
     2. When udp socket is destroyed.
     either gtp->sk0 or gtp->sk1u could be freed by sock_put() in
     gtp_encap_destroy(). so, when gtp_encap_destroy() is called again,
     it would uses freed sk pointer.

     patch makes gtp_encap_destroy() to set either gtp->sk0 or gtp->sk1u to
     null. in addition, both gtp->sk0 and gtp->sk1u pointer are protected
     by rtnl_lock. so, rtnl_lock() is added.

     Test command:
        gtp-link add gtp1 &
        killall gtp-link
        ip link del gtp1

     Splat looks like:
     [   83.182767] BUG: KASAN: use-after-free in __lock_acquire+0x3a20/0x46a0
     [   83.184128] Read of size 8 at addr ffff8880cc7d5360 by task ip/1008
     [   83.185567] CPU: 1 PID: 1008 Comm: ip Not tainted 5.2.0-rc6+ #50
     [   83.188469] Call Trace:
     [ ... ]
     [   83.200126]  lock_acquire+0x141/0x380
     [   83.200575]  ? lock_sock_nested+0x3a/0xf0
     [   83.201069]  _raw_spin_lock_bh+0x38/0x70
     [   83.201551]  ? lock_sock_nested+0x3a/0xf0
     [   83.202044]  lock_sock_nested+0x3a/0xf0
     [   83.202520]  gtp_encap_destroy+0x18/0xe0 [gtp]
     [   83.203065]  gtp_encap_disable.isra.14+0x13/0x50 [gtp]
     [   83.203687]  gtp_dellink+0x56/0x170 [gtp]
     [   83.204190]  rtnl_delete_link+0xb4/0x100
     [ ... ]
     [   83.236513] Allocated by task 976:
     [   83.236925]  save_stack+0x19/0x80
     [   83.237332]  __kasan_kmalloc.constprop.3+0xa0/0xd0
```

```
      [   83.237894]  kmem_cache_alloc+0xd8/0x280
      [   83.238360]  sk_prot_alloc.isra.42+0x50/0x200
      [   83.238874]  sk_alloc+0x32/0x940
      [   83.239264]  inet_create+0x283/0xc20
      [   83.239684]  __sock_create+0x2dd/0x540
      [   83.240136]  __sys_socket+0xca/0x1a0
      [   83.240550]  __x64_sys_socket+0x6f/0xb0
      [   83.240998]  do_syscall_64+0x9c/0x450
      [   83.241466]  entry_SYSCALL_64_after_hwframe+0x49/0xbe
      [   83.242061]
      [   83.242249] Freed by task 0:
      [   83.242616]  save_stack+0x19/0x80
      [   83.243013]  __kasan_slab_free+0x111/0x150
      [   83.243498]  kmem_cache_free+0x89/0x250
      [   83.244444]  __sk_destruct+0x38f/0x5a0
      [   83.245366]  rcu_core+0x7e9/0x1c20
      [   83.245766]  __do_softirq+0x213/0x8fa

      Fixes: 1e3a3abd8b28 ("gtp: make GTP sockets in gtp_newlink optional")
      Signed-off-by: Taehee Yoo <ap420073@gmail.com>
      Signed-off-by: David S. Miller <davem@davemloft.net>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 972c6e23fa38ac4ee2e2ac9ec8b3f716e5b8652e
  Author: Taehee Yoo <ap420073@gmail.com>
  Date:   Wed Jul 3 00:23:13 2019 +0900

      gtp: fix Illegal context switch in RCU read-side critical section.

      [ Upstream commit 3f167e1921865b379a9becf03828e7202c7b4917 ]

      ipv4_pdp_add() is called in RCU read-side critical section.
      So GFP_KERNEL should not be used in the function.
      This patch make ipv4_pdp_add() to use GFP_ATOMIC instead of GFP_KERNEL.

      Test commands:
      gtp-link add gtp1 &
      gtp-tunnel add gtp1 v1 100 200 1.1.1.1 2.2.2.2

      Splat looks like:
      [  130.618881] =============================
      [  130.626382] WARNING: suspicious RCU usage
      [  130.626994] 5.2.0-rc6+ #50 Not tainted
      [  130.627622] -----------------------------
      [  130.628223] ./include/linux/rcupdate.h:266 Illegal context switch in RCU read-side
  critical section!
      [  130.629684]
      [  130.629684] other info that might help us debug this:
      [  130.629684]
      [  130.631022]
      [  130.631022] rcu_scheduler_active = 2, debug_locks = 1
      [  130.632136] 4 locks held by gtp-tunnel/1025:
      [  130.632925]  #0: 000000002b93c8b7 (cb_lock){++++}, at: genl_rcv+0x15/0x40
      [  130.634159]  #1: 00000000f17bc999 (genl_mutex){+.+.}, at: genl_rcv_msg+0xfb/0x130
      [  130.635487]  #2: 00000000c644ed8e (rtnl_mutex){+.+.}, at:
  gtp_genl_new_pdp+0x18c/0x1150 [gtp]
      [  130.636936]  #3: 0000000007a1cde7 (rcu_read_lock){....}, at:
  gtp_genl_new_pdp+0x187/0x1150 [gtp]
      [  130.638348]
      [  130.638348] stack backtrace:
      [  130.639062] CPU: 1 PID: 1025 Comm: gtp-tunnel Not tainted 5.2.0-rc6+ #50
      [  130.641318] Call Trace:
      [  130.641707]  dump_stack+0x7c/0xbb
      [  130.642252]  ___might_sleep+0x2c0/0x3b0
      [  130.642862]  kmem_cache_alloc_trace+0x1cd/0x2b0
```

```
    [  130.643591]  gtp_genl_new_pdp+0x6c5/0x1150 [gtp]
    [  130.644371]  genl_family_rcv_msg+0x63a/0x1030
    [  130.645074]  ? mutex_lock_io_nested+0x1090/0x1090
    [  130.645845]  ? genl_unregister_family+0x630/0x630
    [  130.646592]  ? debug_show_all_locks+0x2d0/0x2d0
    [  130.647293]  ? check_flags.part.40+0x440/0x440
    [  130.648099]  genl_rcv_msg+0xa3/0x130
    [ ... ]

    Fixes: 459aa660eb1d ("gtp: add initial driver for datapath of GPRS Tunneling Protocol
(GTP-U)")
    Signed-off-by: Taehee Yoo <ap420073@gmail.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 76357f65f18f180f44ccbbbf713461881d0ab219
Author: Taehee Yoo <ap420073@gmail.com>
Date:   Wed Jul 3 00:20:51 2019 +0900

    gtp: fix suspicious RCU usage

    [ Upstream commit e198987e7dd7d3645a53875151cd6f8fc425b706 ]

    gtp_encap_enable_socket() and gtp_encap_destroy() are not protected
    by rcu_read_lock(). and it's not safe to write sk->sk_user_data.
    This patch make these functions to use lock_sock() instead of
    rcu_dereference_sk_user_data().

    Test commands:
        gtp-link add gtp1

    Splat looks like:
    [   83.238315] =============================
    [   83.239127] WARNING: suspicious RCU usage
    [   83.239702] 5.2.0-rc6+ #49 Not tainted
    [   83.240268] -----------------------------
    [   83.241205] drivers/net/gtp.c:799 suspicious rcu_dereference_check() usage!
    [   83.243828]
    [   83.243828] other info that might help us debug this:
    [   83.243828]
    [   83.246325]
    [   83.246325] rcu_scheduler_active = 2, debug_locks = 1
    [   83.247314] 1 lock held by gtp-link/1008:
    [   83.248523]  #0: 0000000017772c7f (rtnl_mutex){+.+.}, at: __rtnl_newlink+0x5f5/0x11b0
    [   83.251503]
    [   83.251503] stack backtrace:
    [   83.252173] CPU: 0 PID: 1008 Comm: gtp-link Not tainted 5.2.0-rc6+ #49
    [   83.253271] Hardware name: innotek GmbH VirtualBox/VirtualBox, BIOS VirtualBox
12/01/2006
    [   83.254562] Call Trace:
    [   83.254995]  dump_stack+0x7c/0xbb
    [   83.255567]  gtp_encap_enable_socket+0x2df/0x360 [gtp]
    [   83.256415]  ? gtp_find_dev+0x1a0/0x1a0 [gtp]
    [   83.257161]  ? memset+0x1f/0x40
    [   83.257843]  gtp_newlink+0x90/0xa21 [gtp]
    [   83.258497]  ? __netlink_ns_capable+0xc3/0xf0
    [   83.259260]  __rtnl_newlink+0xb9f/0x11b0
    [   83.260022]  ? rtnl_link_unregister+0x230/0x230
    [ ... ]

    Fixes: 1e3a3abd8b28 ("gtp: make GTP sockets in gtp_newlink optional")
    Signed-off-by: Taehee Yoo <ap420073@gmail.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 33401f6ba0d9f23ca5877540375b88d7cd5b1f8f
Author: csonsino <csonsino@gmail.com>
Date:   Wed Jun 12 15:00:52 2019 -0600

    Bluetooth: validate BLE connection interval updates

    [ Upstream commit c49a8682fc5d298d44e8d911f4fa14690ea9485e ]

    Problem: The Linux Bluetooth stack yields complete control over the BLE
    connection interval to the remote device.

    The Linux Bluetooth stack provides access to the BLE connection interval
    min and max values through /sys/kernel/debug/bluetooth/hci0/
    conn_min_interval and /sys/kernel/debug/bluetooth/hci0/conn_max_interval.
    These values are used for initial BLE connections, but the remote device
    has the ability to request a connection parameter update. In the event
    that the remote side requests to change the connection interval, the Linux
    kernel currently only validates that the desired value is within the
    acceptable range in the Bluetooth specification (6 - 3200, corresponding to
    7.5ms - 4000ms). There is currently no validation that the desired value
    requested by the remote device is within the min/max limits specified in
    the conn_min_interval/conn_max_interval configurations. This essentially
    leads to Linux yielding complete control over the connection interval to
    the remote device.

    The proposed patch adds a verification step to the connection parameter
    update mechanism, ensuring that the desired value is within the min/max
    bounds of the current connection. If the desired value is outside of the
    current connection min/max values, then the connection parameter update
    request is rejected and the negative response is returned to the remote
    device. Recall that the initial connection is established using the local
    conn_min_interval/conn_max_interval values, so this allows the Linux
    administrator to retain control over the BLE connection interval.

    The one downside that I see is that the current default Linux values for
    conn_min_interval and conn_max_interval typically correspond to 30ms and
    50ms respectively. If this change were accepted, then it is feasible that
    some devices would no longer be able to negotiate to their desired
    connection interval values. This might be remedied by setting the default
    Linux conn_min_interval and conn_max_interval values to the widest
    supported range (6 - 3200 / 7.5ms - 4000ms). This could lead to the same
    behavior as the current implementation, where the remote device could
    request to change the connection interval value to any value that is
    permitted by the Bluetooth specification, and Linux would accept the
    desired value.

    Signed-off-by: Carey Sonsino <csonsino@gmail.com>
    Signed-off-by: Marcel Holtmann <marcel@holtmann.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cbdc30d7285437d18389a39ca2769687f713fe36
Author: Taehee Yoo <ap420073@gmail.com>
Date:   Wed Jul 3 00:24:04 2019 +0900

    gtp: add missing gtp_encap_disable_sock() in gtp_encap_enable()

    [ Upstream commit e30155fd23c9c141cbe7d99b786e10a83a328837 ]

    If an invalid role is sent from user space, gtp_encap_enable() will fail.
    Then, it should call gtp_encap_disable_sock() but current code doesn't.
    It makes memory leak.

    Fixes: 91ed81f9abc7 ("gtp: support SGSN-side tunnels")
    Signed-off-by: Taehee Yoo <ap420073@gmail.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
```

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

commit 684392e5bc679514d06b830a4245035ec6ffd106
Author: Dan Carpenter <dan.carpenter@oracle.com>
Date:   Thu May 16 21:24:00 2019 +0300

    Bluetooth: hidp: NUL terminate a string in the compat ioctl

    [ Upstream commit dcae9052ebb0c5b2614de620323d615fcbfda7f8 ]

    This change is similar to commit a1616a5ac99e ("Bluetooth: hidp: fix
    buffer overflow") but for the compat ioctl.  We take a string from the
    user and forgot to ensure that it's NUL terminated.

    I have also changed the strncpy() in to strscpy() in hidp_setup_hid().
    The difference is the strncpy() doesn't necessarily NUL terminate the
    destination string.  Either change would fix the problem but it's nice
    to take a belt and suspenders approach and do both.

    Signed-off-by: Dan Carpenter <dan.carpenter@oracle.com>
    Signed-off-by: Marcel Holtmann <marcel@holtmann.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 497bd28c0f4d9319c86ca2b6ea8367f5262296ff
Author: Matias Karhumaa <matias.karhumaa@gmail.com>
Date:   Tue May 21 13:07:22 2019 +0300

    Bluetooth: Check state in l2cap_disconnect_rsp

    [ Upstream commit 28261da8a26f4915aa257d12d506c6ba179d961f ]

    Because of both sides doing L2CAP disconnection at the same time, it
    was possible to receive L2CAP Disconnection Response with CID that was
    already freed. That caused problems if CID was already reused and L2CAP
    Connection Request with same CID was sent out. Before this patch kernel
    deleted channel context regardless of the state of the channel.

    Example where leftover Disconnection Response (frame #402) causes local
    device to delete L2CAP channel which was not yet connected. This in
    turn confuses remote device's stack because same CID is re-used without
    properly disconnecting.

    Btmon capture before patch:
    ** snip **
    > ACL Data RX: Handle 43 flags 0x02 dlen 8               #394 [hci1] 10.748949
          Channel: 65 len 4 [PSM 3 mode 0] {chan 2}
          RFCOMM: Disconnect (DISC) (0x43)
             Address: 0x03 cr 1 dlci 0x00
             Control: 0x53 poll/final 1
             Length: 0
             FCS: 0xfd
    < ACL Data TX: Handle 43 flags 0x00 dlen 8               #395 [hci1] 10.749062
          Channel: 65 len 4 [PSM 3 mode 0] {chan 2}
          RFCOMM: Unnumbered Ack (UA) (0x63)
             Address: 0x03 cr 1 dlci 0x00
             Control: 0x73 poll/final 1
             Length: 0
             FCS: 0xd7
    < ACL Data TX: Handle 43 flags 0x00 dlen 12              #396 [hci1] 10.749073
          L2CAP: Disconnection Request (0x06) ident 17 len 4
            Destination CID: 65
            Source CID: 65
    > HCI Event: Number of Completed Packets (0x13) plen 5   #397 [hci1] 10.752391
            Num handles: 1
            Handle: 43
```

```
        Count: 1
> HCI Event: Number of Completed Packets (0x13) plen 5    #398 [hci1] 10.753394
        Num handles: 1
        Handle: 43
        Count: 1
> ACL Data RX: Handle 43 flags 0x02 dlen 12               #399 [hci1] 10.756499
      L2CAP: Disconnection Request (0x06) ident 26 len 4
        Destination CID: 65
        Source CID: 65
< ACL Data TX: Handle 43 flags 0x00 dlen 12               #400 [hci1] 10.756548
      L2CAP: Disconnection Response (0x07) ident 26 len 4
        Destination CID: 65
        Source CID: 65
< ACL Data TX: Handle 43 flags 0x00 dlen 12               #401 [hci1] 10.757459
      L2CAP: Connection Request (0x02) ident 18 len 4
        PSM: 1 (0x0001)
        Source CID: 65
> ACL Data RX: Handle 43 flags 0x02 dlen 12               #402 [hci1] 10.759148
      L2CAP: Disconnection Response (0x07) ident 17 len 4
        Destination CID: 65
        Source CID: 65
= bluetoothd: 00:1E:AB:4C:56:54: error updating services: Input/o..   10.759447
> HCI Event: Number of Completed Packets (0x13) plen 5    #403 [hci1] 10.759386
        Num handles: 1
        Handle: 43
        Count: 1
> ACL Data RX: Handle 43 flags 0x02 dlen 12               #404 [hci1] 10.760397
      L2CAP: Connection Request (0x02) ident 27 len 4
        PSM: 3 (0x0003)
        Source CID: 65
< ACL Data TX: Handle 43 flags 0x00 dlen 16               #405 [hci1] 10.760441
      L2CAP: Connection Response (0x03) ident 27 len 8
        Destination CID: 65
        Source CID: 65
        Result: Connection successful (0x0000)
        Status: No further information available (0x0000)
< ACL Data TX: Handle 43 flags 0x00 dlen 27               #406 [hci1] 10.760449
      L2CAP: Configure Request (0x04) ident 19 len 19
        Destination CID: 65
        Flags: 0x0000
        Option: Maximum Transmission Unit (0x01) [mandatory]
          MTU: 1013
        Option: Retransmission and Flow Control (0x04) [mandatory]
          Mode: Basic (0x00)
          TX window size: 0
          Max transmit: 0
          Retransmission timeout: 0
          Monitor timeout: 0
          Maximum PDU size: 0
> HCI Event: Number of Completed Packets (0x13) plen 5    #407 [hci1] 10.761399
        Num handles: 1
        Handle: 43
        Count: 1
> ACL Data RX: Handle 43 flags 0x02 dlen 16               #408 [hci1] 10.762942
      L2CAP: Connection Response (0x03) ident 18 len 8
        Destination CID: 66
        Source CID: 65
        Result: Connection successful (0x0000)
        Status: No further information available (0x0000)
*snip*

Similar case after the patch:
*snip*
> ACL Data RX: Handle 43 flags 0x02 dlen 8            #22702 [hci0] 1664.411056
      Channel: 65 len 4 [PSM 3 mode 0] {chan 3}
```

```
            RFCOMM: Disconnect (DISC) (0x43)
               Address: 0x03 cr 1 dlci 0x00
               Control: 0x53 poll/final 1
               Length: 0
               FCS: 0xfd
      < ACL Data TX: Handle 43 flags 0x00 dlen 8           #22703 [hci0] 1664.411136
            Channel: 65 len 4 [PSM 3 mode 0] {chan 3}
            RFCOMM: Unnumbered Ack (UA) (0x63)
               Address: 0x03 cr 1 dlci 0x00
               Control: 0x73 poll/final 1
               Length: 0
               FCS: 0xd7
      < ACL Data TX: Handle 43 flags 0x00 dlen 12          #22704 [hci0] 1664.411143
            L2CAP: Disconnection Request (0x06) ident 11 len 4
              Destination CID: 65
              Source CID: 65
      > HCI Event: Number of Completed Pac.. (0x13) plen 5  #22705 [hci0] 1664.414009
              Num handles: 1
              Handle: 43
              Count: 1
      > HCI Event: Number of Completed Pac.. (0x13) plen 5  #22706 [hci0] 1664.415007
              Num handles: 1
              Handle: 43
              Count: 1
      > ACL Data RX: Handle 43 flags 0x02 dlen 12          #22707 [hci0] 1664.418674
            L2CAP: Disconnection Request (0x06) ident 17 len 4
              Destination CID: 65
              Source CID: 65
      < ACL Data TX: Handle 43 flags 0x00 dlen 12          #22708 [hci0] 1664.418762
            L2CAP: Disconnection Response (0x07) ident 17 len 4
              Destination CID: 65
              Source CID: 65
      < ACL Data TX: Handle 43 flags 0x00 dlen 12          #22709 [hci0] 1664.421073
            L2CAP: Connection Request (0x02) ident 12 len 4
              PSM: 1 (0x0001)
              Source CID: 65
      > ACL Data RX: Handle 43 flags 0x02 dlen 12          #22710 [hci0] 1664.421371
            L2CAP: Disconnection Response (0x07) ident 11 len 4
              Destination CID: 65
              Source CID: 65
      > HCI Event: Number of Completed Pac.. (0x13) plen 5  #22711 [hci0] 1664.424082
              Num handles: 1
              Handle: 43
              Count: 1
      > HCI Event: Number of Completed Pac.. (0x13) plen 5  #22712 [hci0] 1664.425040
              Num handles: 1
              Handle: 43
              Count: 1
      > ACL Data RX: Handle 43 flags 0x02 dlen 12          #22713 [hci0] 1664.426103
            L2CAP: Connection Request (0x02) ident 18 len 4
              PSM: 3 (0x0003)
              Source CID: 65
      < ACL Data TX: Handle 43 flags 0x00 dlen 16          #22714 [hci0] 1664.426186
            L2CAP: Connection Response (0x03) ident 18 len 8
              Destination CID: 66
              Source CID: 65
              Result: Connection successful (0x0000)
              Status: No further information available (0x0000)
      < ACL Data TX: Handle 43 flags 0x00 dlen 27          #22715 [hci0] 1664.426196
            L2CAP: Configure Request (0x04) ident 13 len 19
              Destination CID: 65
              Flags: 0x0000
              Option: Maximum Transmission Unit (0x01) [mandatory]
                MTU: 1013
              Option: Retransmission and Flow Control (0x04) [mandatory]
```

```
                Mode: Basic (0x00)
                TX window size: 0
                Max transmit: 0
                Retransmission timeout: 0
                Monitor timeout: 0
                Maximum PDU size: 0
      > ACL Data RX: Handle 43 flags 0x02 dlen 16           #22716 [hci0] 1664.428804
            L2CAP: Connection Response (0x03) ident 12 len 8
              Destination CID: 66
              Source CID: 65
              Result: Connection successful (0x0000)
              Status: No further information available (0x0000)
      *snip*

      Fix is to check that channel is in state BT_DISCONN before deleting the
      channel.

      This bug was found while fuzzing Bluez's OBEX implementation using
      Synopsys Defensics.

      Reported-by: Matti Kamunen <matti.kamunen@synopsys.com>
      Reported-by: Ari Timonen <ari.timonen@synopsys.com>
      Signed-off-by: Matias Karhumaa <matias.karhumaa@gmail.com>
      Signed-off-by: Marcel Holtmann <marcel@holtmann.org>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 0b49c8a5fa64dc602bf4187cee0b8b570d581c10
  Author: Seeteena Thoufeek <s1seetee@linux.vnet.ibm.com>
  Date:   Thu Jun 27 15:46:54 2019 +0530

      perf tests: Fix record+probe_libc_inet_pton.sh for powerpc64

      [ Upstream commit bff5a556c149804de29347a88a884d25e4e4e3a2 ]

      'probe libc's inet_pton & backtrace it with ping' testcase sometimes
      fails on powerpc because distro ping binary does not have symbol
      information and thus it prints "[unknown]" function name in the
      backtrace.

      Accept "[unknown]" as valid function name for powerpc as well.

       # perf test -v "probe libc's inet_pton & backtrace it with ping"

      Before:

        59: probe libc's inet_pton & backtrace it with ping       :
        --- start ---
        test child forked, pid 79695
        ping 79718 [077] 96483.787025: probe_libc:inet_pton: (7fff83a754c8)
        7fff83a754c8 __GI___inet_pton+0x8 (/usr/lib64/power9/libc-2.28.so)
        7fff83a2b7a0 gaih_inet.constprop.7+0x1020
        (/usr/lib64/power9/libc-2.28.so)
        7fff83a2c170 getaddrinfo+0x160 (/usr/lib64/power9/libc-2.28.so)
        1171830f4 [unknown] (/usr/bin/ping)
        FAIL: expected backtrace entry
        ".*\+0x[[:xdigit:]]+[[:space:]]\(.*/bin/ping.*\)$"
        got "1171830f4 [unknown] (/usr/bin/ping)"
        test child finished with -1
        ---- end ----
        probe libc's inet_pton & backtrace it with ping: FAILED!

      After:

        59: probe libc's inet_pton & backtrace it with ping       :
        --- start ---
```

```
       test child forked, pid 79085
       ping 79108 [045] 96400.214177: probe_libc:inet_pton: (7fffbb9654c8)
       7fffbb9654c8 __GI___inet_pton+0x8 (/usr/lib64/power9/libc-2.28.so)
       7fffbb91b7a0 gaih_inet.constprop.7+0x1020
       (/usr/lib64/power9/libc-2.28.so)
       7fffbb91c170 getaddrinfo+0x160 (/usr/lib64/power9/libc-2.28.so)
       132e830f4 [unknown] (/usr/bin/ping)
       test child finished with 0
       ---- end ----
       probe libc's inet_pton & backtrace it with ping: Ok

    Signed-off-by: Seeteena Thoufeek <s1seetee@linux.vnet.ibm.com>
    Reviewed-by: Kim Phillips <kim.phillips@amd.com>
    Cc: Alexander Shishkin <alexander.shishkin@linux.intel.com>
    Cc: Hendrik Brueckner <brueckner@linux.ibm.com>
    Cc: Jiri Olsa <jolsa@redhat.com>
    Cc: Michael Petlan <mpetlan@redhat.com>
    Cc: Namhyung Kim <namhyung@kernel.org>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Cc: Sandipan Das <sandipan@linux.ibm.com>
    Fixes: 1632936480a5 ("perf tests: Fix record+probe_libc_inet_pton.sh without ping's
debuginfo")
    Link: http://lkml.kernel.org/r/1561630614-3216-1-git-send-email-
s1seetee@linux.vnet.ibm.com
    Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 75b0fe87722cec3088972f6096925ec91ef35dff
Author: Shijith Thotton <sthotton@marvell.com>
Date:   Fri Jul 5 07:56:20 2019 +0000

    genirq: Update irq stats from NMI handlers

    [ Upstream commit c09cb1293523dd786ae54a12fd88001542cba2f6 ]

    The NMI handlers handle_percpu_devid_fasteoi_nmi() and handle_fasteoi_nmi()
    do not update the interrupt counts. Due to that the NMI interrupt count
    does not show up correctly in /proc/interrupts.

    Add the statistics and treat the NMI handlers in the same way as per cpu
    interrupts and prevent them from updating irq_desc::tot_count as this might
    be corrupted due to concurrency.

    [ tglx: Massaged changelog ]

    Fixes: 2dcf1fbcad35 ("genirq: Provide NMI handlers")
    Signed-off-by: Shijith Thotton <sthotton@marvell.com>
    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Link: https://lkml.kernel.org/r/1562313336-11888-1-git-send-email-sthotton@marvell.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5dface2ce2f28f284502180108aa99e58932c73d
Author: Josua Mayer <josua.mayer@jm0.eu>
Date:   Sat Jul 6 17:54:46 2019 +0200

    Bluetooth: 6lowpan: search for destination address in all peers

    [ Upstream commit b188b03270b7f8568fc714101ce82fbf5e811c5a ]

    Handle overlooked case where the target address is assigned to a peer
    and neither route nor gateway exist.

    For one peer, no checks are performed to see if it is meant to receive
    packets for a given address.
```

```
    As soon as there is a second peer however, checks are performed
    to deal with routes and gateways for handling complex setups with
    multiple hops to a target address.
    This logic assumed that no route and no gateway imply that the
    destination address can not be reached, which is false in case of a
    direct peer.

    Acked-by: Jukka Rissanen <jukka.rissanen@linux.intel.com>
    Tested-by: Michael Scott <mike@foundries.io>
    Signed-off-by: Josua Mayer <josua.mayer@jm0.eu>
    Signed-off-by: Marcel Holtmann <marcel@holtmann.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e7de2b6a3d9d53ba0cd96522500cade791736bab
Author: João Paulo Rechi Vita <jprvita@gmail.com>
Date:   Thu May 23 13:32:02 2019 -0700

    Bluetooth: Add new 13d3:3501 QCA_ROME device

    [ Upstream commit 881cec4f6b4da78e54b73c046a60f39315964c7d ]

    Without the QCA ROME setup routine this adapter fails to establish a SCO
    connection.

    T:  Bus=01 Lev=01 Prnt=01 Port=04 Cnt=01 Dev#=  2 Spd=12  MxCh= 0
    D:  Ver= 1.10 Cls=e0(wlcon) Sub=01 Prot=01 MxPS=64 #Cfgs=  1
    P:  Vendor=13d3 ProdID=3501 Rev=00.01
    C:  #Ifs= 2 Cfg#= 1 Atr=e0 MxPwr=100mA
    I:  If#=0x0 Alt= 0 #EPs= 3 Cls=e0(wlcon) Sub=01 Prot=01 Driver=btusb
    I:  If#=0x1 Alt= 0 #EPs= 2 Cls=e0(wlcon) Sub=01 Prot=01 Driver=btusb

    Signed-off-by: João Paulo Rechi Vita <jprvita@endlessm.com>
    Signed-off-by: Marcel Holtmann <marcel@holtmann.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 78dd5ae3154f92a732c97690b0c15abea113b2eb
Author: João Paulo Rechi Vita <jprvita@gmail.com>
Date:   Thu May 23 13:32:01 2019 -0700

    Bluetooth: Add new 13d3:3491 QCA_ROME device

    [ Upstream commit 44d34af2e4cfd0c5357182f8b43f3e0a1fe30a2e ]

    Without the QCA ROME setup routine this adapter fails to establish a SCO
    connection.

    T:  Bus=01 Lev=01 Prnt=01 Port=08 Cnt=01 Dev#=  2 Spd=12  MxCh= 0
    D:  Ver= 1.10 Cls=e0(wlcon) Sub=01 Prot=01 MxPS=64 #Cfgs=  1
    P:  Vendor=13d3 ProdID=3491 Rev=00.01
    C:  #Ifs= 2 Cfg#= 1 Atr=e0 MxPwr=100mA
    I:  If#=0x0 Alt= 0 #EPs= 3 Cls=e0(wlcon) Sub=01 Prot=01 Driver=btusb
    I:  If#=0x1 Alt= 0 #EPs= 2 Cls=e0(wlcon) Sub=01 Prot=01 Driver=btusb

    Signed-off-by: João Paulo Rechi Vita <jprvita@endlessm.com>
    Signed-off-by: Marcel Holtmann <marcel@holtmann.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b3f0f22055bf90dfadf1c6726f52c0e25d9e9b92
Author: Tomas Bortoli <tomasbortoli@gmail.com>
Date:   Tue May 28 15:42:58 2019 +0200

    Bluetooth: hci_bcsp: Fix memory leak in rx_skb

    [ Upstream commit 4ce9146e0370fcd573f0372d9b4e5a211112567c ]
```

```
    Syzkaller found that it is possible to provoke a memory leak by
    never freeing rx_skb in struct bcsp_struct.

    Fix by freeing in bcsp_close()

    Signed-off-by: Tomas Bortoli <tomasbortoli@gmail.com>
    Reported-by: syzbot+98162c885993b72f19c4@syzkaller.appspotmail.com
    Signed-off-by: Marcel Holtmann <marcel@holtmann.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a5635e3d5a3a562e292835bde77be7678b8feb6f
Author: Jian Shen <shenjian15@huawei.com>
Date:   Thu Jul 4 22:04:22 2019 +0800

    net: hns3: fix port capbility updating issue

    [ Upstream commit 49b1255603de5183c5e377200be3b3afe0dcdb86 ]

    Currently, the driver queries the media port information, and
    updates the port capability periodically. But it sets an error
    mac->speed_type value, which stops update port capability.

    Fixes: 88d10bd6f730 ("net: hns3: add support for multiple media type")
    Signed-off-by: Jian Shen <shenjian15@huawei.com>
    Signed-off-by: Peng Li <lipeng321@huawei.com>
    Signed-off-by: Huazhong Tan <tanhuazhong@huawei.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5bc88c9338f5356cc18fa65d58fadab5d1976fda
Author: Jian Shen <shenjian15@huawei.com>
Date:   Thu Jul 4 22:04:20 2019 +0800

    net: hns3: enable broadcast promisc mode when initializing VF

    [ Upstream commit 2d5066fc175ea77a733d84df9ef414b34f311641 ]

    For revision 0x20, the broadcast promisc is enabled by firmware,
    it's unnecessary to enable it when initializing VF.

    For revision 0x21, it's necessary to enable broadcast promisc mode
    when initializing or re-initializing VF, otherwise, it will be
    unable to send and receive promisc packets.

    Fixes: f01f5559cac8 ("net: hns3: don't allow vf to enable promisc mode")
    Signed-off-by: Jian Shen <shenjian15@huawei.com>
    Signed-off-by: Peng Li <lipeng321@huawei.com>
    Signed-off-by: Huazhong Tan <tanhuazhong@huawei.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0d61dc83f0e8d82e6734fcadd0b2d1eddfe7f977
Author: Jiri Olsa <jolsa@redhat.com>
Date:   Fri Jul 5 14:10:31 2019 +0200

    tools: bpftool: Fix json dump crash on powerpc

    [ Upstream commit aa52bcbe0e72fac36b1862db08b9c09c4caefae3 ]

    Michael reported crash with by bpf program in json mode on powerpc:

      # bpftool prog -p dump jited id 14
      [{
            "name": "0xd00000000a9aa760",
            "insns": [{
```

```
                    "pc": "0x0",
                    "operation": "nop",
                    "operands": [null
                    ]
                },{
                    "pc": "0x4",
                    "operation": "nop",
                    "operands": [null
                    ]
                },{
                    "pc": "0x8",
                    "operation": "mflr",
      Segmentation fault (core dumped)
```

    The code is assuming char pointers in format, which is not always
    true at least for powerpc. Fixing this by dumping the whole string
    into buffer based on its format.

    Please note that libopcodes code does not check return values from
    fprintf callback, but as per Jakub suggestion returning -1 on allocation
    failure so we do the best effort to propagate the error.

    Fixes: 107f041212c1 ("tools: bpftool: add JSON output for `bpftool prog dump jited *`
command")
    Reported-by: Michael Petlan <mpetlan@redhat.com>
    Signed-off-by: Jiri Olsa <jolsa@kernel.org>
    Reviewed-by: Quentin Monnet <quentin.monnet@netronome.com>
    Reviewed-by: Jakub Kicinski <jakub.kicinski@netronome.com>
    Signed-off-by: Daniel Borkmann <daniel@iogearbox.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5a156f038c7de5fb5d4b2d8299f8fcfbf343f3f4
Author: Wen Yang <wen.yang99@zte.com.cn>
Date: Thu Jul 4 16:38:50 2019 +0800

    ASoC: audio-graph-card: fix use-after-free in graph_for_each_link

    [Upstream commit 1bcc1fd64e4dd903f4d868a9e053986e3b102713]

    After calling of_node_put() on the codec_ep and codec_port variables,
    they are still being used, which may result in use-after-free.
    We fix this issue by calling of_node_put() after the last usage.

    Fixes: fce9b90c1ab7 ("ASoC: audio-graph-card: cleanup DAI link loop method - step2")
    Signed-off-by: Wen Yang <wen.yang99@zte.com.cn>
    Cc: Liam Girdwood <lgirdwood@gmail.com>
    Cc: Mark Brown <broonie@kernel.org>
    Cc: Jaroslav Kysela <perex@perex.cz>
    Cc: Takashi Iwai <tiwai@suse.com>
    Cc: Kuninori Morimoto <kuninori.morimoto.gx@renesas.com>
    Cc: alsa-devel@alsa-project.org
    Cc: linux-kernel@vger.kernel.org
    Link: https://lore.kernel.org/r/1562229530-8121-1-git-send-email-wen.yang99@zte.com.cn
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0417b0fda7f992b61fd132fec64df4b90b21c494
Author: Jean-Philippe Brucker <jean-philippe.brucker@arm.com>
Date: Wed Jul 3 12:19:20 2019 +0100

    iommu/arm-smmu-v3: Invalidate ATC when detaching a device

    [Upstream commit 8dd8f005bdd45823fc153ef490239558caf6ff20]

    We make the invalid assumption in arm_smmu_detach_dev() that the ATC is

```
    clear after calling pci_disable_ats(). For one thing, only enabling the
    PCIe ATS capability constitutes an implicit invalidation event, so the
    comment was wrong. More importantly, the ATS capability isn't necessarily
    disabled by pci_disable_ats() in a PF, if the associated VFs have ATS
    enabled. Explicitly invalidate all ATC entries in arm_smmu_detach_dev().
    The endpoint cannot form new ATC entries because STE.EATS is clear.

    Fixes: 9ce27afc0830 ("iommu/arm-smmu-v3: Add support for PCI ATS")
    Reported-by: Manoj Kumar <Manoj.Kumar3@arm.com>
    Reported-by: Robin Murphy <Robin.Murphy@arm.com>
    Signed-off-by: Jean-Philippe Brucker <jean-philippe.brucker@arm.com>
    Acked-by: Will Deacon <will@kernel.org>
    Signed-off-by: Joerg Roedel <jroedel@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 72911828bb44cb846376f50506a378851e24fad4
Author: Geert Uytterhoeven <geert+renesas@glider.be>
Date:   Mon Jul 1 16:27:38 2019 +0200

    gpiolib: Fix references to gpiod_[gs]et_*value_cansleep() variants

    [ Upstream commit 3285170f28a850638794cdfe712eb6d93e51e706 ]

    Commit 372e722ea4dd4ca1 ("gpiolib: use descriptors internally") renamed
    the functions to use a "gpiod" prefix, and commit 79a9becda8940deb
    ("gpiolib: export descriptor-based GPIO interface") introduced the "raw"
    variants, but both changes forgot to update the comments.

    Readd a similar reference to gpiod_set_value(), which was accidentally
    removed by commit 1e77fc82110ac36f ("gpio: Add missing open drain/source
    handling to gpiod_set_value_cansleep()").

    Signed-off-by: Geert Uytterhoeven <geert+renesas@glider.be>
    Link: https://lore.kernel.org/r/20190701142738.25219-1-geert+renesas@glider.be
    Signed-off-by: Linus Walleij <linus.walleij@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit bf26551d73ac1b1c0772995a40cfe889876ea3ec
Author: Cong Wang <xiyou.wangcong@gmail.com>
Date:   Mon Jul 1 20:40:24 2019 -0700

    bonding: validate ip header before check IPPROTO_IGMP

    [ Upstream commit 9d1bc24b52fb8c5d859f9a47084bf1179470e04c ]

    bond_xmit_roundrobin() checks for IGMP packets but it parses
    the IP header even before checking skb->protocol.

    We should validate the IP header with pskb_may_pull() before
    using iph->protocol.

    Reported-and-tested-by: syzbot+e5be16aa39ad6e755391@syzkaller.appspotmail.com
    Fixes: a2fd940f4cff ("bonding: fix broken multicast with round-robin mode")
    Cc: Jay Vosburgh <j.vosburgh@gmail.com>
    Cc: Veaceslav Falico <vfalico@gmail.com>
    Cc: Andy Gospodarek <andy@greyhouse.net>
    Signed-off-by: Cong Wang <xiyou.wangcong@gmail.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6ee610d804a6e795a9a98f7a3b7be48a6779c58f
Author: Jiri Benc <jbenc@redhat.com>
Date:   Tue Jul 2 19:40:31 2019 +0200

    selftests: bpf: fix inlines in test_lwt_seg6local
```

```
    [ Upstream commit 11aca65ec4db09527d3e9b6b41a0615b7da4386b ]

    Selftests are reporting this failure in test_lwt_seg6local.sh:

    + ip netns exec ns2 ip -6 route add fb00::6 encap bpf in obj test_lwt_seg6local.o sec
 encap_srh dev veth2
    Error fetching program/map!
    Failed to parse eBPF program: Operation not permitted

    The problem is __attribute__((always_inline)) alone is not enough to prevent
    clang from inserting those functions in .text. In that case, .text is not
    marked as relocateable.

    See the output of objdump -h test_lwt_seg6local.o:

    Idx Name          Size      VMA               LMA               File off  Algn
      0 .text         00003530  0000000000000000  0000000000000000  00000040  2**3
                      CONTENTS, ALLOC, LOAD, READONLY, CODE

    This causes the iproute bpf loader to fail in bpf_fetch_prog_sec:
    bpf_has_call_data returns true but bpf_fetch_prog_relo fails as there's no
    relocateable .text section in the file.

    To fix this, convert to 'static __always_inline'.

    v2: Use 'static __always_inline' instead of 'static inline
        __attribute__((always_inline))'

    Fixes: c99a84eac026 ("selftests/bpf: test for seg6local End.BPF action")
    Signed-off-by: Jiri Benc <jbenc@redhat.com>
    Acked-by: Yonghong Song <yhs@fb.com>
    Signed-off-by: Daniel Borkmann <daniel@iogearbox.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit dab7df3072b4e32731107d3219c98507342b0f96
 Author: Leo Yan <leo.yan@linaro.org>
 Date:   Tue Jul 2 18:25:31 2019 +0800

    bpf, libbpf, smatch: Fix potential NULL pointer dereference

    [ Upstream commit 33bae185f74d49a0d7b1bfaafb8e959efce0f243 ]

    Based on the following report from Smatch, fix the potential NULL
    pointer dereference check:

      tools/lib/bpf/libbpf.c:3493
      bpf_prog_load_xattr() warn: variable dereferenced before check 'attr'
      (see line 3483)

      3479 int bpf_prog_load_xattr(const struct bpf_prog_load_attr *attr,
      3480                           struct bpf_object **pobj, int *prog_fd)
      3481 {
      3482         struct bpf_object_open_attr open_attr = {
      3483                 .file           = attr->file,
      3484                 .prog_type      = attr->prog_type,
                                              ^^^^^^
      3485         };

    At the head of function, it directly access 'attr' without checking
    if it's NULL pointer. This patch moves the values assignment after
    validating 'attr' and 'attr->file'.

    Signed-off-by: Leo Yan <leo.yan@linaro.org>
    Acked-by: Yonghong Song <yhs@fb.com>
```

```
    Signed-off-by: Daniel Borkmann <daniel@iogearbox.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ba38126fbd973264d4c4c35c937c13de511097f6
Author: Andrii Nakryiko <andriin@fb.com>
Date:   Tue Jul 2 08:16:20 2019 -0700

    libbpf: fix GCC8 warning for strncpy

    [ Upstream commit cdfc7f888c2a355b01308e97c6df108f1c2b64e8 ]

    GCC8 started emitting warning about using strncpy with number of bytes
    exactly equal destination size, which is generally unsafe, as can lead
    to non-zero terminated string being copied. Use IFNAMSIZ - 1 as number
    of bytes to ensure name is always zero-terminated.

    Signed-off-by: Andrii Nakryiko <andriin@fb.com>
    Cc: Magnus Karlsson <magnus.karlsson@intel.com>
    Acked-by: Yonghong Song <yhs@fb.com>
    Acked-by: Magnus Karlsson <magnus.karlsson@intel.com>
    Signed-off-by: Daniel Borkmann <daniel@iogearbox.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c352c3e84b6e795f8ee8055f5f5152bba6ac9508
Author: David Howells <dhowells@redhat.com>
Date:   Tue Jul 2 16:04:19 2019 +0100

    rxrpc: Fix oops in tracepoint

    [ Upstream commit 99f0eae653b2db64917d0b58099eb51e300b311d ]

    If the rxrpc_eproto tracepoint is enabled, an oops will be cause by the
    trace line that rxrpc_extract_header() tries to emit when a protocol error
    occurs (typically because the packet is short) because the call argument is
    NULL.

    Fix this by using ?: to assume 0 as the debug_id if call is NULL.

    This can then be induced by:

            echo -e '\0\0\0\0\0\0\0' | ncat -4u --send-only <addr> 20001

    where addr has the following program running on it:

            #include <stdio.h>
            #include <stdlib.h>
            #include <string.h>
            #include <unistd.h>
            #include <sys/socket.h>
            #include <arpa/inet.h>
            #include <linux/rxrpc.h>
            int main(void)
            {
                    struct sockaddr_rxrpc srx;
                    int fd;
                    memset(&srx, 0, sizeof(srx));
                    srx.srx_family                  = AF_RXRPC;
                    srx.srx_service                 = 0;
                    srx.transport_type              = AF_INET;
                    srx.transport_len               = sizeof(srx.transport.sin);
                    srx.transport.sin.sin_family    = AF_INET;
                    srx.transport.sin.sin_port      = htons(0x4e21);
                    fd = socket(AF_RXRPC, SOCK_DGRAM, AF_INET6);
                    bind(fd, (struct sockaddr *)&srx, sizeof(srx));
                    sleep(20);
```

```
                return 0;
        }
```

    It results in the following oops.

```
        BUG: kernel NULL pointer dereference, address: 0000000000000340
        #PF: supervisor read access in kernel mode
        #PF: error_code(0x0000) - not-present page
        ...
        RIP: 0010:trace_event_raw_event_rxrpc_rx_eproto+0x47/0xac
        ...
        Call Trace:
         <IRQ>
         rxrpc_extract_header+0x86/0x171
         ? rcu_read_lock_sched_held+0x5d/0x63
         ? rxrpc_new_skb+0xd4/0x109
         rxrpc_input_packet+0xef/0x14fc
         ? rxrpc_input_data+0x986/0x986
         udp_queue_rcv_one_skb+0xbf/0x3d0
         udp_unicast_rcv_skb.isra.8+0x64/0x71
         ip_protocol_deliver_rcu+0xe4/0x1b4
         ip_local_deliver+0xf0/0x154
         __netif_receive_skb_one_core+0x50/0x6c
         netif_receive_skb_internal+0x26b/0x2e9
         napi_gro_receive+0xf8/0x1da
         rtl8169_poll+0x303/0x4c4
         net_rx_action+0x10e/0x333
         __do_softirq+0x1a5/0x38f
         irq_exit+0x54/0xc4
         do_IRQ+0xda/0xf8
         common_interrupt+0xf/0xf
         </IRQ>
         ...
         ? cpuidle_enter_state+0x23c/0x34d
         cpuidle_enter+0x2a/0x36
         do_idle+0x163/0x1ea
         cpu_startup_entry+0x1d/0x1f
         start_secondary+0x157/0x172
         secondary_startup_64+0xa4/0xb0
```

    Fixes: a25e21f0bcd2 ("rxrpc, afs: Use debug_ids rather than pointers in traces")
    Signed-off-by: David Howells <dhowells@redhat.com>
    Reviewed-by: Marc Dionne <marc.dionne@auristor.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1c1fd391d4ca3eef9fc979b9b90456c9fe7fcdaf
Author: Phong Tran <tranmanphong@gmail.com>
Date: Tue Jul 2 07:10:08 2019 +0700

    net: usb: asix: init MAC address buffers

    [Upstream commit 78226f6eaac80bf30256a33a4926c194ceefdf36]

    This is for fixing bug KMSAN: uninit-value in ax88772_bind

    Tested by
    https://groups.google.com/d/msg/syzkaller-bugs/aFQurGotng4/eB_HlNhhCwAJ

    Reported-by: syzbot+8a3fc6674bbc3978ed4e@syzkaller.appspotmail.com

    syzbot found the following crash on:

    HEAD commit: f75e4cfe kmsan: use kmsan_handle_urb() in urb.c
    git tree: kmsan

```
    console output: https://syzkaller.appspot.com/x/log.txt?x=136d720ea00000
    kernel config:
    https://syzkaller.appspot.com/x/.config?x=602468164ccdc30a
    dashboard link:
    https://syzkaller.appspot.com/bug?extid=8a3fc6674bbc3978ed4e
    compiler:       clang version 9.0.0 (/home/glider/llvm/clang
    06d00afa61eef8f7f501ebdb4e8612ea43ec2d78)
    syz repro:
    https://syzkaller.appspot.com/x/repro.syz?x=12788316a00000
    C reproducer:   https://syzkaller.appspot.com/x/repro.c?x=120359aaa00000

    ==================================================================
    BUG: KMSAN: uninit-value in is_valid_ether_addr
    include/linux/etherdevice.h:200 [inline]
    BUG: KMSAN: uninit-value in asix_set_netdev_dev_addr
    drivers/net/usb/asix_devices.c:73 [inline]
    BUG: KMSAN: uninit-value in ax88772_bind+0x93d/0x11e0
    drivers/net/usb/asix_devices.c:724
    CPU: 0 PID: 3348 Comm: kworker/0:2 Not tainted 5.1.0+ #1
    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+0x191/0x1f0 lib/dump_stack.c:113
      kmsan_report+0x130/0x2a0 mm/kmsan/kmsan.c:622
      __msan_warning+0x75/0xe0 mm/kmsan/kmsan_instr.c:310
      is_valid_ether_addr include/linux/etherdevice.h:200 [inline]
      asix_set_netdev_dev_addr drivers/net/usb/asix_devices.c:73 [inline]
      ax88772_bind+0x93d/0x11e0 drivers/net/usb/asix_devices.c:724
      usbnet_probe+0x10f5/0x3940 drivers/net/usb/usbnet.c:1728
      usb_probe_interface+0xd66/0x1320 drivers/usb/core/driver.c:361
      really_probe+0xdae/0x1d80 drivers/base/dd.c:513
      driver_probe_device+0x1b3/0x4f0 drivers/base/dd.c:671
      __device_attach_driver+0x5b8/0x790 drivers/base/dd.c:778
      bus_for_each_drv+0x28e/0x3b0 drivers/base/bus.c:454
      __device_attach+0x454/0x730 drivers/base/dd.c:844
      device_initial_probe+0x4a/0x60 drivers/base/dd.c:891
      bus_probe_device+0x137/0x390 drivers/base/bus.c:514
      device_add+0x288d/0x30e0 drivers/base/core.c:2106
      usb_set_configuration+0x30dc/0x3750 drivers/usb/core/message.c:2027
      generic_probe+0xe7/0x280 drivers/usb/core/generic.c:210
      usb_probe_device+0x14c/0x200 drivers/usb/core/driver.c:266
      really_probe+0xdae/0x1d80 drivers/base/dd.c:513
      driver_probe_device+0x1b3/0x4f0 drivers/base/dd.c:671
      __device_attach_driver+0x5b8/0x790 drivers/base/dd.c:778
      bus_for_each_drv+0x28e/0x3b0 drivers/base/bus.c:454
      __device_attach+0x454/0x730 drivers/base/dd.c:844
      device_initial_probe+0x4a/0x60 drivers/base/dd.c:891
      bus_probe_device+0x137/0x390 drivers/base/bus.c:514
      device_add+0x288d/0x30e0 drivers/base/core.c:2106
      usb_new_device+0x23e5/0x2ff0 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+0x48d1/0x7290 drivers/usb/core/hub.c:5432
      process_one_work+0x1572/0x1f00 kernel/workqueue.c:2269
      process_scheduled_works kernel/workqueue.c:2331 [inline]
      worker_thread+0x189c/0x2460 kernel/workqueue.c:2417
      kthread+0x4b5/0x4f0 kernel/kthread.c:254
      ret_from_fork+0x35/0x40 arch/x86/entry/entry_64.S:355

    Signed-off-by: Phong Tran <tranmanphong@gmail.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 846027be71489ffb092996791b9a081db5cda2b6
Author: Guilherme G. Piccoli <gpiccoli@canonical.com>
Date:   Thu Jun 27 13:31:33 2019 -0300

    bnx2x: Prevent ptp_task to be rescheduled indefinitely

    [ Upstream commit 3c91f25c2f72ba6001775a5932857c1d2131c531 ]

    Currently bnx2x ptp worker tries to read a register with timestamp
    information in case of TX packet timestamping and in case it fails,
    the routine reschedules itself indefinitely. This was reported as a
    kworker always at 100% of CPU usage, which was narrowed down to be
    bnx2x ptp_task.

    By following the ioctl handler, we could narrow down the problem to
    an NTP tool (chrony) requesting HW timestamping from bnx2x NIC with
    RX filter zeroed; this isn't reproducible for example with ptp4l
    (from linuxptp) since this tool requests a supported RX filter.
    It seems NIC FW timestamp mechanism cannot work well with
    RX_FILTER_NONE - driver's PTP filter init routine skips a register
    write to the adapter if there's not a supported filter request.

    This patch addresses the problem of bnx2x ptp thread's everlasting
    reschedule by retrying the register read 10 times; between the read
    attempts the thread sleeps for an increasing amount of time starting
    in 1ms to give FW some time to perform the timestamping. If it still
    fails after all retries, we bail out in order to prevent an unbound
    resource consumption from bnx2x.

    The patch also adds an ethtool statistic for accounting the skipped
    TX timestamp packets and it reduces the priority of timestamping
    error messages to prevent log flooding. The code was tested using
    both linuxptp and chrony.

    Reported-and-tested-by: Przemyslaw Hausman <przemyslaw.hausman@canonical.com>
    Suggested-by: Sudarsana Reddy Kalluru <skalluru@marvell.com>
    Signed-off-by: Guilherme G. Piccoli <gpiccoli@canonical.com>
    Acked-by: Sudarsana Reddy Kalluru <skalluru@marvell.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7429e6e2fb002bfd1117975abd0d659e17ada2ed
Author: Taehee Yoo <ap420073@gmail.com>
Date:   Fri Jun 28 14:07:25 2019 +0900

    vxlan: do not destroy fdb if register_netdevice() is failed

    [ Upstream commit 7c31e54aeee517d1318dfc0bde9fa7de75893dc6 ]

    __vxlan_dev_create() destroys FDB using specific pointer which indicates
    a fdb when error occurs.
    But that pointer should not be used when register_netdevice() fails because
    register_netdevice() internally destroys fdb when error occurs.

    This patch makes vxlan_fdb_create() to do not link fdb entry to vxlan dev
    internally.
    Instead, a new function vxlan_fdb_insert() is added to link fdb to vxlan
    dev.

    vxlan_fdb_insert() is called after calling register_netdevice().
    This routine can avoid situation that ->ndo_uninit() destroys fdb entry
    in error path of register_netdevice().
    Hence, error path of __vxlan_dev_create() routine can have an opportunity
    to destroy default fdb entry by hand.
```

```
    Test command
        ip link add bonding_masters type vxlan id 0 group 239.1.1.1 \
                dev enp0s9 dstport 4789

    Splat looks like:
    [  213.392816] kasan: GPF could be caused by NULL-ptr deref or user memory access
    [  213.401257] general protection fault: 0000 [#1] SMP DEBUG_PAGEALLOC KASAN PTI
    [  213.402178] CPU: 0 PID: 1414 Comm: ip Not tainted 5.2.0-rc5+ #256
    [  213.402178] RIP: 0010:vxlan_fdb_destroy+0x120/0x220 [vxlan]
    [  213.402178] Code: df 48 8b 2b 48 89 fa 48 c1 ea 03 80 3c 02 00 0f 85 06 01 00 00 4c
8b 63 08 48 b8 00 00 00 00 00 fc d
    [  213.402178] RSP: 0018:ffff88810cb9f0a0 EFLAGS: 00010202
    [  213.402178] RAX: dfffc0000000000 RBX: ffff888101d4a8c8 RCX: 0000000000000000
    [  213.402178] RDX: 1bd5a00000000040 RSI: ffff888101d4a8c8 RDI: ffff888101d4a8d0
    [  213.402178] RBP: 0000000000000000 R08: fffffbfff22b72d9 R09: 0000000000000000
    [  213.402178] R10: 00000000ffffffef R11: 0000000000000000 R12: dead000000000200
    [  213.402178] R13: ffff88810cb9f1f8 R14: ffff88810efccda0 R15: ffff88810efccda0
    [  213.402178] FS:  00007f7f6621a0c0(0000) GS:ffff88811b000000(0000)
knlGS:0000000000000000
    [  213.402178] CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    [  213.402178] CR2: 000055746f0807d0 CR3: 00000001123e0000 CR4: 00000000001006f0
    [  213.402178] Call Trace:
    [  213.402178]  __vxlan_dev_create+0x3a9/0x7d0 [vxlan]
    [  213.402178]  ? vxlan_changelink+0x740/0x740 [vxlan]
    [  213.402178]  ? rcu_read_unlock+0x60/0x60 [vxlan]
    [  213.402178]  ? __kasan_kmalloc.constprop.3+0xa0/0xd0
    [  213.402178]  vxlan_newlink+0x8d/0xc0 [vxlan]
    [  213.402178]  ? __vxlan_dev_create+0x7d0/0x7d0 [vxlan]
    [  213.554119]  ? __netlink_ns_capable+0xc3/0xf0
    [  213.554119]  __rtnl_newlink+0xb75/0x1180
    [  213.554119]  ? rtnl_link_unregister+0x230/0x230
    [ ... ]

    Fixes: 0241b836732f ("vxlan: fix default fdb entry netlink notify ordering during netdev
create")
    Suggested-by: Roopa Prabhu <roopa@cumulusnetworks.com>
    Signed-off-by: Taehee Yoo <ap420073@gmail.com>
    Acked-by: Roopa Prabhu <roopa@cumulusnetworks.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7c51ec4612034215f28d1f2d5fdad908cc321aae
Author: Andi Kleen <ak@linux.intel.com>
Date:   Mon Jun 24 12:37:10 2019 -0700

    perf stat: Fix group lookup for metric group

    [ Upstream commit 2f87f33f4226523df9c9cc28f9874ea02fcc3d3f ]

    The metric group code tries to find a group it added earlier in the
    evlist. Fix the lookup to handle groups with partially overlaps
    correctly. When a sub string match fails and we reset the match, we have
    to compare the first element again.

    I also renamed the find_evsel function to find_evsel_group to make its
    purpose clearer.

    With the earlier changes this fixes:

    Before:

      % perf stat -M UPI,IPC sleep 1
      ...
             1,032,922      uops_retired.retire_slots #      1.1 UPI
```

```
             1,896,096      inst_retired.any
             1,896,096      inst_retired.any
             1,177,254      cpu_clk_unhalted.thread
```

After:

```
  % perf stat -M UPI,IPC sleep 1
  ...
        1,013,193      uops_retired.retire_slots #      1.1 UPI
           932,033      inst_retired.any
           932,033      inst_retired.any          #      0.9 IPC
         1,091,245      cpu_clk_unhalted.thread
```

Signed-off-by: Andi Kleen <ak@linux.intel.com>
Acked-by: Jiri Olsa <jolsa@kernel.org>
Cc: Kan Liang <kan.liang@linux.intel.com>
Fixes: b18f3e365019 ("perf stat: Support JSON metrics in perf stat")
Link: http://lkml.kernel.org/r/20190624193711.35241-4-andi@firstfloor.org
Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b610b5f080128142b36cda853f2add987da3faec
Author: Andi Kleen <ak@linux.intel.com>
Date: Mon Jun 24 12:37:09 2019 -0700

perf stat: Don't merge events in the same PMU

[Upstream commit 6c5f4e5cb35b4694dc035d91092d23f596ecd06a]

Event merging is mainly to collapse similar events in lots of different
duplicated PMUs.

It can break metric displaying. It's possible for two metrics to have
the same event, and when the two events happen in a row the second
wouldn't be displayed. This would also not show the second metric.

To avoid this don't merge events in the same PMU. This makes sense, if
we have multiple events in the same PMU there is likely some reason for
it (e.g. using multiple groups) and we better not merge them.

While in theory it would be possible to construct metrics that have
events with the same name in different PMU no current metrics have this
problem.

This is the fix for perf stat -M UPI,IPC (needs also another bug fix to
completely work)

Signed-off-by: Andi Kleen <ak@linux.intel.com>
Acked-by: Jiri Olsa <jolsa@kernel.org>
Cc: Kan Liang <kan.liang@linux.intel.com>
Fixes: 430daf2dc7af ("perf stat: Collapse identically named events")
Link: http://lkml.kernel.org/r/20190624193711.35241-3-andi@firstfloor.org
Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 97ceeb9d1a5747eb236b0972515b489771b0d828
Author: Andi Kleen <ak@linux.intel.com>
Date: Mon Jun 24 12:37:11 2019 -0700

perf stat: Fix metrics with --no-merge

[Upstream commit e3a9427323a53ceee540276a74af7706f350d052]

Since Fixes: 8c5421c016a4 ("perf pmu: Display pmu name when printing
unmerged events in stat") using --no-merge adds the PMU name to the

```
    evsel name.

    This breaks the metric value lookup because the parser doesn't know
    about this.

    Remove the extra postfixes for the metric evaluation.

    Signed-off-by: Andi Kleen <ak@linux.intel.com>
    Acked-by: Jiri Olsa <jolsa@kernel.org>
    Cc: Agustin Vega-Frias <agustinv@codeaurora.org>
    Cc: Kan Liang <kan.liang@linux.intel.com>
    Fixes: 8c5421c016a4 ("perf pmu: Display pmu name when printing unmerged events in stat")
    Link: http://lkml.kernel.org/r/20190624193711.35241-5-andi@firstfloor.org
    Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 50d549cefaf4668ce372936b96779aa1fc748308
Author: Andi Kleen <ak@linux.intel.com>
Date:   Mon Jun 24 12:37:08 2019 -0700

    perf stat: Make metric event lookup more robust

    [ Upstream commit 145c407c808352acd625be793396fd4f33c794f8 ]

    After setting up metric groups through the event parser, the metricgroup
    code looks them up again in the event list.

    Make sure we only look up events that haven't been used by some other
    metric. The data structures currently cannot handle more than one metric
    per event. This avoids problems with multiple events partially
    overlapping.

    Signed-off-by: Andi Kleen <ak@linux.intel.com>
    Acked-by: Jiri Olsa <jolsa@kernel.org>
    Cc: Kan Liang <kan.liang@linux.intel.com>
    Link: http://lkml.kernel.org/r/20190624193711.35241-2-andi@firstfloor.org
    Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 54f43ea4c28742dd3b44ca5517bdb7e08703398b
Author: Rander Wang <rander.wang@linux.intel.com>
Date:   Mon Jul 1 15:46:30 2019 +0800

    ALSA: hda: Fix a headphone detection issue when using SOF

    [ Upstream commit 7c2b3629d09ddec810dc4c1d3a6657c32def8f71 ]

    To save power, the hda hdmi driver in ASoC invokes snd_hdac_ext_bus_link_put
    to disable CORB/RIRB buffers DMA if there is no user of bus and invokes
    snd_hdac_ext_bus_link_get to set up CORB/RIRB buffers when it is used.
    Unsolicited responses is disabled in snd_hdac_bus_stop_cmd_io called by
    snd_hdac_ext_bus_link_put , but it is not enabled in snd_hdac_bus_init_cmd_io
    called by snd_hdac_ext_bus_link_get. So for put-get sequence, Unsolicited
    responses is disabled and headphone can't be detected by hda codecs.

    Now unsolicited responses is only enabled in snd_hdac_bus_reset_link
    which resets controller. The function is only called for setup of
    controller. This patch enables Unsolicited responses after RIRB is
    initialized in snd_hdac_bus_init_cmd_io which works together with
    snd_hdac_bus_reset_link to set up controller.

    Tested legacy hda driver and SOF driver on intel whiskeylake.

    Reviewed-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Rander Wang <rander.wang@linux.intel.com>
```

```
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 74ba703a2575424a4d944d30a2d3e978bc43f765
Author: Michael Chan <michael.chan@broadcom.com>
Date:   Sat Jun 29 11:16:47 2019 -0400

    bnxt_en: Cap the returned MSIX vectors to the RDMA driver.

    [ Upstream commit 1dbc59fa4bbaa108b641cd65a54f662b75e4ed36 ]

    In an earlier commit to improve NQ reservations on 57500 chips, we
    set the resv_irqs on the 57500 VFs to the fixed value assigned by
    the PF regardless of how many are actually used.  The current
    code assumes that resv_irqs minus the ones used by the network driver
    must be the ones for the RDMA driver.  This is no longer true and
    we may return more MSIX vectors than requested, causing inconsistency.
    Fix it by capping the value.

    Fixes: 01989c6b69d9 ("bnxt_en: Improve NQ reservations.")
    Signed-off-by: Michael Chan <michael.chan@broadcom.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 25f88001893442b7bb6cb5af6dc0854e8fdbc8a0
Author: Michael Chan <michael.chan@broadcom.com>
Date:   Sat Jun 29 11:16:46 2019 -0400

    bnxt_en: Fix statistics context reservation logic for RDMA driver.

    [ Upstream commit d77b1ad8e87dc5a6cd0d9158b097a4817946ca3b ]

    The current logic assumes that the RDMA driver uses one statistics
    context adjacent to the ones used by the network driver.  This
    assumption is not true and the statistics context used by the
    RDMA driver is tied to its MSIX base vector.  This wrong assumption
    can cause RDMA driver failure after changing ethtool rings on the
    network side.  Fix the statistics reservation logic accordingly.

    Fixes: 780baad44f0f ("bnxt_en: Reserve 1 stat_ctx for RDMA driver.")
    Signed-off-by: Michael Chan <michael.chan@broadcom.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9935cc5df1c52a194d76acc1c3e8880e359a3268
Author: Michael Chan <michael.chan@broadcom.com>
Date:   Sat Jun 29 11:16:44 2019 -0400

    bnxt_en: Disable bus master during PCI shutdown and driver unload.

    [ Upstream commit c20dc142dd7b2884b8570eeab323bcd4a84294fa ]

    Some chips with older firmware can continue to perform DMA read from
    context memory even after the memory has been freed.  In the PCI shutdown
    method, we need to call pci_disable_device() to shutdown DMA to prevent
    this DMA before we put the device into D3hot.  DMA memory request in
    D3hot state will generate PCI fatal error.  Similarly, in the driver
    remove method, the context memory should only be freed after DMA has
    been shutdown for correctness.

    Fixes: 98f04cf0f1fc ("bnxt_en: Check context memory requirements from firmware.")
    Signed-off-by: Michael Chan <michael.chan@broadcom.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 6ea320766c8be9f50694a97ae3dbfa86b7739476
Author: Shahar S Matityahu <shahar.s.matityahu@intel.com>
Date:   Thu May 16 10:31:17 2019 +0300

    iwlwifi: dbg: fix debug monitor stop and restart delays

    [ Upstream commit fc838c775f35e272e5cc7ef43853f0b55babbe37 ]

    The driver should delay only in recording stop flow between writing to
    DBGC_IN_SAMPLE register and DBGC_OUT_CTRL register. Any other delay is
    not needed.

    Change the following:
    1. Remove any unnecessary delays in the flow
    2. Increase the delay in the stop recording flow since 100 micro is
       not enough
    3. Use usleep_range instead of delay since the driver is allowed to
       sleep in this flow.

    Signed-off-by: Shahar S Matityahu <shahar.s.matityahu@intel.com>
    Fixes: 5cfe79c8d92a ("iwlwifi: fw: stop and start debugging using host command")
    Signed-off-by: Luca Coelho <luciano.coelho@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit fba9941e8248bc02487d49a82c6cfae63ebc690d
Author: He Zhe <zhe.he@windriver.com>
Date:   Mon Jun 24 11:17:38 2019 +0800

    netfilter: Fix remainder of pseudo-header protocol 0

    [ Upstream commit 5d1549847c76b1ffcf8e388ef4d0f229bdd1d7e8 ]

    Since v5.1-rc1, some types of packets do not get unreachable reply with the
    following iptables setting. Fox example,

    $ iptables -A INPUT -p icmp --icmp-type 8 -j REJECT
    $ ping 127.0.0.1 -c 1
    PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
    — 127.0.0.1 ping statistics —
    1 packets transmitted, 0 received, 100% packet loss, time 0ms

    We should have got the following reply from command line, but we did not.
    From 127.0.0.1 icmp_seq=1 Destination Port Unreachable

    Yi Zhao reported it and narrowed it down to:
    7fc38225363d ("netfilter: reject: skip csum verification for protocols that don't
support it"),

    This is because nf_ip_checksum still expects pseudo-header protocol type 0 for
    packets that are of neither TCP or UDP, and thus ICMP packets are mistakenly
    treated as TCP/UDP.

    This patch corrects the conditions in nf_ip_checksum and all other places that
    still call it with protocol 0.

    Fixes: 7fc38225363d ("netfilter: reject: skip csum verification for protocols that don't
support it")
    Reported-by: Yi Zhao <yi.zhao@windriver.com>
    Signed-off-by: He Zhe <zhe.he@windriver.com>
    Signed-off-by: Pablo Neira Ayuso <pablo@netfilter.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 85965e2caf5d37bdf0a453b168b13db1c2f75adb
Author: Baruch Siach <baruch@tkos.co.il>
Date:   Fri Jun 28 07:08:45 2019 +0300
```

```
    bpf: fix uapi bpf_prog_info fields alignment

    [ Upstream commit 0472301a28f6cf53a6bc5783e48a2d0bbff4682f ]

    Merge commit 1c8c5a9d38f60 ("Merge
    git://git.kernel.org/pub/scm/linux/kernel/git/davem/net-next") undid the
    fix from commit 36f9814a494 ("bpf: fix uapi hole for 32 bit compat
    applications") by taking the gpl_compatible 1-bit field definition from
    commit b85fab0e67b162 ("bpf: Add gpl_compatible flag to struct
    bpf_prog_info") as is. That breaks architectures with 16-bit alignment
    like m68k. Add 31-bit pad after gpl_compatible to restore alignment of
    following fields.

    Thanks to Dmitry V. Levin his analysis of this bug history.

    Signed-off-by: Baruch Siach <baruch@tkos.co.il>
    Acked-by: Song Liu <songliubraving@fb.com>
    Cc: Jiri Olsa <jolsa@kernel.org>
    Cc: Daniel Borkmann <daniel@iogearbox.net>
    Cc: Geert Uytterhoeven <geert@linux-m68k.org>
    Cc: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Daniel Borkmann <daniel@iogearbox.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 735e6867dd66e45f9d652e7302f7de707db1d0cc
Author: Andrei Otcheretianski <andrei.otcheretianski@intel.com>
Date:   Mon Apr 15 16:45:04 2019 +0300

    iwlwifi: mvm: Drop large non sta frames

    [ Upstream commit ac70499ee97231a418dc1a4d6c9dc102e8f64631 ]

    In some buggy scenarios we could possible attempt to transmit frames larger
    than maximum MSDU size. Since our devices don't know how to handle this,
    it may result in asserts, hangs etc.
    This can happen, for example, when we receive a large multicast frame
    and try to transmit it back to the air in AP mode.
    Since in a legal scenario this should never happen, drop such frames and
    warn about it.

    Signed-off-by: Andrei Otcheretianski <andrei.otcheretianski@intel.com>
    Signed-off-by: Luca Coelho <luciano.coelho@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2b4d50ae998bcfb35c13c4d154e9fb05656e72ec
Author: Dann Frazier <dann.frazier@canonical.com>
Date:   Wed May 22 17:22:58 2019 -0600

    ixgbe: Avoid NULL pointer dereference with VF on non-IPsec hw

    [ Upstream commit 92924064106e410cdc015f1dbfc0499309f9f5b1 ]

    An ipsec structure will not be allocated if the hardware does not support
    offload. Fixes the following Oops:

    [  191.045452] Unable to handle kernel NULL pointer dereference at virtual address
0000000000000000
    [  191.054232] Mem abort info:
    [  191.057014]   ESR = 0x96000004
    [  191.060057]   Exception class = DABT (current EL), IL = 32 bits
    [  191.065963]   SET = 0, FnV = 0
    [  191.069004]   EA = 0, S1PTW = 0
    [  191.072132] Data abort info:
    [  191.074999]   ISV = 0, ISS = 0x00000004
```

```
    [  191.078822]   CM = 0, WnR = 0
    [  191.081780] user pgtable: 4k pages, 48-bit VAs, pgdp = 0000000043d9e467
    [  191.088382] [0000000000000000] pgd=0000000000000000
    [  191.093252] Internal error: Oops: 96000004 [#1] SMP
    [  191.098119] Modules linked in: vhost_net vhost tap vfio_pci vfio_virqfd
 vfio_iommu_type1 vfio xt_CHECKSUM iptable_mangle ipt_MASQUERADE iptable_nat nf_nat_ipv4
 nf_nat xt_conntrack nf_conntrack nf_defrag_ipv6 nf_defrag_ipv4 ipt_REJECT nf_reject_ipv4
 xt_tcpudp bridge stp llc ebtable_filter devlink ebtables ip6table_filter ip6_tables
 iptable_filter bpfilter ipmi_ssif nls_iso8859_1 input_leds joydev ipmi_si hns_roce_hw_v2
 ipmi_devintf hns_roce ipmi_msghandler cppc_cpufreq sch_fq_codel ib_iser rdma_cm iw_cm ib_cm
 ib_core iscsi_tcp libiscsi_tcp libiscsi scsi_transport_iscsi ip_tables x_tables autofs4 ses
 enclosure btrfs zstd_compress raid10 raid456 async_raid6_recov async_memcpy async_pq
 async_xor async_tx xor hid_generic usbhid hid raid6_pq libcrc32c raid1 raid0 multipath
 linear ixgbevf hibmc_drm ttm
    [  191.168607]  drm_kms_helper aes_ce_blk aes_ce_cipher syscopyarea crct10dif_ce
 sysfillrect ghash_ce qla2xxx sysimgblt sha2_ce sha256_arm64 hisi_sas_v3_hw fb_sys_fops
 sha1_ce uas nvme_fc mpt3sas ixgbe drm hisi_sas_main nvme_fabrics usb_storage hclge
 scsi_transport_fc ahci libsas hnae3 raid_class libahci xfrm_algo scsi_transport_sas mdio
 aes_neon_bs aes_neon_blk crypto_simd cryptd aes_arm64
    [  191.202952] CPU: 94 PID: 0 Comm: swapper/94 Not tainted 4.19.0-rc1+ #11
    [  191.209553] Hardware name: Huawei D06 /D06, BIOS Hisilicon D06 UEFI RC0 - V1.20.01
 04/26/2019
    [  191.218064] pstate: 20400089 (nzCv daIf +PAN -UAO)
    [  191.222873] pc : ixgbe_ipsec_vf_clear+0x60/0xd0 [ixgbe]
    [  191.228093] lr : ixgbe_msg_task+0x2d0/0x1088 [ixgbe]
    [  191.233044] sp : ffff000009b3bcd0
    [  191.236346] x29: ffff000009b3bcd0 x28: 0000000000000000
    [  191.241647] x27: ffff000009628000 x26: 0000000000000000
    [  191.246946] x25: ffff803f652d7600 x24: 0000000000000004
    [  191.252246] x23: ffff803f6a718900 x22: 0000000000000000
    [  191.257546] x21: 0000000000000000 x20: 0000000000000000
    [  191.262845] x19: 0000000000000000 x18: 0000000000000000
    [  191.268144] x17: 0000000000000000 x16: 0000000000000000
    [  191.273443] x15: 0000000000000000 x14: 0000000100000026
    [  191.278742] x13: 0000000100000025 x12: ffff8a5f7fbe0df0
    [  191.284042] x11: 000000010000000b x10: 0000000000000040
    [  191.289341] x9 : 0000000000001100 x8 : ffff803f6a824fd8
    [  191.294640] x7 : ffff803f6a825098 x6 : 0000000000000001
    [  191.299939] x5 : ffff000000f0ffc0 x4 : 0000000000000000
    [  191.305238] x3 : ffff000028c00000 x2 : ffff803f652d7600
    [  191.310538] x1 : 0000000000000000 x0 : ffff000000f205f0
    [  191.315838] Process swapper/94 (pid: 0, stack limit = 0x00000000addfed5a)
    [  191.322613] Call trace:
    [  191.325055]  ixgbe_ipsec_vf_clear+0x60/0xd0 [ixgbe]
    [  191.329927]  ixgbe_msg_task+0x2d0/0x1088 [ixgbe]
    [  191.334536]  ixgbe_msix_other+0x274/0x330 [ixgbe]
    [  191.339233]  __handle_irq_event_percpu+0x78/0x270
    [  191.343924]  handle_irq_event_percpu+0x40/0x98
    [  191.348355]  handle_irq_event+0x50/0xa8
    [  191.352180]  handle_fasteoi_irq+0xbc/0x148
    [  191.356263]  generic_handle_irq+0x34/0x50
    [  191.360259]  __handle_domain_irq+0x68/0xc0
    [  191.364343]  gic_handle_irq+0x84/0x180
    [  191.368079]  el1_irq+0xe8/0x180
    [  191.371208]  arch_cpu_idle+0x30/0x1a8
    [  191.374860]  do_idle+0x1dc/0x2a0
    [  191.378077]  cpu_startup_entry+0x2c/0x30
    [  191.381988]  secondary_start_kernel+0x150/0x1e0
    [  191.386506] Code: 6b15003f 54000320 f1404a9f 54000060 (79400260)

    Fixes: eda0333ac2930 ("ixgbe: add VF IPsec management")
    Signed-off-by: Dann Frazier <dann.frazier@canonical.com>
    Acked-by: Shannon Nelson <snelson@pensando.io>
    Tested-by: Andrew Bowers <andrewx.bowers@intel.com>
    Signed-off-by: Jeff Kirsher <jeffrey.t.kirsher@intel.com>
```

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

commit 62073ba5f4db6da0bb69558cbcf9e9f1300acc3c
Author: Marek Vasut <marex@denx.de>
Date:   Sun Jun 23 14:11:43 2019 +0200

    net: ethernet: ti: cpsw: Assign OF node to slave devices

    [ Upstream commit 337d1727a3895775b5e5ef67d3ca0fea2e2ae768 ]

    Assign OF node to CPSW slave devices, otherwise it is not possible to
    bind e.g. DSA switch to them. Without this patch, the DSA code tries
    to find the ethernet device by OF match, but fails to do so because
    the slave device has NULL OF node.

    Signed-off-by: Marek Vasut <marex@denx.de>
    Cc: David S. Miller <davem@davemloft.net>
    Cc: Ivan Khoronzhuk <ivan.khoronzhuk@linaro.org>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 270b7ad3527edbec259887f734608d9645750f9d
Author: Yonglong Liu <liuyonglong@huawei.com>
Date:   Fri Jun 28 19:50:12 2019 +0800

    net: hns3: add Asym Pause support to fix autoneg problem

    [ Upstream commit bc3781edcea017aa1a29abd953b776cdba298ce2 ]

    Local device and link partner config auto-negotiation on both,
    local device config pause frame use as: rx on/tx off,
    link partner config pause frame use as: rx off/tx on.

    We except the result is:
    Local device:
    Autonegotiate:  on
    RX:             on
    TX:             off
    RX negotiated:  on
    TX negotiated:  off

    Link partner:
    Autonegotiate:  on
    RX:             off
    TX:             on
    RX negotiated:  off
    TX negotiated:  on

    But actually, the result of Local device and link partner is both:
    Autonegotiate:  on
    RX:             off
    TX:             off
    RX negotiated:  off
    TX negotiated:  off

    The root cause is that the supported flag is has only Pause,
    reference to the function genphy_config_advert():
    static int genphy_config_advert(struct phy_device *phydev)
    {
            ...
            linkmode_and(phydev->advertising, phydev->advertising,
                         phydev->supported);
            ...
    }
    The pause frame use of link partner is rx off/tx on, so its
```

```
    advertising only set the bit Asym_Pause, and the supported is
    only set the bit Pause, so the result of linkmode_and(), is
    rx off/tx off.

    This patch adds Asym_Pause to the supported flag to fix it.

    Signed-off-by: Yonglong Liu <liuyonglong@huawei.com>
    Signed-off-by: Peng Li <lipeng321@huawei.com>
    Signed-off-by: Huazhong Tan <tanhuazhong@huawei.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ee0725334e604ce88e76b97868652973439e4694
Author: Vedang Patel <vedang.patel@intel.com>
Date:   Tue Jun 25 15:07:12 2019 -0700

    igb: clear out skb->tstamp after reading the txtime

    [ Upstream commit 1e08511d5d01884a3c9070afd52a47799312074a ]

    If a packet which is utilizing the launchtime feature (via SO_TXTIME socket
    option) also requests the hardware transmit timestamp, the hardware
    timestamp is not delivered to the userspace. This is because the value in
    skb->tstamp is mistaken as the software timestamp.

    Applications, like ptp4l, request a hardware timestamp by setting the
    SOF_TIMESTAMPING_TX_HARDWARE socket option. Whenever a new timestamp is
    detected by the driver (this work is done in igb_ptp_tx_work() which calls
    igb_ptp_tx_hwtstamps() in igb_ptp.c[1]), it will queue the timestamp in the
    ERR_QUEUE for the userspace to read. When the userspace is ready, it will
    issue a recvmsg() call to collect this timestamp.  The problem is in this
    recvmsg() call. If the skb->tstamp is not cleared out, it will be
    interpreted as a software timestamp and the hardware tx timestamp will not
    be successfully sent to the userspace. Look at skb_is_swtx_tstamp() and the
    callee function __sock_recv_timestamp() in net/socket.c for more details.

    Signed-off-by: Vedang Patel <vedang.patel@intel.com>
    Tested-by: Aaron Brown <aaron.f.brown@intel.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5a75d06e67c86b098474039407354dfb688d8c35
Author: Maxime Chevallier <maxime.chevallier@bootlin.com>
Date:   Thu Jun 20 11:42:45 2019 +0200

    net: mvpp2: prs: Don't override the sign bit in SRAM parser shift

    [ Upstream commit 8ec3ede559956f8ad58db7b57d25ac724bab69e9 ]

    The Header Parser allows identifying various fields in the packet
    headers, used for various kind of filtering and classification
    steps.

    This is a re-entrant process, where the offset in the packet header
    depends on the previous lookup results. This offset is represented in
    the SRAM results of the TCAM, as a shift to be operated.

    This shift can be negative in some cases, such as in IPv6 parsing.

    This commit prevents overriding the sign bit when setting the shift
    value, which could cause instabilities when parsing IPv6 flows.

    Fixes: 3f518509dedc ("ethernet: Add new driver for Marvell Armada 375 network unit")
    Suggested-by: Alan Winkowski <walan@marvell.com>
    Signed-off-by: Maxime Chevallier <maxime.chevallier@bootlin.com>
```

```
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0071ff72a496c1bda5faee16c043798d24a6ea91
Author: Wen Gong <wgong@codeaurora.org>
Date:   Thu Jun 27 21:21:51 2019 +0300

    ath10k: destroy sdio workqueue while remove sdio module

    [ Upstream commit 3ed39f8e747a7aafeec07bb244f2c3a1bdca5730 ]

    The workqueue need to flush and destory while remove sdio module,
    otherwise it will have thread which is not destory after remove
    sdio modules.

    Tested with QCA6174 SDIO with firmware
    WLAN.RMH.4.4.1-00007-QCARMSWP-1.

    Signed-off-by: Wen Gong <wgong@codeaurora.org>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0c2f0afee5b6ec2ef763d28d5814d8d5c2fb954f
Author: Dundi Raviteja <dundi@codeaurora.org>
Date:   Tue Jun 25 19:55:48 2019 +0530

    ath10k: Fix memory leak in qmi

    [ Upstream commit c709df58832c5f575f0255bea4b09ad477fc62ea ]

    Currently the memory allocated for qmi handle is
    not being freed during de-init which leads to memory leak.

    Free the allocated qmi memory in qmi deinit
    to avoid memory leak.

    Tested HW: WCN3990
    Tested FW: WLAN.HL.3.1-01040-QCAHLSWMTPLZ-1

    Fixes: fda6fee0001e ("ath10k: add QMI message handshake for wcn3990 client")
    Signed-off-by: Dundi Raviteja <dundi@codeaurora.org>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6b3420b506a77a5d2edb02ee6ec1e6845bdd18e8
Author: Yunsheng Lin <linyunsheng@huawei.com>
Date:   Fri Jun 28 19:50:10 2019 +0800

    net: hns3: add some error checking in hclge_tm module

    [ Upstream commit 04f25edb48c441fc278ecc154c270f16966cbb90 ]

    When hdev->tx_sch_mode is HCLGE_FLAG_VNET_BASE_SCH_MODE, the
    hclge_tm_schd_mode_vnet_base_cfg calls hclge_tm_pri_schd_mode_cfg
    with vport->vport_id as pri_id, which is used as index for
    hdev->tm_info.tc_info, it will cause out of bound access issue
    if vport_id is equal to or larger than HNAE3_MAX_TC.

    Also hardware only support maximum speed of HCLGE_ETHER_MAX_RATE.

    So this patch adds two checks for above cases.

    Fixes: 848440544b41 ("net: hns3: Add support of TX Scheduler & Shaper to HNS3 driver")
    Signed-off-by: Yunsheng Lin <linyunsheng@huawei.com>
    Signed-off-by: Peng Li <lipeng321@huawei.com>
```

```
    Signed-off-by: Huazhong Tan <tanhuazhong@huawei.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4202fc26542a765dc865ec666d8a0d95d7d72b45
Author: Yonglong Liu <liuyonglong@huawei.com>
Date:   Fri Jun 28 19:50:11 2019 +0800

    net: hns3: fix a -Wformat-nonliteral compile warning

    [ Upstream commit 18d219b783da61a6cc77581f55fc4af2fa16bc36 ]

    When setting -Wformat=2, there is a compiler warning like this:

    hclge_main.c:xxx:x: warning: format not a string literal and no
    format arguments [-Wformat-nonliteral]
    strs[i].desc);
    ^~~~

    This patch adds missing format parameter "%s" to snprintf() to
    fix it.

    Fixes: 46a3df9f9718 ("Add HNS3 Acceleration Engine & Compatibility Layer Support")
    Signed-off-by: Yonglong Liu <liuyonglong@huawei.com>
    Signed-off-by: Peng Li <lipeng321@huawei.com>
    Signed-off-by: Huazhong Tan <tanhuazhong@huawei.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1ba0d730c0ca6825225171b74721bc75f3d12da8
Author: Coly Li <colyli@suse.de>
Date:   Fri Jun 28 19:59:49 2019 +0800

    bcache: fix potential deadlock in cached_def_free()

    [ Upstream commit 7e865eba00a3df2dc8c4746173a8ca1c1c7f042e ]

    When enable lockdep and reboot system with a writeback mode bcache
    device, the following potential deadlock warning is reported by lockdep
    engine.

    [  101.536569][  T401] kworker/2:2/401 is trying to acquire lock:
    [  101.538575][  T401] 00000000bbf6e6c7 ((wq_completion)bcache_writeback_wq){+.+.}, at:
flush_workqueue+0x87/0x4c0
    [  101.542054][  T401]
    [  101.542054][  T401] but task is already holding lock:
    [  101.544587][  T401] 00000000f5f305b3 ((work_completion)(&cl->work)#2){+.+.}, at:
process_one_work+0x21e/0x640
    [  101.548386][  T401]
    [  101.548386][  T401] which lock already depends on the new lock.
    [  101.548386][  T401]
    [  101.551874][  T401]
    [  101.551874][  T401] the existing dependency chain (in reverse order) is:
    [  101.555000][  T401]
    [  101.555000][  T401] -> #1 ((work_completion)(&cl->work)#2){+.+.}:
    [  101.557860][  T401]        process_one_work+0x277/0x640
    [  101.559661][  T401]        worker_thread+0x39/0x3f0
    [  101.561340][  T401]        kthread+0x125/0x140
    [  101.562963][  T401]        ret_from_fork+0x3a/0x50
    [  101.564718][  T401]
    [  101.564718][  T401] -> #0 ((wq_completion)bcache_writeback_wq){+.+.}:
    [  101.567701][  T401]        lock_acquire+0xb4/0x1c0
    [  101.569651][  T401]        flush_workqueue+0xae/0x4c0
    [  101.571494][  T401]        drain_workqueue+0xa9/0x180
    [  101.573234][  T401]        destroy_workqueue+0x17/0x250
```

```
[  101.575109][  T401]        cached_dev_free+0x44/0x120 [bcache]
[  101.577304][  T401]        process_one_work+0x2a4/0x640
[  101.579357][  T401]        worker_thread+0x39/0x3f0
[  101.581055][  T401]        kthread+0x125/0x140
[  101.582709][  T401]        ret_from_fork+0x3a/0x50
[  101.584592][  T401]
[  101.584592][  T401] other info that might help us debug this:
[  101.584592][  T401]
[  101.588355][  T401]  Possible unsafe locking scenario:
[  101.588355][  T401]
[  101.590974][  T401]        CPU0                    CPU1
[  101.592889][  T401]        ----                    ----
[  101.594743][  T401]   lock((work_completion)(&cl->work)#2);
[  101.596785][  T401]
lock((wq_completion)bcache_writeback_wq);
[  101.600072][  T401]                                lock((work_completion)(&cl-
>work)#2);
[  101.602971][  T401]   lock((wq_completion)bcache_writeback_wq);
[  101.605255][  T401]
[  101.605255][  T401]  *** DEADLOCK ***
[  101.605255][  T401]
[  101.608310][  T401] 2 locks held by kworker/2:2/401:
[  101.610208][  T401]  #0: 00000000cf2c7d17 ((wq_completion)events){+.+.}, at:
process_one_work+0x21e/0x640
[  101.613709][  T401]  #1: 00000000f5f305b3 ((work_completion)(&cl->work)#2){+.+.}, at:
process_one_work+0x21e/0x640
[  101.617480][  T401]
[  101.617480][  T401] stack backtrace:
[  101.619539][  T401] CPU: 2 PID: 401 Comm: kworker/2:2 Tainted: G        W
5.2.0-rc4-lp151.20-default+ #1
[  101.623225][  T401] Hardware name: VMware, Inc. VMware Virtual Platform/440BX Desktop
Reference Platform, BIOS 6.00 04/13/2018
[  101.627210][  T401] Workqueue: events cached_dev_free [bcache]
[  101.629239][  T401] Call Trace:
[  101.630360][  T401]  dump_stack+0x85/0xcb
[  101.631777][  T401]  print_circular_bug+0x19a/0x1f0
[  101.633485][  T401]  __lock_acquire+0x16cd/0x1850
[  101.635184][  T401]  ? __lock_acquire+0x6a8/0x1850
[  101.636863][  T401]  ? lock_acquire+0xb4/0x1c0
[  101.638421][  T401]  ? find_held_lock+0x34/0xa0
[  101.640015][  T401]  lock_acquire+0xb4/0x1c0
[  101.641513][  T401]  ? flush_workqueue+0x87/0x4c0
[  101.643248][  T401]  flush_workqueue+0xae/0x4c0
[  101.644832][  T401]  ? flush_workqueue+0x87/0x4c0
[  101.646476][  T401]  ? drain_workqueue+0xa9/0x180
[  101.648303][  T401]  drain_workqueue+0xa9/0x180
[  101.649867][  T401]  destroy_workqueue+0x17/0x250
[  101.651503][  T401]  cached_dev_free+0x44/0x120 [bcache]
[  101.653328][  T401]  process_one_work+0x2a4/0x640
[  101.655029][  T401]  worker_thread+0x39/0x3f0
[  101.656693][  T401]  ? process_one_work+0x640/0x640
[  101.658501][  T401]  kthread+0x125/0x140
[  101.660012][  T401]  ? kthread_create_worker_on_cpu+0x70/0x70
[  101.661985][  T401]  ret_from_fork+0x3a/0x50
[  101.691318][  T401] bcache: bcache_device_free() bcache0 stopped

    Here is how the above potential deadlock may happen in reboot/shutdown
    code path,
    1) bcache_reboot() is called firstly in the reboot/shutdown code path,
       then in bcache_reboot(), bcache_device_stop() is called.
    2) bcache_device_stop() sets BCACHE_DEV_CLOSING on d->falgs, then call
       closure_queue(&d->cl) to invoke cached_dev_flush(). And in turn
       cached_dev_flush() calls cached_dev_free() via closure_at()
    3) In cached_dev_free(), after stopped writebach kthread
       dc->writeback_thread, the kwork dc->writeback_write_wq is stopping by
```

```
       destroy_workqueue().
    4) Inside destroy_workqueue(), drain_workqueue() is called. Inside
       drain_workqueue(), flush_workqueue() is called. Then wq->lockdep_map
       is acquired by lock_map_acquire() in flush_workqueue(). After the
       lock acquired the rest part of flush_workqueue() just wait for the
       workqueue to complete.
    5) Now we look back at writeback thread routine bch_writeback_thread(),
       in the main while-loop, write_dirty() is called via continue_at() in
       read_dirty_submit(), which is called via continue_at() in while-loop
       level called function read_dirty(). Inside write_dirty() it may be
       re-called on workqueeu dc->writeback_write_wq via continue_at().
       It means when the writeback kthread is stopped in cached_dev_free()
       there might be still one kworker queued on dc->writeback_write_wq
       to execute write_dirty() again.
    6) Now this kworker is scheduled on dc->writeback_write_wq to run by
       process_one_work() (which is called by worker_thread()). Before
       calling the kwork routine, wq->lockdep_map is acquired.
    7) But wq->lockdep_map is acquired already in step 4), so a A-A lock
       (lockdep terminology) scenario happens.

    Indeed on multiple cores syatem, the above deadlock is very rare to
    happen, just as the code comments in process_one_work() says,
    2263     * AFAICT there is no possible deadlock scenario between the
    2264     * flush_work() and complete() primitives (except for
               single-threaded
    2265     * workqueues), so hiding them isn't a problem.

    But it is still good to fix such lockdep warning, even no one running
    bcache on single core system.

    The fix is simple. This patch solves the above potential deadlock by,
    - Do not destroy workqueue dc->writeback_write_wq in cached_dev_free().
    - Flush and destroy dc->writeback_write_wq in writebach kthread routine
      bch_writeback_thread(), where after quit the thread main while-loop
      and before cached_dev_put() is called.

    By this fix, dc->writeback_write_wq will be stopped and destroy before
    the writeback kthread stopped, so the chance for a A-A locking on
    wq->lockdep_map is disappeared, such A-A deadlock won't happen
    any more.

    Signed-off-by: Coly Li <colyli@suse.de>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9af2708055bdaceba9db99435a4f72fc82d42759
Author: Coly Li <colyli@suse.de>
Date:   Fri Jun 28 19:59:46 2019 +0800

    bcache: avoid a deadlock in bcache_reboot()

    [ Upstream commit a59ff6ccc2bf2e2934b31bbf734f0bc04b5ec78a ]

    It is quite frequently to observe deadlock in bcache_reboot() happens
    and hang the system reboot process. The reason is, in bcache_reboot()
    when calling bch_cache_set_stop() and bcache_device_stop() the mutex
    bch_register_lock is held. But in the process to stop cache set and
    bcache device, bch_register_lock will be acquired again. If this mutex
    is held here, deadlock will happen inside the stopping process. The
    aftermath of the deadlock is, whole system reboot gets hung.

    The fix is to avoid holding bch_register_lock for the following loops
    in bcache_reboot(),
           list_for_each_entry_safe(c, tc, &bch_cache_sets, list)
                   bch_cache_set_stop(c);
```

```
            list_for_each_entry_safe(dc, tdc, &uncached_devices, list)
                    bcache_device_stop(&dc->disk);

    A module range variable 'bcache_is_reboot' is added, it sets to true
    in bcache_reboot(). In register_bcache(), if bcache_is_reboot is checked
    to be true, reject the registration by returning -EBUSY immediately.

    Signed-off-by: Coly Li <colyli@suse.de>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit abcad34e02292335831e1132ea5824f7981e2112
Author: Coly Li <colyli@suse.de>
Date:   Fri Jun 28 19:59:25 2019 +0800

    bcache: check c->gc_thread by IS_ERR_OR_NULL in cache_set_flush()

    [ Upstream commit b387e9b58679c60f5b1e4313939bd4878204fc37 ]

    When system memory is in heavy pressure, bch_gc_thread_start() from
    run_cache_set() may fail due to out of memory. In such condition,
    c->gc_thread is assigned to -ENOMEM, not NULL pointer. Then in following
    failure code path bch_cache_set_error(), when cache_set_flush() gets
    called, the code piece to stop c->gc_thread is broken,
             if (!IS_ERR_OR_NULL(c->gc_thread))
                     kthread_stop(c->gc_thread);

    And KASAN catches such NULL pointer deference problem, with the warning
    information:

    [  561.207881] ======================================================================
    [  561.207900] BUG: KASAN: null-ptr-deref in kthread_stop+0x3b/0x440
    [  561.207904] Write of size 4 at addr 000000000000001c by task kworker/15:1/313

    [  561.207913] CPU: 15 PID: 313 Comm: kworker/15:1 Tainted: G        W         5.0.0-
vanilla+ #3
    [  561.207916] Hardware name: Lenovo ThinkSystem SR650 -[7X05CTO1WW]-/-[7X05CTO1WW]-,
BIOS -[IVE136T-2.10]- 03/22/2019
    [  561.207935] Workqueue: events cache_set_flush [bcache]
    [  561.207940] Call Trace:
    [  561.207948]  dump_stack+0x9a/0xeb
    [  561.207955]  ? kthread_stop+0x3b/0x440
    [  561.207960]  ? kthread_stop+0x3b/0x440
    [  561.207965]  kasan_report+0x176/0x192
    [  561.207973]  ? kthread_stop+0x3b/0x440
    [  561.207981]  kthread_stop+0x3b/0x440
    [  561.207995]  cache_set_flush+0xd4/0x6d0 [bcache]
    [  561.208008]  process_one_work+0x856/0x1620
    [  561.208015]  ? find_held_lock+0x39/0x1d0
    [  561.208028]  ? drain_workqueue+0x380/0x380
    [  561.208048]  worker_thread+0x87/0xb80
    [  561.208058]  ? __kthread_parkme+0xb6/0x180
    [  561.208067]  ? process_one_work+0x1620/0x1620
    [  561.208072]  kthread+0x326/0x3e0
    [  561.208079]  ? kthread_create_worker_on_cpu+0xc0/0xc0
    [  561.208090]  ret_from_fork+0x3a/0x50
    [  561.208110] ======================================================================
    [  561.208113] Disabling lock debugging due to kernel taint
    [  561.208115] irq event stamp: 11800231
    [  561.208126] hardirqs last  enabled at (11800231): [<ffffffff83008538>]
do_syscall_64+0x18/0x410
    [  561.208127] BUG: unable to handle kernel NULL pointer dereference at 000000000000001c
    [  561.208129] #PF error: [WRITE]
    [  561.312253] hardirqs last disabled at (11800230): [<ffffffff830052ff>]
```

```
 trace_hardirqs_off_thunk+0x1a/0x1c
     [  561.312259] softirqs last  enabled at (11799832): [<ffffffff850005c7>]
 __do_softirq+0x5c7/0x8c3
     [  561.405975] PGD 0 P4D 0
     [  561.442494] softirqs last disabled at (11799821): [<ffffffff831add2c>]
 irq_exit+0x1ac/0x1e0
     [  561.791359] Oops: 0002 [#1] SMP KASAN NOPTI
     [  561.791362] CPU: 15 PID: 313 Comm: kworker/15:1 Tainted: G    B   W         5.0.0-
 vanilla+ #3
     [  561.791363] Hardware name: Lenovo ThinkSystem SR650 -[7X05CTO1WW]-/-[7X05CTO1WW]-,
 BIOS -[IVE136T-2.10]- 03/22/2019
     [  561.791371] Workqueue: events cache_set_flush [bcache]
     [  561.791374] RIP: 0010:kthread_stop+0x3b/0x440
     [  561.791376] Code: 00 00 65 8b 05 26 d5 e0 7c 89 c0 48 0f a3 05 ec aa df 02 0f 82 dc
 02 00 00 4c 8d 63 20 be 04 00 00 00 4c 89 e7 e8 65 c5 53 00 <f0> ff 43 20 48 8d 7b 24 48 b8
 00 00 00 00 00 fc ff df 48 89 fa 48
     [  561.791377] RSP: 0018:ffff88872fc8fd10 EFLAGS: 00010286
     [  561.838895] bcache: bch_count_io_errors() nvme0n1: IO error on writing btree.
     [  561.838916] bcache: bch_count_io_errors() nvme0n1: IO error on writing btree.
     [  561.838934] bcache: bch_count_io_errors() nvme0n1: IO error on writing btree.
     [  561.838948] bcache: bch_count_io_errors() nvme0n1: IO error on writing btree.
     [  561.838966] bcache: bch_count_io_errors() nvme0n1: IO error on writing btree.
     [  561.838979] bcache: bch_count_io_errors() nvme0n1: IO error on writing btree.
     [  561.838996] bcache: bch_count_io_errors() nvme0n1: IO error on writing btree.
     [  563.067028] RAX: 0000000000000000 RBX: fffffffffffffffc RCX: ffffffff832dd314
     [  563.067030] RDX: 0000000000000000 RSI: 0000000000000004 RDI: 0000000000000297
     [  563.067032] RBP: ffff88872fc8fe88 R08: fffffbfff0b8213d R09: fffffbfff0b8213d
     [  563.067034] R10: 0000000000000001 R11: fffffbfff0b8213c R12: 000000000000001c
     [  563.408618] R13: ffff88dc61cc0f68 R14: ffff888102b94900 R15: ffff88dc61cc0f68
     [  563.408620] FS:  0000000000000000(0000) GS:ffff888f7dc00000(0000)
 knlGS:0000000000000000
     [  563.408622] CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
     [  563.408623] CR2: 000000000000001c CR3: 0000000f48a1a004 CR4: 0000000007606e0
     [  563.408625] DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000
     [  563.408627] DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400
     [  563.904795] bcache: bch_count_io_errors() nvme0n1: IO error on writing btree.
     [  563.915796] PKRU: 55555554
     [  563.915797] Call Trace:
     [  563.915807]  cache_set_flush+0xd4/0x6d0 [bcache]
     [  563.915812]  process_one_work+0x856/0x1620
     [  564.001226] bcache: bch_count_io_errors() nvme0n1: IO error on writing btree.
     [  564.033563]  ? find_held_lock+0x39/0x1d0
     [  564.033567]  ? drain_workqueue+0x380/0x380
     [  564.033574]  worker_thread+0x87/0xb80
     [  564.062823] bcache: bch_count_io_errors() nvme0n1: IO error on writing btree.
     [  564.118042]  ? __kthread_parkme+0xb6/0x180
     [  564.118046]  ? process_one_work+0x1620/0x1620
     [  564.118048]  kthread+0x326/0x3e0
     [  564.118050]  ? kthread_create_worker_on_cpu+0xc0/0xc0
     [  564.167066] bcache: bch_count_io_errors() nvme0n1: IO error on writing btree.
     [  564.252441]  ret_from_fork+0x3a/0x50
     [  564.252447] Modules linked in: msr rpcrdma sunrpc rdma_ucm ib_iser ib_umad rdma_cm
 ib_ipoib i40iw configfs iw_cm ib_cm libiscsi scsi_transport_iscsi mlx4_ib ib_uverbs mlx4_en
 ib_core nls_iso8859_1 nls_cp437 vfat fat intel_rapl skx_edac x86_pkg_temp_thermal coretemp
 iTCO_wdt iTCO_vendor_support crct10dif_pclmul crc32_pclmul crc32c_intel ghash_clmulni_intel
 ses raid0 aesni_intel cdc_ether enclosure usbnet ipmi_ssif joydev aes_x86_64 i40e
 scsi_transport_sas mii bcache md_mod crypto_simd mei_me ioatdma crc64 ptp cryptd pcspkr
 i2c_i801 mlx4_core glue_helper pps_core mei lpc_ich dca wmi ipmi_si ipmi_devintf nd_pmem
 dax_pmem nd_btt ipmi_msghandler device_dax pcc_cpufreq button hid_generic usbhid mgag200
 i2c_algo_bit drm_kms_helper syscopyarea sysfillrect xhci_pci sysimgblt fb_sys_fops xhci_hcd
 ttm megaraid_sas drm usbcore nfit libnvdimm sg dm_multipath dm_mod scsi_dh_rdac scsi_dh_emc
 scsi_dh_alua efivarfs
     [  564.299390] bcache: bch_count_io_errors() nvme0n1: IO error on writing btree.
     [  564.348360] CR2: 000000000000001c
     [  564.348362] ---[ end trace b7f0e5cc7b2103b0 ]---
```

```
    Therefore, it is not enough to only check whether c->gc_thread is NULL,
    we should use IS_ERR_OR_NULL() to check both NULL pointer and error
    value.

    This patch changes the above buggy code piece in this way,
             if (!IS_ERR_OR_NULL(c->gc_thread))
                     kthread_stop(c->gc_thread);

    Signed-off-by: Coly Li <colyli@suse.de>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c586e2a593cd17f6c049863a6f8dbdc20fa401b5
Author: Coly Li <colyli@suse.de>
Date:   Fri Jun 28 19:59:48 2019 +0800

    bcache: acquire bch_register_lock later in cached_dev_free()

    [ Upstream commit 80265d8dfd77792e133793cef44a21323aac2908 ]

    When enable lockdep engine, a lockdep warning can be observed when
    reboot or shutdown system,

    [ 3142.764557][    T1] bcache: bcache_reboot() Stopping all devices:
    [ 3142.776265][ T2649]
    [ 3142.777159][ T2649] ======================================================
    [ 3142.780039][ T2649] WARNING: possible circular locking dependency detected
    [ 3142.782869][ T2649] 5.2.0-rc4-lp151.20-default+ #1 Tainted: G        W
    [ 3142.785684][ T2649] ------------------------------------------------------
    [ 3142.788479][ T2649] kworker/3:67/2649 is trying to acquire lock:
    [ 3142.790738][ T2649] 00000000aaf02291 ((wq_completion)bcache_writeback_wq){+.+.}, at:
flush_workqueue+0x87/0x4c0
    [ 3142.794678][ T2649]
    [ 3142.794678][ T2649] but task is already holding lock:
    [ 3142.797402][ T2649] 000000004fcf89c5 (&bch_register_lock){+.+.}, at:
cached_dev_free+0x17/0x120 [bcache]
    [ 3142.801462][ T2649]
    [ 3142.801462][ T2649] which lock already depends on the new lock.
    [ 3142.801462][ T2649]
    [ 3142.805277][ T2649]
    [ 3142.805277][ T2649] the existing dependency chain (in reverse order) is:
    [ 3142.808902][ T2649]
    [ 3142.808902][ T2649] -> #2 (&bch_register_lock){+.+.}:
    [ 3142.812396][ T2649]        __mutex_lock+0x7a/0x9d0
    [ 3142.814184][ T2649]        cached_dev_free+0x17/0x120 [bcache]
    [ 3142.816415][ T2649]        process_one_work+0x2a4/0x640
    [ 3142.818413][ T2649]        worker_thread+0x39/0x3f0
    [ 3142.820276][ T2649]        kthread+0x125/0x140
    [ 3142.822061][ T2649]        ret_from_fork+0x3a/0x50
    [ 3142.823965][ T2649]
    [ 3142.823965][ T2649] -> #1 ((work_completion)(&cl->work)#2){+.+.}:
    [ 3142.827244][ T2649]        process_one_work+0x277/0x640
    [ 3142.829160][ T2649]        worker_thread+0x39/0x3f0
    [ 3142.830958][ T2649]        kthread+0x125/0x140
    [ 3142.832674][ T2649]        ret_from_fork+0x3a/0x50
    [ 3142.834915][ T2649]
    [ 3142.834915][ T2649] -> #0 ((wq_completion)bcache_writeback_wq){+.+.}:
    [ 3142.838121][ T2649]        lock_acquire+0xb4/0x1c0
    [ 3142.840025][ T2649]        flush_workqueue+0xae/0x4c0
    [ 3142.842035][ T2649]        drain_workqueue+0xa9/0x180
    [ 3142.844042][ T2649]        destroy_workqueue+0x17/0x250
    [ 3142.846142][ T2649]        cached_dev_free+0x52/0x120 [bcache]
    [ 3142.848530][ T2649]        process_one_work+0x2a4/0x640
    [ 3142.850663][ T2649]        worker_thread+0x39/0x3f0
```

```
    [ 3142.852464][ T2649]        kthread+0x125/0x140
    [ 3142.854106][ T2649]        ret_from_fork+0x3a/0x50
    [ 3142.855880][ T2649]
    [ 3142.855880][ T2649] other info that might help us debug this:
    [ 3142.855880][ T2649]
    [ 3142.859663][ T2649] Chain exists of:
    [ 3142.859663][ T2649]   (wq_completion)bcache_writeback_wq --> (work_completion)(&cl-
>work)#2 --> &bch_register_lock
    [ 3142.859663][ T2649]
    [ 3142.865424][ T2649]  Possible unsafe locking scenario:
    [ 3142.865424][ T2649]
    [ 3142.868022][ T2649]        CPU0                    CPU1
    [ 3142.869885][ T2649]        ----                    ----
    [ 3142.871751][ T2649]   lock(&bch_register_lock);
    [ 3142.873379][ T2649]                                lock((work_completion)(&cl-
>work)#2);
    [ 3142.876399][ T2649]                                lock(&bch_register_lock);
    [ 3142.879727][ T2649]   lock((wq_completion)bcache_writeback_wq);
    [ 3142.882064][ T2649]
    [ 3142.882064][ T2649]  *** DEADLOCK ***
    [ 3142.882064][ T2649]
    [ 3142.885060][ T2649] 3 locks held by kworker/3:67/2649:
    [ 3142.887245][ T2649]  #0: 00000000e774cdd0 ((wq_completion)events){+.+.}, at:
process_one_work+0x21e/0x640
    [ 3142.890815][ T2649]  #1: 00000000f7df89da ((work_completion)(&cl->work)#2){+.+.}, at:
process_one_work+0x21e/0x640
    [ 3142.894884][ T2649]  #2: 000000004fcf89c5 (&bch_register_lock){+.+.}, at:
cached_dev_free+0x17/0x120 [bcache]
    [ 3142.898797][ T2649]
    [ 3142.898797][ T2649] stack backtrace:
    [ 3142.900961][ T2649] CPU: 3 PID: 2649 Comm: kworker/3:67 Tainted: G        W
5.2.0-rc4-lp151.20-default+ #1
    [ 3142.904789][ T2649] Hardware name: VMware, Inc. VMware Virtual Platform/440BX Desktop
Reference Platform, BIOS 6.00 04/13/2018
    [ 3142.909168][ T2649] Workqueue: events cached_dev_free [bcache]
    [ 3142.911422][ T2649] Call Trace:
    [ 3142.912656][ T2649]  dump_stack+0x85/0xcb
    [ 3142.914181][ T2649]  print_circular_bug+0x19a/0x1f0
    [ 3142.916193][ T2649]  __lock_acquire+0x16cd/0x1850
    [ 3142.917936][ T2649]  ? __lock_acquire+0x6a8/0x1850
    [ 3142.919704][ T2649]  ? lock_acquire+0xb4/0x1c0
    [ 3142.921335][ T2649]  ? find_held_lock+0x34/0xa0
    [ 3142.923052][ T2649]  lock_acquire+0xb4/0x1c0
    [ 3142.924635][ T2649]  ? flush_workqueue+0x87/0x4c0
    [ 3142.926375][ T2649]  flush_workqueue+0xae/0x4c0
    [ 3142.928047][ T2649]  ? flush_workqueue+0x87/0x4c0
    [ 3142.929824][ T2649]  ? drain_workqueue+0xa9/0x180
    [ 3142.931686][ T2649]  drain_workqueue+0xa9/0x180
    [ 3142.933534][ T2649]  destroy_workqueue+0x17/0x250
    [ 3142.935787][ T2649]  cached_dev_free+0x52/0x120 [bcache]
    [ 3142.937795][ T2649]  process_one_work+0x2a4/0x640
    [ 3142.939803][ T2649]  worker_thread+0x39/0x3f0
    [ 3142.941487][ T2649]  ? process_one_work+0x640/0x640
    [ 3142.943389][ T2649]  kthread+0x125/0x140
    [ 3142.944894][ T2649]  ? kthread_create_worker_on_cpu+0x70/0x70
    [ 3142.947744][ T2649]  ret_from_fork+0x3a/0x50
    [ 3142.970358][ T2649] bcache: bcache_device_free() bcache0 stopped

    Here is how the deadlock happens.
    1) bcache_reboot() calls bcache_device_stop(), then inside
       bcache_device_stop() BCACHE_DEV_CLOSING bit is set on d->flags.
       Then closure_queue(&d->cl) is called to invoke cached_dev_flush().
    2) In cached_dev_flush(), cached_dev_free() is called by continu_at().
    3) In cached_dev_free(), when stopping the writeback kthread of the
       cached device by kthread_stop(), dc->writeback_thread will be waken
```

```
        up to quite the kthread while-loop, then cached_dev_put() is called
        in bch_writeback_thread().
     4) Calling cached_dev_put() in writeback kthread may drop dc->count to
        0, then dc->detach kworker is scheduled, which is initialized as
        cached_dev_detach_finish().
     5) Inside cached_dev_detach_finish(), the last line of code is to call
        closure_put(&dc->disk.cl), which drops the last reference counter of
        closrure dc->disk.cl, then the callback cached_dev_flush() gets
        called.
     Now cached_dev_flush() is called for second time in the code path, the
     first time is in step 2). And again bch_register_lock will be acquired
     again, and a A-A lock (lockdep terminology) is happening.

     The root cause of the above A-A lock is in cached_dev_free(), mutex
     bch_register_lock is held before stopping writeback kthread and other
     kworkers. Fortunately now we have variable 'bcache_is_reboot', which may
     prevent device registration or unregistration during reboot/shutdown
     time, so it is unncessary to hold bch_register_lock such early now.

     This is how this patch fixes the reboot/shutdown time A-A lock issue:
     After moving mutex_lock(&bch_register_lock) to a later location where
     before atomic_read(&dc->running) in cached_dev_free(), such A-A lock
     problem can be solved without any reboot time registration race.

     Signed-off-by: Coly Li <colyli@suse.de>
     Signed-off-by: Jens Axboe <axboe@kernel.dk>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a8dc74d1af9c149e3cc2292c2ebaf79d6329fbc9
Author: Coly Li <colyli@suse.de>
Date:   Fri Jun 28 19:59:36 2019 +0800

    bcache: check CACHE_SET_IO_DISABLE bit in bch_journal()

    [ Upstream commit 383ff2183ad16a8842d1fbd9dd3e1cbd66813e64 ]

    When too many I/O errors happen on cache set and CACHE_SET_IO_DISABLE
    bit is set, bch_journal() may continue to work because the journaling
    bkey might be still in write set yet. The caller of bch_journal() may
    believe the journal still work but the truth is in-memory journal write
    set won't be written into cache device any more. This behavior may
    introduce potential inconsistent metadata status.

    This patch checks CACHE_SET_IO_DISABLE bit at the head of bch_journal(),
    if the bit is set, bch_journal() returns NULL immediately to notice
    caller to know journal does not work.

    Signed-off-by: Coly Li <colyli@suse.de>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 51e5050bf9ccea95bbe8918a811b471bfd311603
Author: Coly Li <colyli@suse.de>
Date:   Fri Jun 28 19:59:35 2019 +0800

    bcache: check CACHE_SET_IO_DISABLE in allocator code

    [ Upstream commit e775339e1ae1205b47d94881db124c11385e597c ]

    If CACHE_SET_IO_DISABLE of a cache set flag is set by too many I/O
    errors, currently allocator routines can still continue allocate
    space which may introduce inconsistent metadata state.

    This patch checkes CACHE_SET_IO_DISABLE bit in following allocator
    routines,
```

```
      - bch_bucket_alloc()
      - __bch_bucket_alloc_set()
      Once CACHE_SET_IO_DISABLE is set on cache set, the allocator routines
      may reject allocation request earlier to avoid potential inconsistent
      metadata.

      Signed-off-by: Coly Li <colyli@suse.de>
      Signed-off-by: Jens Axboe <axboe@kernel.dk>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit c067bf154c846fc9eb6f1e1205031bf9615475c4
  Author: Coly Li <colyli@suse.de>
  Date:   Fri Jun 28 19:59:26 2019 +0800

      bcache: fix return value error in bch_journal_read()

      [ Upstream commit 0ae49cb7aa005ed18fe8f4d6ccf73019b78ac7b2 ]

      When everything is OK in bch_journal_read(), finally the return value
      is returned by,
              return ret;
      which assumes ret will be 0 here. This assumption is wrong when all
      journal buckets as are full and filled with valid journal entries. In
      such cache the last location referencess read_bucket() sets 'ret' to
      1, which means new jset added into jset list. The jset list is list
      'journal' in caller run_cache_set().

      Return 1 to run_cache_set() means something wrong and the cache set
      won't start, but indeed everything is OK.

      This patch changes the line at end of bch_journal_read() to directly
      return 0 since everything if verything is good. Then a bogus error
      is fixed.

      Signed-off-by: Coly Li <colyli@suse.de>
      Signed-off-by: Jens Axboe <axboe@kernel.dk>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 577753b998fae544d6f7683b6243a761a3bf0d11
  Author: Maxim Mikityanskiy <maximmi@mellanox.com>
  Date:   Wed Jun 26 17:35:23 2019 +0300

      net/mlx5e: Attach/detach XDP program safely

      [ Upstream commit e18953240de8b46360a67090c87ee1ef8160b35d ]

      When an XDP program is set, a full reopen of all channels happens in two
      cases:

      1. When there was no program set, and a new one is being set.

      2. When there was a program set, but it's being unset.

      The full reopen is necessary, because the channel parameters may change
      if XDP is enabled or disabled. However, it's performed in an unsafe way:
      if the new channels fail to open, the old ones are already closed, and
      the interface goes down. Use the safe way to switch channels instead.
      The same way is already used for other configuration changes.

      Signed-off-by: Maxim Mikityanskiy <maximmi@mellanox.com>
      Signed-off-by: Tariq Toukan <tariqt@mellanox.com>
      Signed-off-by: Saeed Mahameed <saeedm@mellanox.com>
      Signed-off-by: Daniel Borkmann <daniel@iogearbox.net>
      Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 3a774a136e41fda8a236863a3a75e3bcd1c59a4a
Author: Eiichi Tsukata <devel@etsukata.com>
Date:   Wed Jun 26 14:40:11 2019 +0900

    EDAC: Fix global-out-of-bounds write when setting edac_mc_poll_msec

    [ Upstream commit d8655e7630dafa88bc37f101640e39c736399771 ]

    Commit 9da21b1509d8 ("EDAC: Poll timeout cannot be zero, p2") assumes
    edac_mc_poll_msec to be unsigned long, but the type of the variable still
    remained as int. Setting edac_mc_poll_msec can trigger out-of-bounds
    write.

    Reproducer:

      # echo 1001 > /sys/module/edac_core/parameters/edac_mc_poll_msec

    KASAN report:

      BUG: KASAN: global-out-of-bounds in edac_set_poll_msec+0x140/0x150
      Write of size 8 at addr ffffffffb91b2d00 by task bash/1996

      CPU: 1 PID: 1996 Comm: bash Not tainted 5.2.0-rc6+ #23
      Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS 1.12.0-2.fc30 04/01/2014
      Call Trace:
       dump_stack+0xca/0x13e
       print_address_description.cold+0x5/0x246
       __kasan_report.cold+0x75/0x9a
       ? edac_set_poll_msec+0x140/0x150
       kasan_report+0xe/0x20
       edac_set_poll_msec+0x140/0x150
       ? dimmdev_location_show+0x30/0x30
       ? vfs_lock_file+0xe0/0xe0
       ? _raw_spin_lock+0x87/0xe0
       param_attr_store+0x1b5/0x310
       ? param_array_set+0x4f0/0x4f0
       module_attr_store+0x58/0x80
       ? module_attr_show+0x80/0x80
       sysfs_kf_write+0x13d/0x1a0
       kernfs_fop_write+0x2bc/0x460
       ? sysfs_kf_bin_read+0x270/0x270
       ? kernfs_notify+0x1f0/0x1f0
       __vfs_write+0x81/0x100
       vfs_write+0x1e1/0x560
       ksys_write+0x126/0x250
       ? __ia32_sys_read+0xb0/0xb0
       ? do_syscall_64+0x1f/0x390
       do_syscall_64+0xc1/0x390
       entry_SYSCALL_64_after_hwframe+0x49/0xbe
      RIP: 0033:0x7fa7caa5e970
      Code: 73 01 c3 48 8b 0d 28 d5 2b 00 f7 d8 64 89 01 48 83 c8 ff c3 66 0f 1f 44 00 00 83
3d 99 2d 2c 00 00 75 10 b8 01 00 00 00 04
      RSP: 002b:00007fff6acfdfe8 EFLAGS: 00000246 ORIG_RAX: 0000000000000001
      RAX: ffffffffffffffda RBX: 0000000000000005 RCX: 00007fa7caa5e970
      RDX: 0000000000000005 RSI: 0000000000e95c08 RDI: 0000000000000001
      RBP: 0000000000e95c08 R08: 00007fa7cad1e760 R09: 00007fa7cb36a700
      R10: 0000000000000073 R11: 0000000000000246 R12: 0000000000000005
      R13: 0000000000000001 R14: 00007fa7cad1d600 R15: 0000000000000005

      The buggy address belongs to the variable:
       edac_mc_poll_msec+0x0/0x40

      Memory state around the buggy address:
       ffffffffb91b2c00: 00 00 00 00 fa fa fa fa 00 00 00 00 fa fa fa fa
       ffffffffb91b2c80: 00 00 00 00 fa fa fa fa 00 00 00 00 fa fa fa fa
```

```
       >ffffffffb91b2d00: 04 fa fa fa fa fa fa fa 04 fa fa fa fa fa fa fa
                          ^
        ffffffffb91b2d80: 04 fa fa fa fa fa fa fa 00 00 00 00 00 00 00 00
        ffffffffb91b2e00: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

    Fix it by changing the type of edac_mc_poll_msec to unsigned int.
    The reason why this patch adopts unsigned int rather than unsigned long
    is msecs_to_jiffies() assumes arg to be unsigned int. We can avoid
    integer conversion bugs and unsigned int will be large enough for
    edac_mc_poll_msec.

    Reviewed-by: James Morse <james.morse@arm.com>
    Fixes: 9da21b1509d8 ("EDAC: Poll timeout cannot be zero, p2")
    Signed-off-by: Eiichi Tsukata <devel@etsukata.com>
    Signed-off-by: Tony Luck <tony.luck@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0ecba23bbc730e9b88e8f5d2c30573b39d2095f4
Author: Ahmad Masri <amasri@codeaurora.org>
Date: Sun Jun 16 10:26:07 2019 +0300

    wil6210: drop old event after wmi_call timeout

    [Upstream commit 1a276003111c0404f6bfeffe924c5a21f482428b]

    This change fixes a rare race condition of handling WMI events after
    wmi_call expires.

    wmi_recv_cmd immediately handles an event when reply_buf is defined and
    a wmi_call is waiting for the event.
    However, in case the wmi_call has already timed-out, there will be no
    waiting/running wmi_call and the event will be queued in WMI queue and
    will be handled later in wmi_event_handle.
    Meanwhile, a new similar wmi_call for the same command and event may
    be issued. In this case, when handling the queued event we got WARN_ON
    printed.

    Fixing this case as a valid timeout and drop the unexpected event.

    Signed-off-by: Ahmad Masri <amasri@codeaurora.org>
    Signed-off-by: Maya Erez <merez@codeaurora.org>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a1b5ed07288f0ec7967585929e404b5068903178
Author: Zefir Kurtisi <zefir.kurtisi@neratec.com>
Date: Tue Jun 11 15:36:56 2019 +0200

    ath9k: correctly handle short radar pulses

    [Upstream commit df5c4150501ee7e86383be88f6490d970adcf157]

    In commit 3c0efb745a17 ("ath9k: discard undersized packets")
    the lower bound of RX packets was set to 10 (min ACK size) to
    filter those that would otherwise be treated as invalid at
    mac80211.

    Alas, short radar pulses are reported as PHY_ERROR frames
    with length set to 3. Therefore their detection stopped
    working after that commit.

    NOTE: ath9k drivers built thereafter will not pass DFS
    certification.

    This extends the criteria for short packets to explicitly

```
    handle PHY_ERROR frames.

    Fixes: 3c0efb745a17 ("ath9k: discard undersized packets")
    Signed-off-by: Zefir Kurtisi <zefir.kurtisi@neratec.com>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b8aa9d2a9e60091cec89fda4438a3ee234fbebcb
Author: Arnd Bergmann <arnd@arndb.de>
Date:   Tue Jun 18 14:13:47 2019 +0200

    crypto: asymmetric_keys - select CRYPTO_HASH where needed

    [ Upstream commit 90acc0653d2bee203174e66d519fbaaa513502de ]

    Build testing with some core crypto options disabled revealed
    a few modules that are missing CRYPTO_HASH:

    crypto/asymmetric_keys/x509_public_key.o: In function `x509_get_sig_params':
    x509_public_key.c:(.text+0x4c7): undefined reference to `crypto_alloc_shash'
    x509_public_key.c:(.text+0x5e5): undefined reference to `crypto_shash_digest'
    crypto/asymmetric_keys/pkcs7_verify.o: In function `pkcs7_digest.isra.0':
    pkcs7_verify.c:(.text+0xab): undefined reference to `crypto_alloc_shash'
    pkcs7_verify.c:(.text+0x1b2): undefined reference to `crypto_shash_digest'
    pkcs7_verify.c:(.text+0x3c1): undefined reference to `crypto_shash_update'
    pkcs7_verify.c:(.text+0x411): undefined reference to `crypto_shash_finup'

    This normally doesn't show up in randconfig tests because there is
    a large number of other options that select CRYPTO_HASH.

    Signed-off-by: Arnd Bergmann <arnd@arndb.de>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4c9801c86165024c72e9fabc151f5341178e69ea
Author: Arnd Bergmann <arnd@arndb.de>
Date:   Tue Jun 18 13:19:42 2019 +0200

    crypto: serpent - mark __serpent_setkey_sbox noinline

    [ Upstream commit 473971187d6727609951858c63bf12b0307ef015 ]

    The same bug that gcc hit in the past is apparently now showing
    up with clang, which decides to inline __serpent_setkey_sbox:

    crypto/serpent_generic.c:268:5: error: stack frame size of 2112 bytes in function
'__serpent_setkey' [-Werror,-Wframe-larger-than=]

    Marking it 'noinline' reduces the stack usage from 2112 bytes to
    192 and 96 bytes, respectively, and seems to generate more
    useful object code.

    Fixes: c871c10e4ea7 ("crypto: serpent - improve __serpent_setkey with UBSAN")
    Signed-off-by: Arnd Bergmann <arnd@arndb.de>
    Reviewed-by: Eric Biggers <ebiggers@kernel.org>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e5b1304989c8384cf7c3ab3ae79c3a85fbefdddb
Author: Mauro S. M. Rodrigues <maurosr@linux.vnet.ibm.com>
Date:   Thu May 23 16:11:12 2019 -0300

    ixgbe: Check DDM existence in transceiver before access

    [ Upstream commit 655c91414579d7bb115a4f7898ee726fc18e0984 ]
```

```
    Some transceivers may comply with SFF-8472 but not implement the Digital
    Diagnostic Monitoring (DDM) interface described in it. The existence of
    such area is specified by bit 6 of byte 92, set to 1 if implemented.

    Currently, due to not checking this bit ixgbe fails trying to read SFP
    module's eeprom with the follow message:

    ethtool -m enP51p1s0f0
    Cannot get Module EEPROM data: Input/output error

    Because it fails to read the additional 256 bytes in which it was assumed
    to exist the DDM data.

    This issue was noticed using a Mellanox Passive DAC PN 01FT738. The eeprom
    data was confirmed by Mellanox as correct and present in other Passive
    DACs in from other manufacturers.

    Signed-off-by: "Mauro S. M. Rodrigues" <maurosr@linux.vnet.ibm.com>
    Reviewed-by: Jesse Brandeburg <jesse.brandeburg@intel.com>
    Tested-by: Andrew Bowers <andrewx.bowers@intel.com>
    Signed-off-by: Jeff Kirsher <jeffrey.t.kirsher@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit fd4067966daa72cf199977ff4f877939b93d332c
Author: Jianbo Liu <jianbol@mellanox.com>
Date:   Tue Jun 25 17:47:52 2019 +0000

    net/mlx5: Get vport ACL namespace by vport index

    [ Upstream commit f53297d67800feb5fafd94abd926c889aefee690 ]

    The ingress and egress ACL root namespaces are created per vport and
    stored into arrays. However, the vport number is not the same as the
    index. Passing the array index, instead of vport number, to get the
    correct ingress and egress acl namespace.

    Fixes: 9b93ab981e3b ("net/mlx5: Separate ingress/egress namespaces for each vport")
    Signed-off-by: Jianbo Liu <jianbol@mellanox.com>
    Reviewed-by: Oz Shlomo <ozsh@mellanox.com>
    Reviewed-by: Eli Britstein <elibr@mellanox.com>
    Reviewed-by: Roi Dayan <roid@mellanox.com>
    Reviewed-by: Mark Bloch <markb@mellanox.com>
    Signed-off-by: Saeed Mahameed <saeedm@mellanox.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit af8e2b4872bc990d23556aa8070b57af422fd41f
Author: Jian Shen <shenjian15@huawei.com>
Date:   Thu Jun 20 16:52:38 2019 +0800

    net: hns3: restore the MAC autoneg state after reset

    [ Upstream commit d736fc6c68a5f76e89a6c2c4100e3678706003a3 ]

    When doing global reset, the MAC autoneg state of fibre
    port is set to default, which may cause user configuration
    lost. This patch fixes it by restore the MAC autoneg state
    after reset.

    Fixes: 22f48e24a23d ("net: hns3: add autoneg and change speed support for fibre port")
    Signed-off-by: Jian Shen <shenjian15@huawei.com>
    Signed-off-by: Peng Li <lipeng321@huawei.com>
    Signed-off-by: Huazhong Tan <tanhuazhong@huawei.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 5319277968c1f3d3cc9a3e8c2a804ae9bbf0ea31
Author: Waibel Georg <Georg.Waibel@sensor-technik.de>
Date:   Thu Jun 20 21:37:08 2019 +0000

    gpio: Fix return value mismatch of function gpiod_get_from_of_node()

    [ Upstream commit 025bf37725f1929542361eef2245df30badf242e ]

    In case the requested gpio property is not found in the device tree, some
    callers of gpiod_get_from_of_node() expect a return value of NULL, others
    expect -ENOENT.
    In particular devm_fwnode_get_index_gpiod_from_child() expects -ENOENT.
    Currently it gets a NULL, which breaks the loop that tries all
    gpio_suffixes. The result is that a gpio property is not found, even
    though it is there.

    This patch changes gpiod_get_from_of_node() to return -ENOENT instead
    of NULL when the requested gpio property is not found in the device
    tree. Additionally it modifies all calling functions to properly
    evaluate the return value.

    Another approach would be to leave the return value of
    gpiod_get_from_of_node() as is and fix the bug in
    devm_fwnode_get_index_gpiod_from_child(). Other callers would still need
    to be reworked. The effort would be the same as with the chosen solution.

    Signed-off-by: Georg Waibel <georg.waibel@sensor-technik.de>
    Reviewed-by: Krzysztof Kozlowski <krzk@kernel.org>
    Reviewed-by: Linus Walleij <linus.walleij@linaro.org>
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cc1c99e8a265e5c6cfbdd4255d141590b30d585e
Author: Ferdinand Blomqvist <ferdinand.blomqvist@gmail.com>
Date:   Thu Jun 20 17:10:37 2019 +0300

    rslib: Fix handling of of caller provided syndrome

    [ Upstream commit ef4d6a8556b637ad27c8c2a2cff1dda3da38e9a9 ]

    Check if the syndrome provided by the caller is zero, and act
    accordingly.

    Signed-off-by: Ferdinand Blomqvist <ferdinand.blomqvist@gmail.com>
    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Link: https://lkml.kernel.org/r/20190620141039.9874-6-ferdinand.blomqvist@gmail.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 57d50e607b37fa45646170e2ced268209e01d639
Author: Jiong Wang <jiong.wang@netronome.com>
Date:   Tue Jun 25 17:41:50 2019 +0100

    bpf: fix BPF_ALU32 | BPF_ARSH on BE arches

    [ Upstream commit 75672dda27bd00109a84cd975c17949ad9c45663 ]

    Yauheni reported the following code do not work correctly on BE arches:

           ALU_ARSH_X:
                   DST = (u64) (u32) ((*(s32 *) &DST) >> SRC);
                   CONT;
           ALU_ARSH_K:
                   DST = (u64) (u32) ((*(s32 *) &DST) >> IMM);
                   CONT;
```

```
    and are causing failure of test_verifier test 'arsh32 on imm 2' on BE
    arches.

    The code is taking address and interpreting memory directly, so is not
    endianness neutral. We should instead perform standard C type casting on
    the variable. A u64 to s32 conversion will drop the high 32-bit and reserve
    the low 32-bit as signed integer, this is all we want.

    Fixes: 2dc6b100f928 ("bpf: interpreter support BPF_ALU | BPF_ARSH")
    Reported-by: Yauheni Kaliuta <yauheni.kaliuta@redhat.com>
    Reviewed-by: Jakub Kicinski <jakub.kicinski@netronome.com>
    Reviewed-by: Quentin Monnet <quentin.monnet@netronome.com>
    Signed-off-by: Jiong Wang <jiong.wang@netronome.com>
    Acked-by: Song Liu <songliubraving@fb.com>
    Signed-off-by: Daniel Borkmann <daniel@iogearbox.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 503b1de9c15fe28f7edbcec80a67fe9279bcfa13
Author: Ferdinand Blomqvist <ferdinand.blomqvist@gmail.com>
Date:   Thu Jun 20 17:10:34 2019 +0300

    rslib: Fix decoding of shortened codes

    [ Upstream commit 2034a42d1747fc1e1eeef2c6f1789c4d0762cb9c ]

    The decoding of shortenend codes is broken. It only works as expected if
    there are no erasures.

    When decoding with erasures, Lambda (the error and erasure locator
    polynomial) is initialized from the given erasure positions. The pad
    parameter is not accounted for by the initialisation code, and hence
    Lambda is initialized from incorrect erasure positions.

    The fix is to adjust the erasure positions by the supplied pad.

    Signed-off-by: Ferdinand Blomqvist <ferdinand.blomqvist@gmail.com>
    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Link: https://lkml.kernel.org/r/20190620141039.9874-3-ferdinand.blomqvist@gmail.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b02beb2e50ed7ad5c21b7c66dbaf10d846285309
Author: Nathan Chancellor <natechancellor@gmail.com>
Date:   Tue Jun 25 11:23:52 2019 -0700

    xsk: Properly terminate assignment in xskq_produce_flush_desc

    [ Upstream commit f7019b7b0ad14bde732b8953161994edfc384953 ]

    Clang warns:

    In file included from net/xdp/xsk_queue.c:10:
    net/xdp/xsk_queue.h:292:2: warning: expression result unused
    [-Wunused-value]
            WRITE_ONCE(q->ring->producer, q->prod_tail);
            ^~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    include/linux/compiler.h:284:6: note: expanded from macro 'WRITE_ONCE'
            __u.__val;                                      \
            ~~~ ^~~~~
    1 warning generated.

    The q->prod_tail assignment has a comma at the end, not a semi-colon.
    Fix that so clang no longer warns and everything works as expected.

    Fixes: c497176cb2e4 ("xsk: add Rx receive functions and poll support")
```

```
    Link: https://github.com/ClangBuiltLinux/linux/issues/544
    Signed-off-by: Nathan Chancellor <natechancellor@gmail.com>
    Acked-by: Nick Desaulniers <ndesaulniers@google.com>
    Acked-by: Jonathan Lemon <jonathan.lemon@gmail.com>
    Acked-by: Björn Töpel <bjorn.topel@intel.com>
    Acked-by: Song Liu <songliubraving@fb.com>
    Signed-off-by: Daniel Borkmann <daniel@iogearbox.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 381dc73f8216252904d6578d7229282029aa430d
Author: Felix Kaechele <felix@kaechele.ca>
Date:   Tue Jun 25 16:48:59 2019 -0400

    netfilter: ctnetlink: Fix regression in conntrack entry deletion

    [ Upstream commit e7600865db32b69deb0109b8254244dca592adcf ]

    Commit f8e608982022 ("netfilter: ctnetlink: Resolve conntrack
    L3-protocol flush regression") introduced a regression in which deletion
    of conntrack entries would fail because the L3 protocol information
    is replaced by AF_UNSPEC. As a result the search for the entry to be
    deleted would turn up empty due to the tuple used to perform the search
    is now different from the tuple used to initially set up the entry.

    For flushing the conntrack table we do however want to keep the option
    for nfgenmsg->version to have a non-zero value to allow for newer
    user-space tools to request treatment under the new behavior. With that
    it is possible to independently flush tables for a defined L3 protocol.
    This was introduced with the enhancements in in commit 59c08c69c278
    ("netfilter: ctnetlink: Support L3 protocol-filter on flush").

    Older user-space tools will retain the behavior of flushing all tables
    regardless of defined L3 protocol.

    Fixes: f8e608982022 ("netfilter: ctnetlink: Resolve conntrack L3-protocol flush
regression")
    Suggested-by: Pablo Neira Ayuso <pablo@netfilter.org>
    Signed-off-by: Felix Kaechele <felix@kaechele.ca>
    Signed-off-by: Pablo Neira Ayuso <pablo@netfilter.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9a290609499e84bf2b762cf82d8cea9f4f03e902
Author: Marek Szyprowski <m.szyprowski@samsung.com>
Date:   Thu May 30 12:50:43 2019 +0200

    clocksource/drivers/exynos_mct: Increase priority over ARM arch timer

    [ Upstream commit 6282edb72bed5324352522d732080d4c1b9dfed6 ]

    Exynos SoCs based on CA7/CA15 have 2 timer interfaces: custom Exynos MCT
    (Multi Core Timer) and standard ARM Architected Timers.

    There are use cases, where both timer interfaces are used simultanously.
    One of such examples is using Exynos MCT for the main system timer and
    ARM Architected Timers for the KVM and virtualized guests (KVM requires
    arch timers).

    Exynos Multi-Core Timer driver (exynos_mct) must be however started
    before ARM Architected Timers (arch_timer), because they both share some
    common hardware blocks (global system counter) and turning on MCT is
    needed to get ARM Architected Timer working properly.

    To ensure selecting Exynos MCT as the main system timer, increase MCT
    timer rating. To ensure proper starting order of both timers during
    suspend/resume cycle, increase MCT hotplug priority over ARM Archictected
```

```
    Timers.

    Signed-off-by: Marek Szyprowski <m.szyprowski@samsung.com>
    Reviewed-by: Krzysztof Kozlowski <krzk@kernel.org>
    Reviewed-by: Chanwoo Choi <cw00.choi@samsung.com>
    Signed-off-by: Daniel Lezcano <daniel.lezcano@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c21c81b6207767a4d161dbed698ab88550f08014
Author: Dmitry Osipenko <digetx@gmail.com>
Date:   Tue Jun 18 17:03:56 2019 +0300

    clocksource/drivers/tegra: Restore base address before cleanup

    [ Upstream commit fc9babc2574691d3bbf0428f007b22261fed55c6 ]

    We're adjusting the timer's base for each per-CPU timer to point to the
    actual start of the timer since device-tree defines a compound registers
    range that includes all of the timers. In this case the original base
    need to be restore before calling iounmap to unmap the proper address.

    Signed-off-by: Dmitry Osipenko <digetx@gmail.com>
    Acked-by: Jon Hunter <jonathanh@nvidia.com>
    Acked-by: Thierry Reding <treding@nvidia.com>
    Signed-off-by: Daniel Lezcano <daniel.lezcano@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 848ae42e2332c45381c0297261a0351369a05d42
Author: Tejun Heo <tj@kernel.org>
Date:   Mon Jun 24 09:32:50 2019 -0700

    libata: don't request sense data on !ZAC ATA devices

    [ Upstream commit ca156e006add67e4beea7896be395160735e09b0 ]

    ZAC support added sense data requesting on error for both ZAC and ATA
    devices. This seems to cause erratic error handling behaviors on some
    SSDs where the device reports sense data availability and then
    delivers the wrong content making EH take the wrong actions.  The
    failure mode was sporadic on a LITE-ON ssd and couldn't be reliably
    reproduced.

    There is no value in requesting sense data from non-ZAC ATA devices
    while there's a significant risk of introducing EH misbehaviors which
    are difficult to reproduce and fix.  Let's do the sense data dancing
    only for ZAC devices.

    Reviewed-by: Hannes Reinecke <hare@suse.com>
    Tested-by: Masato Suzuki <masato.suzuki@wdc.com>
    Reviewed-by: Damien Le Moal <damien.lemoal@wdc.com>
    Signed-off-by: Tejun Heo <tj@kernel.org>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8f7bce8f064f3a0822fbafa7c23885e1e544bf0c
Author: Dmitry Osipenko <digetx@gmail.com>
Date:   Mon Jun 3 21:59:43 2019 +0300

    clocksource/drivers/tegra: Release all IRQ's on request_irq() error

    [ Upstream commit 7a3916706e858ad0bc3b5629c68168e1449de26a ]

    Release all requested IRQ's on the request error to properly clean up
    allocated resources.
```

```
    Signed-off-by: Dmitry Osipenko <digetx@gmail.com>
    Acked-By: Peter De Schrijver <pdeschrijver@nvidia.com>
    Signed-off-by: Daniel Lezcano <daniel.lezcano@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 265c3b864dae188fe7682a92b0d1e073e399239e
Author: Paolo Valente <paolo.valente@linaro.org>
Date:   Tue Jun 25 07:12:44 2019 +0200

    block, bfq: fix rq_in_driver check in bfq_update_inject_limit

    [ Upstream commit db599f9ed9bd31b018b6c48ad7c6b21d5b790ecf ]

    One of the cases where the parameters for injection may be updated is
    when there are no more in-flight I/O requests. The number of in-flight
    requests is stored in the field bfqd->rq_in_driver of the descriptor
    bfqd of the device. So, the controlled condition is
    bfqd->rq_in_driver == 0.

    Unfortunately, this is wrong because, the instruction that checks this
    condition is in the code path that handles the completion of a
    request, and, in particular, the instruction is executed before
    bfqd->rq_in_driver is decremented in such a code path.

    This commit fixes this issue by just replacing 0 with 1 in the
    comparison.

    Reported-by: Srivatsa S. Bhat (VMware) <srivatsa@csail.mit.edu>
    Tested-by: Srivatsa S. Bhat (VMware) <srivatsa@csail.mit.edu>
    Signed-off-by: Paolo Valente <paolo.valente@linaro.org>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit fee45bc534f2661d1654f8ef7e4f0a34705f0b5a
Author: Amadeusz Sławiński <amadeuszx.slawinski@linux.intel.com>
Date:   Mon Jun 17 13:36:42 2019 +0200

    ASoC: Intel: hdac_hdmi: Set ops to NULL on remove

    [ Upstream commit 0f6ff78540bd1b4df1e0f17806b0ce2e1dff0d78 ]

    When we unload Skylake driver we may end up calling
    hdac_component_master_unbind(), it uses acomp->audio_ops, which we set
    in hdmi_codec_probe(), so we need to set it to NULL in hdmi_codec_remove(),
    otherwise we will dereference no longer existing pointer.

    Signed-off-by: Amadeusz Sławiński <amadeuszx.slawinski@linux.intel.com>
    Reviewed-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 280aa2675c5fbf108cb04edab91fbfda0b551f75
Author: Kyle Meyer <kyle.meyer@hpe.com>
Date:   Thu Jun 20 14:36:30 2019 -0500

    perf tools: Increase MAX_NR_CPUS and MAX_CACHES

    [ Upstream commit 9f94c7f947e919c343b30f080285af53d0fa9902 ]

    Attempting to profile 1024 or more CPUs with perf causes two errors:

      perf record -a
      [ perf record: Woken up X times to write data ]
      way too many cpu caches..
      [ perf record: Captured and wrote X MB perf.data (X samples) ]
```

```
       perf report -C 1024
       Error: failed to set  cpu bitmap
       Requested CPU 1024 too large. Consider raising MAX_NR_CPUS

       Increasing MAX_NR_CPUS from 1024 to 2048 and redefining MAX_CACHES as
       MAX_NR_CPUS * 4 returns normal functionality to perf:

       perf record -a
       [ perf record: Woken up X times to write data ]
       [ perf record: Captured and wrote X MB perf.data (X samples) ]

       perf report -C 1024
       ...

    Signed-off-by: Kyle Meyer <kyle.meyer@hpe.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>
    Link: http://lkml.kernel.org/r/20190620193630.154025-1-
meyerk@stormcage.eag.rdlabs.hpecorp.net
    Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4b018480e30429fb0c57d7c851bd6ff0991be227
Author: Amadeusz Sławiński <amadeuszx.slawinski@linux.intel.com>
Date:   Mon Jun 17 13:36:35 2019 +0200

    ALSA: hdac: Fix codec name after machine driver is unloaded and reloaded

    [ Upstream commit 8a5b0177a7f6099ff534a4d9ce72673af5c3cade ]

    Currently on each driver reload internal counter is being increased. It
    causes failure to enumerate driver devices, as they have hardcoded:
    .codec_name = "ehdaudio0D2",
    As there is currently no devices with multiple hda codecs and there is
    currently no established way to reliably differentiate, between them,
    always assign bus->idx = 0;

    This fixes a problem when we unload and reload machine driver idx gets
    incremented, so .codec_name would've needed to be set to "ehdaudio1D2"
    after first reload and so on.

    Signed-off-by: Amadeusz Sławiński <amadeuszx.slawinski@linux.intel.com>
    Acked-by: Takashi Iwai <tiwai@suse.de>
    Reviewed-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 6c280573a3c2fe1b114bbb2a1547b6e9bfbcd0d2
Author: Miaoqing Pan <miaoqing@codeaurora.org>
Date:   Thu May 30 09:49:20 2019 +0800

    ath10k: fix PCIE device wake up failed

    [ Upstream commit 011d4111c8c602ea829fa4917af1818eb0500a90 ]

    Observed PCIE device wake up failed after ~120 iterations of
    soft-reboot test. The error message is
    "ath10k_pci 0000:01:00.0: failed to wake up device : -110"

    The call trace as below:
    ath10k_pci_probe -> ath10k_pci_force_wake -> ath10k_pci_wake_wait ->
    ath10k_pci_is_awake
```

```
    Once trigger the device to wake up, we will continuously check the RTC
    state until it returns RTC_STATE_V_ON or timeout.

    But for QCA99x0 chips, we use wrong value for RTC_STATE_V_ON.
    Occasionally, we get 0x7 on the fist read, we thought as a failure
    case, but actually is the right value, also verified with the spec.
    So fix the issue by changing RTC_STATE_V_ON from 0x5 to 0x7, passed
    ~2000 iterations.

    Tested HW: QCA9984

    Signed-off-by: Miaoqing Pan <miaoqing@codeaurora.org>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 76d2350d04c6b285c3e6d8614039e2dabf209666
Author: Miaoqing Pan <miaoqing@codeaurora.org>
Date:   Fri May 24 11:16:22 2019 +0800

    ath10k: fix fw crash by moving chip reset after napi disabled

    [ Upstream commit 08d80e4cd27ba19f9bee9e5f788f9a9fc440a22f ]

    On SMP platform, when continuously running wifi up/down, the napi
    poll can be scheduled during chip reset, which will call
    ath10k_pci_has_fw_crashed() to check the fw status. But in the reset
    period, the value from FW_INDICATOR_ADDRESS register will return
    0xdeadbeef, which also be treated as fw crash. Fix the issue by
    moving chip reset after napi disabled.

    ath10k_pci 0000:01:00.0: firmware crashed! (guid 73b30611-5b1e-4bdd-90b4-64c81eb947b6)
    ath10k_pci 0000:01:00.0: qca9984/qca9994 hw1.0 target 0x01000000 chip_id 0x00000000 sub
168c:cafe
    ath10k_pci 0000:01:00.0: htt-ver 2.2 wmi-op 6 htt-op 4 cal otp max-sta 512 raw 0
hwcrypto 1
    ath10k_pci 0000:01:00.0: failed to get memcpy hi address for firmware address 4: -16
    ath10k_pci 0000:01:00.0: failed to read firmware dump area: -16
    ath10k_pci 0000:01:00.0: Copy Engine register dump:
    ath10k_pci 0000:01:00.0: [00]: 0x0004a000   0   0   0   0
    ath10k_pci 0000:01:00.0: [01]: 0x0004a400   0   0   0   0
    ath10k_pci 0000:01:00.0: [02]: 0x0004a800   0   0   0   0
    ath10k_pci 0000:01:00.0: [03]: 0x0004ac00   0   0   0   0
    ath10k_pci 0000:01:00.0: [04]: 0x0004b000   0   0   0   0
    ath10k_pci 0000:01:00.0: [05]: 0x0004b400   0   0   0   0
    ath10k_pci 0000:01:00.0: [06]: 0x0004b800   0   0   0   0
    ath10k_pci 0000:01:00.0: [07]: 0x0004bc00   1   0   1   0
    ath10k_pci 0000:01:00.0: [08]: 0x0004c000   0   0   0   0
    ath10k_pci 0000:01:00.0: [09]: 0x0004c400   0   0   0   0
    ath10k_pci 0000:01:00.0: [10]: 0x0004c800   0   0   0   0
    ath10k_pci 0000:01:00.0: [11]: 0x0004cc00   0   0   0   0

    Tested HW: QCA9984,QCA9887,WCN3990

    Signed-off-by: Miaoqing Pan <miaoqing@codeaurora.org>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 187d75b567fe8da2f1582f64ed1a2ac0ce81c215
Author: Claire Chang <tientzu@chromium.org>
Date:   Thu May 23 15:15:34 2019 +0800

    ath10k: add missing error handling

    [ Upstream commit 4b553f3ca4cbde67399aa3a756c37eb92145b8a1 ]
```

```
    In function ath10k_sdio_mbox_rx_alloc() [sdio.c],
    ath10k_sdio_mbox_alloc_rx_pkt() is called without handling the error cases.
    This will make the driver think the allocation for skb is successful and
    try to access the skb. If we enable failslab, system will easily crash with
    NULL pointer dereferencing.

    Call trace of CONFIG_FAILSLAB:
    ath10k_sdio_irq_handler+0x570/0xa88 [ath10k_sdio]
    process_sdio_pending_irqs+0x4c/0x174
    sdio_run_irqs+0x3c/0x64
    sdio_irq_work+0x1c/0x28

    Fixes: d96db25d2025 ("ath10k: add initial SDIO support")
    Signed-off-by: Claire Chang <tientzu@chromium.org>
    Reviewed-by: Brian Norris <briannorris@chromium.org>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 64493e8d0f9542e11fdb2f9014c32d87899381ac
Author: Lorenzo Bianconi <lorenzo@kernel.org>
Date:   Tue May 14 11:18:52 2019 +0200

    mt76: mt7615: do not process rx packets if the device is not initialized

    [ Upstream commit 2dcb79cde6129d948a237ef7b48a73a0c82f1e01 ]

    Fix following crash that occurs when the driver is processing rx packets
    while the device is not initialized yet

    $ rmmod mt7615e
    [   67.210261] mt7615e 0000:01:00.0: Message -239 (seq 2) timeout
    $ modprobe mt7615e
    [   72.406937] bus=0x1, slot = 0x0, irq=0x16
    [   72.436590] CPU 0 Unable to handle kernel paging request at virtual address 00000004,
epc == 8eec4240, ra == 8eec41e0
    [   72.450291] mt7615e 0000:01:00.0: Firmware is not ready for download
    [   72.457724] Oops[#1]:
    [   72.470494] mt7615e: probe of 0000:01:00.0 failed with error -5
    [   72.474829] CPU: 0 PID: 0 Comm: swapper/0 Not tainted 4.14.114 #0
    [   72.498702] task: 805769e0 task.stack: 80564000
    [   72.507709] $ 0   : 00000000 00000001 00000000 00000001
    [   72.518106] $ 4   : 8f704dbc 00000000 00000000 8f7046c0
    [   72.528500] $ 8   : 00000024 8045e98c 81210008 11000000
    [   72.538895] $12   : 8fc09f60 00000008 00000019 00000033
    [   72.549289] $16   : 8f704d80 e00000ff 8f0c7800 3c182406
    [   72.559684] $20   : 00000006 8ee615a0 4e000108 00000000
    [   72.570078] $24   : 0000004c 8000cf94
    [   72.580474] $28   : 80564000 8fc09e38 00000001 8eec41e0
    [   72.590869] Hi    : 00000001
    [   72.596582] Lo    : 00000000
    [   72.602319] epc   : 8eec4240 mt7615_mac_fill_rx+0xac/0x494 [mt7615e]
    [   72.614953] ra    : 8eec41e0 mt7615_mac_fill_rx+0x4c/0x494 [mt7615e]
    [   72.627580] Status: 11008403 KERNEL EXL IE
    [   72.635899] Cause : 40800008 (ExcCode 02)
    [   72.643860] BadVA : 00000004
    [   72.649573] PrId  : 0001992f (MIPS 1004Kc)
    [   72.657704] Modules linked in: mt7615e pppoe ppp_async pppox ppp_generic
nf_conntrack_ipv6 mt76x2e mt76x2_common mt76x02_lib mt7603e mt76 mac80211 iptable_nat
ipt_REJECT ipt_MASQUERADE cfg80211 xt_time xt_tcpudp xt_state xt_nat xt_mu]
    [   72.792717] Process swapper/0 (pid: 0, threadinfo=80564000, task=805769e0,
tls=00000000)
    [   72.808799] Stack : 8f0c7800 00000800 8f0c7800 8032b874 00000000 40000000 8f704d80
8ee615a0
    [   72.825428]         8dc88010 00000001 8ee615e0 8eec09b0 8dc88010 8032b914 8f3aee80
```

```
 80567d20
     [   72.842055]         00000000 8ee615e0 40000000 8f0c7800 00000108 8eec9944 00000000
 00000000
     [   72.858682]         80508f10 80510000 00000001 80567d20 8ee615a0 00000000 00000000
 8ee61c00
     [   72.875308]         8ee61c40 00000040 80610000 80580000 00000000 8ee615dc 8ee61a68
 00000001
     [   72.891936]         ...
     [   72.896793] Call Trace:
     [   72.901649] [<8eec4240>] mt7615_mac_fill_rx+0xac/0x494 [mt7615e]
     [   72.913602] [<8eec09b0>] mt7615_queue_rx_skb+0xe4/0x12c [mt7615e]
     [   72.925734] [<8eec9944>] mt76_dma_cleanup+0x390/0x42c [mt76]
     [   72.936988] Code: ae020018  8ea20004  24030001 <94420004> a602002a  8ea20004
 90420000  14430003  a2020034
     [   72.956390]
     [   72.959676] ---[ end trace f176967739edb19f ]---

     Fixes: 04b8e65922f6 ("mt76: add mac80211 driver for MT7615 PCIe-based chipsets")
     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 eb92aca132b2bd1942dcd86993d6e9d76a636887
 Author: Julian Anastasov <ja@ssi.bg>
 Date:   Tue Jun 18 23:07:36 2019 +0300

     ipvs: fix tinfo memory leak in start_sync_thread

     [ Upstream commit 5db7c8b9f9fc2aeec671ae3ca6375752c162e0e7 ]

     syzkaller reports for memory leak in start_sync_thread [1]

     As Eric points out, kthread may start and stop before the
     threadfn function is called, so there is no chance the
     data (tinfo in our case) to be released in thread.

     Fix this by releasing tinfo in the controlling code instead.

     [1]
     BUG: memory leak
     unreferenced object 0xffff8881206bf700 (size 32):
      comm "syz-executor761", pid 7268, jiffies 4294943441 (age 20.470s)
      hex dump (first 32 bytes):
        00 40 7c 09 81 88 ff ff 80 45 b8 21 81 88 ff ff  .@|......E.!....
        00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
      backtrace:
        [<0000000057619e23>] kmemleak_alloc_recursive include/linux/kmemleak.h:55 [inline]
        [<0000000057619e23>] slab_post_alloc_hook mm/slab.h:439 [inline]
        [<0000000057619e23>] slab_alloc mm/slab.c:3326 [inline]
        [<0000000057619e23>] kmem_cache_alloc_trace+0x13d/0x280 mm/slab.c:3553
        [<000000086ce5479>] kmalloc include/linux/slab.h:547 [inline]
        [<000000086ce5479>] start_sync_thread+0x5d2/0xe10
 net/netfilter/ipvs/ip_vs_sync.c:1862
        [<000000001a9229cc>] do_ip_vs_set_ctl+0x4c5/0x780 net/netfilter/ipvs/ip_vs_ctl.c:2402
        [<00000000ece457c8>] nf_sockopt net/netfilter/nf_sockopt.c:106 [inline]
        [<00000000ece457c8>] nf_setsockopt+0x4c/0x80 net/netfilter/nf_sockopt.c:115
        [<0000000942f62d4>] ip_setsockopt net/ipv4/ip_sockglue.c:1258 [inline]
        [<0000000942f62d4>] ip_setsockopt+0x9b/0xb0 net/ipv4/ip_sockglue.c:1238
        [<00000000a56a8ffd>] udp_setsockopt+0x4e/0x90 net/ipv4/udp.c:2616
        [<00000000fa895401>] sock_common_setsockopt+0x38/0x50 net/core/sock.c:3130
        [<0000000095eef4cf>] __sys_setsockopt+0x98/0x120 net/socket.c:2078
        [<000000009747cf88>] __do_sys_setsockopt net/socket.c:2089 [inline]
        [<000000009747cf88>] __se_sys_setsockopt net/socket.c:2086 [inline]
        [<000000009747cf88>] __x64_sys_setsockopt+0x26/0x30 net/socket.c:2086
        [<00000000ded8ba80>] do_syscall_64+0x76/0x1a0 arch/x86/entry/common.c:301
```

```
        [<00000000893b4ac8>] entry_SYSCALL_64_after_hwframe+0x44/0xa9

    Reported-by: syzbot+7e2e50c8adfccd2e5041@syzkaller.appspotmail.com
    Suggested-by: Eric Biggers <ebiggers@kernel.org>
    Fixes: 998e7a76804b ("ipvs: Use kthread_run() instead of doing a double-fork via
kernel_thread()")
    Signed-off-by: Julian Anastasov <ja@ssi.bg>
    Acked-by: Simon Horman <horms@verge.net.au>
    Signed-off-by: Pablo Neira Ayuso <pablo@netfilter.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0f5046af45396a3a842d933862915e43a6af65f1
Author: Lorenzo Bianconi <lorenzo@kernel.org>
Date:   Fri Jun 7 13:48:10 2019 +0200

    mt7601u: fix possible memory leak when the device is disconnected

    [ Upstream commit 23377c200b2eb48a60d0f228b2a2e75ed6ee6060 ]

    When the device is disconnected while passing traffic it is possible
    to receive out of order urbs causing a memory leak since the skb linked
    to the current tx urb is not removed. Fix the issue deallocating the skb
    cleaning up the tx ring. Moreover this patch fixes the following kernel
    warning

    [   57.480771] usb 1-1: USB disconnect, device number 2
    [   57.483451] ------------[ cut here ]------------
    [   57.483462] TX urb mismatch
    [   57.483481] WARNING: CPU: 1 PID: 32 at
drivers/net/wireless/mediatek/mt7601u/dma.c:245 mt7601u_complete_tx+0x165/00
    [   57.483483] Modules linked in:
    [   57.483496] CPU: 1 PID: 32 Comm: kworker/1:1 Not tainted 5.2.0-rc1+ #72
    [   57.483498] Hardware name: QEMU Standard PC (Q35 + ICH9, 2009), BIOS 1.12.0-2.fc30
04/01/2014
    [   57.483502] Workqueue: usb_hub_wq hub_event
    [   57.483507] RIP: 0010:mt7601u_complete_tx+0x165/0x1e0
    [   57.483510] Code: 8b b5 10 04 00 00 8b 8d 14 04 00 00 eb 8b 80 3d b1 cb e1 00 00 75
9e 48 c7 c7 a4 ea 05 82 c6 05 f
    [   57.483513] RSP: 0000:ffffc900000a0d28 EFLAGS: 00010092
    [   57.483516] RAX: 000000000000000f RBX: ffff88802c0a62c0 RCX: ffffc900000a0c2c
    [   57.483518] RDX: 0000000000000000 RSI: 0000000000000000 RDI: ffffffff810a8371
    [   57.483520] RBP: ffff88803ced6858 R08: 0000000000000000 R09: 0000000000000001
    [   57.483540] R10: 0000000000000002 R11: 0000000000000000 R12: 0000000000000046
    [   57.483542] R13: ffff88802c0a6c88 R14: ffff88803baab540 R15: ffff88803a0cc078
    [   57.483548] FS:  0000000000000000(0000) GS:ffff88803eb00000(0000)
knlGS:0000000000000000
    [   57.483550] CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    [   57.483552] CR2: 000055e7f6780100 CR3: 0000000028c86000 CR4: 00000000000006a0
    [   57.483554] DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000
    [   57.483556] DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400
    [   57.483559] Call Trace:
    [   57.483561]  <IRQ>
    [   57.483565]  __usb_hcd_giveback_urb+0x77/0xe0
    [   57.483570]  xhci_giveback_urb_in_irq.isra.0+0x8b/0x140
    [   57.483574]  handle_cmd_completion+0xf5b/0x12c0
    [   57.483577]  xhci_irq+0x1f6/0x1810
    [   57.483581]  ? lockdep_hardirqs_on+0x9e/0x180
    [   57.483584]  ? _raw_spin_unlock_irq+0x24/0x30
    [   57.483588]  __handle_irq_event_percpu+0x3a/0x260
    [   57.483592]  handle_irq_event_percpu+0x1c/0x60
    [   57.483595]  handle_irq_event+0x2f/0x4c
    [   57.483599]  handle_edge_irq+0x7e/0x1a0
    [   57.483603]  handle_irq+0x17/0x20
    [   57.483607]  do_IRQ+0x54/0x110
    [   57.483610]  common_interrupt+0xf/0xf
```

```
      [   57.483612]  </IRQ>

      Acked-by: Jakub Kicinski <kubakici@wp.pl>
      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 df324bab425f70bf2a1d11f5880c88940b7e403d
  Author: Masahiro Yamada <yamada.masahiro@socionext.com>
  Date:   Tue Jun 25 16:26:22 2019 +0900

      x86/build: Add 'set -e' to mkcapflags.sh to delete broken capflags.c

      [ Upstream commit bc53d3d777f81385c1bb08b07bd1c06450ecc2c1 ]

      Without 'set -e', shell scripts continue running even after any
      error occurs. The missed 'set -e' is a typical bug in shell scripting.

      For example, when a disk space shortage occurs while this script is
      running, it actually ends up with generating a truncated capflags.c.

      Yet, mkcapflags.sh continues running and exits with 0. So, the build
      system assumes it has succeeded.

      It will not be re-generated in the next invocation of Make since its
      timestamp is newer than that of any of the source files.

      Add 'set -e' so that any error in this script is caught and propagated
      to the build system.

      Since 9c2af1c7377a ("kbuild: add .DELETE_ON_ERROR special target"),
      make automatically deletes the target on any failure. So, the broken
      capflags.c will be deleted automatically.

      Signed-off-by: Masahiro Yamada <yamada.masahiro@socionext.com>
      Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
      Cc: "H. Peter Anvin" <hpa@zytor.com>
      Cc: Borislav Petkov <bp@alien8.de>
      Link: https://lkml.kernel.org/r/20190625072622.17679-1-yamada.masahiro@socionext.com
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 01f6a9de905a1421e9ffc00cc45174819e6a7659
  Author: Lorenzo Bianconi <lorenzo@kernel.org>
  Date:   Fri Jun 7 13:48:09 2019 +0200

      mt7601u: do not schedule rx_tasklet when the device has been disconnected

      [ Upstream commit 4079e8ccabc3b6d1b503f2376123cb515d14921f ]

      Do not schedule rx_tasklet when the usb dongle is disconnected.
      Moreover do not grub rx_lock in mt7601u_kill_rx since usb_poison_urb
      can run concurrently with urb completion and we can unlink urbs from rx
      ring in any order.
      This patch fixes the common kernel warning reported when
      the device is removed.

      [   24.921354] usb 3-14: USB disconnect, device number 7
      [   24.921593] ------------[ cut here ]------------
      [   24.921594] RX urb mismatch
      [   24.921675] WARNING: CPU: 4 PID: 163 at
  drivers/net/wireless/mediatek/mt7601u/dma.c:200 mt7601u_complete_rx+0xcb/0xd0 [mt7601u]
      [   24.921769] CPU: 4 PID: 163 Comm: kworker/4:2 Tainted: G           OE     4.19.31-
  041931-generic #201903231635
      [   24.921770] Hardware name: To Be Filled By O.E.M. To Be Filled By O.E.M./Z97
  Extreme4, BIOS P1.30 05/23/2014
```

```
     [   24.921782] Workqueue: usb_hub_wq hub_event
     [   24.921797] RIP: 0010:mt7601u_complete_rx+0xcb/0xd0 [mt7601u]
     [   24.921800] RSP: 0018:ffff9bd9cfd03d08 EFLAGS: 00010086
     [   24.921802] RAX: 0000000000000000 RBX: ffff9bd9bf043540 RCX: 0000000000000006
     [   24.921803] RDX: 0000000000000007 RSI: 0000000000000096 RDI: ffff9bd9cfd16420
     [   24.921804] RBP: ffff9bd9cfd03d28 R08: 0000000000000002 R09: 00000000000003a8
     [   24.921805] R10: 0000002f485fca34 R11: 0000000000000000 R12: ffff9bd9bf043c1c
     [   24.921806] R13: ffff9bd9c62fa3c0 R14: 0000000000000082 R15: 0000000000000000
     [   24.921807] FS:  0000000000000000(0000) GS:ffff9bd9cfd00000(0000)
 knlGS:0000000000000000
     [   24.921808] CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
     [   24.921808] CR2: 00007fb2648b0000 CR3: 0000000142c0a004 CR4: 00000000001606e0
     [   24.921809] Call Trace:
     [   24.921812]  <IRQ>
     [   24.921819]  __usb_hcd_giveback_urb+0x8b/0x140
     [   24.921821]  usb_hcd_giveback_urb+0xca/0xe0
     [   24.921828]  xhci_giveback_urb_in_irq.isra.42+0x82/0xf0
     [   24.921834]  handle_cmd_completion+0xe02/0x10d0
     [   24.921837]  xhci_irq+0x274/0x4a0
     [   24.921838]  xhci_msi_irq+0x11/0x20
     [   24.921851]  __handle_irq_event_percpu+0x44/0x190
     [   24.921856]  handle_irq_event_percpu+0x32/0x80
     [   24.921861]  handle_irq_event+0x3b/0x5a
     [   24.921867]  handle_edge_irq+0x80/0x190
     [   24.921874]  handle_irq+0x20/0x30
     [   24.921889]  do_IRQ+0x4e/0xe0
     [   24.921891]  common_interrupt+0xf/0xf
     [   24.921892]  </IRQ>
     [   24.921900] RIP: 0010:usb_hcd_flush_endpoint+0x78/0x180
     [   24.921354] usb 3-14: USB disconnect, device number 7

     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 229632d74a5b87edca4f2a4a099159a885998919
 Author: Ping-Ke Shih <pkshih@realtek.com>
 Date:   Wed May 29 14:57:30 2019 +0800

     rtlwifi: rtl8192cu: fix error handle when usb probe failed

     [ Upstream commit 6c0ed66f1a5b84e2a812c7c2d6571a5621bf3396 ]

     rtl_usb_probe() must do error handle rtl_deinit_core() only if
     rtl_init_core() is done, otherwise goto error_out2.

     | usb 1-1: New USB device strings: Mfr=0, Product=0, SerialNumber=0
     | rtl_usb: reg 0xf0, usbctrl_vendorreq TimeOut! status:0xffffffb9 value=0x0
     | rtl8192cu: Chip version 0x10
     | rtl_usb: reg 0xa, usbctrl_vendorreq TimeOut! status:0xffffffb9 value=0x0
     | rtl_usb: Too few input end points found
     | INFO: trying to register non-static key.
     | the code is fine but needs lockdep annotation.
     | turning off the locking correctness validator.
     | CPU: 0 PID: 12 Comm: kworker/0:1 Not tainted 5.1.0-rc4-319354-g9a33b36 #3
     | 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+0xe8/0x16e lib/dump_stack.c:113
     |   assign_lock_key kernel/locking/lockdep.c:786 [inline]
     |   register_lock_class+0x11b8/0x1250 kernel/locking/lockdep.c:1095
     |   __lock_acquire+0xfb/0x37c0 kernel/locking/lockdep.c:3582
     |   lock_acquire+0x10d/0x2f0 kernel/locking/lockdep.c:4211
```

```
      |   __raw_spin_lock_irqsave include/linux/spinlock_api_smp.h:110 [inline]
      |   _raw_spin_lock_irqsave+0x44/0x60 kernel/locking/spinlock.c:152
      |   rtl_c2hcmd_launcher+0xd1/0x390
      | drivers/net/wireless/realtek/rtlwifi/base.c:2344
      |   rtl_deinit_core+0x25/0x2d0 drivers/net/wireless/realtek/rtlwifi/base.c:574
      |   rtl_usb_probe.cold+0x861/0xa70
      | drivers/net/wireless/realtek/rtlwifi/usb.c:1093
      |   usb_probe_interface+0x31d/0x820 drivers/usb/core/driver.c:361
      |   really_probe+0x2da/0xb10 drivers/base/dd.c:509
      |   driver_probe_device+0x21d/0x350 drivers/base/dd.c:671
      |   __device_attach_driver+0x1d8/0x290 drivers/base/dd.c:778
      |   bus_for_each_drv+0x163/0x1e0 drivers/base/bus.c:454
      |   __device_attach+0x223/0x3a0 drivers/base/dd.c:844
      |   bus_probe_device+0x1f1/0x2a0 drivers/base/bus.c:514
      |   device_add+0xad2/0x16e0 drivers/base/core.c:2106
      |   usb_set_configuration+0xdf7/0x1740 drivers/usb/core/message.c:2021
      |   generic_probe+0xa2/0xda drivers/usb/core/generic.c:210
      |   usb_probe_device+0xc0/0x150 drivers/usb/core/driver.c:266
      |   really_probe+0x2da/0xb10 drivers/base/dd.c:509
      |   driver_probe_device+0x21d/0x350 drivers/base/dd.c:671
      |   __device_attach_driver+0x1d8/0x290 drivers/base/dd.c:778
      |   bus_for_each_drv+0x163/0x1e0 drivers/base/bus.c:454
      |   __device_attach+0x223/0x3a0 drivers/base/dd.c:844
      |   bus_probe_device+0x1f1/0x2a0 drivers/base/bus.c:514
      |   device_add+0xad2/0x16e0 drivers/base/core.c:2106
      |   usb_new_device.cold+0x537/0xccf 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+0x138e/0x3b00 drivers/usb/core/hub.c:5432
      |   process_one_work+0x90f/0x1580 kernel/workqueue.c:2269
      |   worker_thread+0x9b/0xe20 kernel/workqueue.c:2415
      |   kthread+0x313/0x420 kernel/kthread.c:253
      |   ret_from_fork+0x3a/0x50 arch/x86/entry/entry_64.S:352

    Reported-by: syzbot+1fcc5ef45175fc774231@syzkaller.appspotmail.com
    Signed-off-by: Ping-Ke Shih <pkshih@realtek.com>
    Acked-by: Larry Finger <Larry.Finger@lwfinger.net>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 95b09c87f75588a4a4024aedbefcc13a9cda499a
Author: Icenowy Zheng <icenowy@aosc.io>
Date:   Thu Jun 20 15:47:44 2019 +0200

    net: stmmac: sun8i: force select external PHY when no internal one

    [ Upstream commit 0fec7e72ae1391bb2d7527efb54fe6ae88acabce ]

    The PHY selection bit also exists on SoCs without an internal PHY; if it's
    set to 1 (internal PHY, default value) then the MAC will not make use of
    any PHY on such SoCs.

    This problem appears when adapting for H6, which has no real internal PHY
    (the "internal PHY" on H6 is not on-die, but on a co-packaged AC200 chip,
    connected via RMII interface at GPIO bank A).

    Force the PHY selection bit to 0 when the SOC doesn't have an internal PHY,
    to address the problem of a wrong default value.

    Signed-off-by: Icenowy Zheng <icenowy@aosc.io>
    Signed-off-by: Ondrej Jirman <megous@megous.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit d0ce031decee3c68e4b5b81c882f76a912ff2a64
Author: Hans Verkuil <hverkuil@xs4all.nl>
Date:   Thu Jun 20 07:43:41 2019 -0400

    media: hdpvr: fix locking and a missing msleep

    [ Upstream commit 6bc5a4a1927556ff9adce1aa95ea408c95453225 ]

    This driver has three locking issues:

    - The wait_event_interruptible() condition calls hdpvr_get_next_buffer(dev)
      which uses a mutex, which is not allowed. Rewrite with list_empty_careful()
      that doesn't need locking.

    - In hdpvr_read() the call to hdpvr_stop_streaming() didn't lock io_mutex,
      but it should have since stop_streaming expects that.

    - In hdpvr_device_release() io_mutex was locked when calling flush_work(),
      but there it shouldn't take that mutex since the work done by flush_work()
      also wants to lock that mutex.

    There are also two other changes (suggested by Keith):

    - msecs_to_jiffies(4000); (a NOP) should have been msleep(4000).
    - Change v4l2_dbg to v4l2_info to always log if streaming had to be restarted.

    Reported-by: Keith Pyle <kpyle@austin.rr.com>
    Suggested-by: Keith Pyle <kpyle@austin.rr.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 adb1d60e704dc95b57a5e26ee51a8f73ae963122
Author: André Almeida <andrealmeid@collabora.com>
Date:   Mon Jun 17 12:28:02 2019 -0400

    media: vimc: cap: check v4l2_fill_pixfmt return value

    [ Upstream commit 77ae46e11df5c96bb4582633851f838f5d954df4 ]

    v4l2_fill_pixfmt() returns -EINVAL if the pixelformat used as parameter is
    invalid or if the user is trying to use a multiplanar format with the
    singleplanar API. Currently, the vimc_cap_try_fmt_vid_cap() returns such
    value, but vimc_cap_s_fmt_vid_cap() is ignoring it. Fix that and returns
    an error value if vimc_cap_try_fmt_vid_cap() has failed.

    Signed-off-by: André Almeida <andrealmeid@collabora.com>
    Suggested-by: Helen Koike <helen.koike@collabora.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 9a9ff8f128445688f43b9afc1b837a3de4548586
Author: Philipp Zabel <p.zabel@pengutronix.de>
Date:   Tue Jun 18 12:45:22 2019 -0400

    media: coda: increment sequence offset for the last returned frame

    [ Upstream commit b3b7d96817cdb8b6fc353867705275dce8f41ccc ]

    If no more frames are decoded in bitstream end mode, and a previously
    decoded frame has been returned, the firmware still increments the frame
    number. To avoid a sequence number mismatch after decoder restart,
    increment the sequence_offset correction parameter.
```

```
    Signed-off-by: Philipp Zabel <p.zabel@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 0c4616125cc28c908107da6888079c546e93c104
Author: Marco Felsch <m.felsch@pengutronix.de>
Date:   Tue Jun 18 12:45:11 2019 -0400

    media: coda: fix last buffer handling in V4L2_ENC_CMD_STOP

    [ Upstream commit f3775f89852d167990b0d718587774cf00d22ac2 ]

    coda_encoder_cmd() is racy, as the last scheduled picture run worker can
    still be in-flight while the ENC_CMD_STOP command is issued. Depending
    on the exact timing the sequence numbers might already be changed, but
    the last buffer might not have been put on the destination queue yet.

    In this case the current implementation would prematurely wake the
    destination queue with last_buffer_dequeued=true, causing userspace to
    call streamoff before the last buffer is handled.

    Close this race window by synchronizing with the pic_run_worker before
    doing the sequence check.

    Signed-off-by: Marco Felsch <m.felsch@pengutronix.de>
    [l.stach@pengutronix.de: switch to flush_work, reword commit message]
    Signed-off-by: Lucas Stach <l.stach@pengutronix.de>
    Signed-off-by: Philipp Zabel <p.zabel@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 2d8e49ffcb63f23726c561247f9e19b61f6e7ac6
Author: Philipp Zabel <p.zabel@pengutronix.de>
Date:   Tue Jun 18 12:45:10 2019 -0400

    media: coda: fix mpeg2 sequence number handling

    [ Upstream commit 56d159a4ec6d8da7313aac6fcbb95d8fffe689ba ]

    Sequence number handling assumed that the BIT processor frame number
    starts counting at 1, but this is not true for the MPEG-2 decoder,
    which starts at 0. Fix the sequence counter offset detection to handle
    this.

    Signed-off-by: Philipp Zabel <p.zabel@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 2d04fe8a5c590ab70235638fab038456abb67eb7
Author: Ard Biesheuvel <ard.biesheuvel@linaro.org>
Date:   Wed Jun 19 14:18:31 2019 +0200

    acpi/arm64: ignore 5.1 FADTs that are reported as 5.0

    [ Upstream commit 2af22f3ec3ca452f1e79b967f634708ff01ced8a ]

    Some Qualcomm Snapdragon based laptops built to run Microsoft Windows
    are clearly ACPI 5.1 based, given that that is the first ACPI revision
    that supports ARM, and introduced the FADT 'arm_boot_flags' field,
    which has a non-zero field on those systems.

    So in these cases, infer from the ARM boot flags that the FADT must be
```

```
    5.1 or later, and treat it as 5.1.

    Acked-by: Sudeep Holla <sudeep.holla@arm.com>
    Tested-by: Lee Jones <lee.jones@linaro.org>
    Reviewed-by: Graeme Gregory <graeme.gregory@linaro.org>
    Acked-by: Lorenzo Pieralisi <lorenzo.pieralisi@arm.com>
    Acked-by: Hanjun Guo <guohanjun@huawei.com>
    Signed-off-by: Ard Biesheuvel <ard.biesheuvel@linaro.org>
    Signed-off-by: Catalin Marinas <catalin.marinas@arm.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f4fe0e0273aa738ecddba1d72b598a2be84b0f6a
Author: Kuninori Morimoto <kuninori.morimoto.gx@renesas.com>
Date:   Wed Jun 19 10:07:19 2019 +0900

    ASoC: soc-core: call snd_soc_unbind_card() under mutex_lock;

    [ Upstream commit b545542a0b866f7975254e41c595836e9bc0ff2f ]

    commit 34ac3c3eb8f0c07 ("ASoC: core: lock client_mutex while removing
    link components") added mutex_lock() at soc_remove_link_components().

    Is is called from snd_soc_unbind_card()

            snd_soc_unbind_card()
    =>              soc_remove_link_components()
                    soc_cleanup_card_resources()
                            soc_remove_dai_links()
    =>                              soc_remove_link_components()

    And, there are 2 way to call it.

    (1)
            snd_soc_unregister_component()
    **              mutex_lock()
                            snd_soc_component_del_unlocked()
    =>                              snd_soc_unbind_card()
    **              mutex_unlock()

    (2)
            snd_soc_unregister_card()
    =>              snd_soc_unbind_card()

    (1) case is already using mutex_lock() when it calles
    snd_soc_unbind_card(), thus, we will get lockdep warning.

    commit 495f926c68ddb90 ("ASoC: core: Fix deadlock in
    snd_soc_instantiate_card()") tried to fixup it, but still not
    enough. We still have lockdep warning when we try unbind/bind.

    We need mutex_lock() under snd_soc_unregister_card()
    instead of snd_remove_link_components()/snd_soc_unbind_card().

    Fixes: 34ac3c3eb8f0c07 ("ASoC: core: lock client_mutex while removing link components")
    Fixes: 495f926c68ddb90 ("ASoC: core: Fix deadlock in snd_soc_instantiate_card()")
    Signed-off-by: Kuninori Morimoto <kuninori.morimoto.gx@renesas.com>
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8a0e34e49f89e36edb2324826b9f84e729658cde
Author: Robert Jarzmik <robert.jarzmik@free.fr>
Date:   Mon Jun 3 16:01:55 2019 -0400

    media: mt9m111: fix fw-node refactoring
```

```
    [ Upstream commit 8d4e29a51a954b43e06d916772fa4f50b7e5bbd6 ]

    In the patch refactoring the fw-node, the mt9m111 was broken for all
    platform_data based platforms, which were the first aim of this
    driver. Only the devicetree platform are still functional, probably
    because the testing was done on these.

    The result is that -EINVAL is systematically return for such platforms,
    what this patch fixes.

    [Sakari Ailus: Rework this to resolve a merge conflict and use dev_fwnode]

    Fixes: 98480d65c48c ("media: mt9m111: allow to setup pixclk polarity")
    Signed-off-by: Robert Jarzmik <robert.jarzmik@free.fr>
    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 c0730f79dc9742975a5a4968a315289fb416b6e3
Author: Nathan Huckleberry <nhuck@google.com>
Date:   Fri Jun 14 11:16:04 2019 -0700

    timer_list: Guard procfs specific code

    [ Upstream commit a9314773a91a1d3b36270085246a6715a326ff00 ]

    With CONFIG_PROC_FS=n the following warning is emitted:

    kernel/time/timer_list.c:361:36: warning: unused variable
    'timer_list_sops' [-Wunused-const-variable]
       static const struct seq_operations timer_list_sops = {

    Add #ifdef guard around procfs specific code.

    Signed-off-by: Nathan Huckleberry <nhuck@google.com>
    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Reviewed-by: Nick Desaulniers <ndesaulniers@google.com>
    Cc: john.stultz@linaro.org
    Cc: sboyd@kernel.org
    Cc: clang-built-linux@googlegroups.com
    Link: https://github.com/ClangBuiltLinux/linux/issues/534
    Link: https://lkml.kernel.org/r/20190614181604.112297-1-nhuck@google.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b2e049098962d6fdc2d79b5bcaf816196c837e79
Author: Miroslav Lichvar <mlichvar@redhat.com>
Date:   Tue Jun 18 17:47:13 2019 +0200

    ntp: Limit TAI-UTC offset

    [ Upstream commit d897a4ab11dc8a9fda50d2eccc081a96a6385998 ]

    Don't allow the TAI-UTC offset of the system clock to be set by adjtimex()
    to a value larger than 100000 seconds.

    This prevents an overflow in the conversion to int, prevents the CLOCK_TAI
    clock from getting too far ahead of the CLOCK_REALTIME clock, and it is
    still large enough to allow leap seconds to be inserted at the maximum rate
    currently supported by the kernel (once per day) for the next ~270 years,
    however unlikely it is that someone can survive a catastrophic event which
    slowed down the rotation of the Earth so much.

    Reported-by: Weikang shi <swkhack@gmail.com>
    Signed-off-by: Miroslav Lichvar <mlichvar@redhat.com>
    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
```

```
    Cc: John Stultz <john.stultz@linaro.org>
    Cc: Prarit Bhargava <prarit@redhat.com>
    Cc: Richard Cochran <richardcochran@gmail.com>
    Cc: Stephen Boyd <sboyd@kernel.org>
    Link: https://lkml.kernel.org/r/20190618154713.20929-1-mlichvar@redhat.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 41e612fa1c63f80adf1774c6d8953f7b60a9cc84
Author: Anders Roxell <anders.roxell@linaro.org>
Date:   Wed Jun 12 12:19:35 2019 -0400

    media: i2c: fix warning same module names

    [ Upstream commit b2ce5617dad254230551feda3599f2cc68e53ad8 ]

    When building with CONFIG_VIDEO_ADV7511 and CONFIG_DRM_I2C_ADV7511
    enabled as loadable modules, we see the following warning:

      drivers/gpu/drm/bridge/adv7511/adv7511.ko
      drivers/media/i2c/adv7511.ko

    Rework so that the file is named adv7511-v4l2.c.

    Signed-off-by: Anders Roxell <anders.roxell@linaro.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 f37485cd10b201f9c9d86e3bcef2427ec9fabd7b
Author: Marek Szyprowski <m.szyprowski@samsung.com>
Date:   Thu Jun 13 06:48:34 2019 -0400

    media: s5p-mfc: Make additional clocks optional

    [ Upstream commit e08efef8fe7db87206314c19b341612c719f891a ]

    Since the beginning the second clock ('special', 'sclk') was optional and
    it is not available on some variants of Exynos SoCs (i.e. Exynos5420 with
    v7 of MFC hardware).

    However commit 1bce6fb3edf1 ("[media] s5p-mfc: Rework clock handling")
    made handling of all specified clocks mandatory. This patch restores
    original behavior of the driver and fixes its operation on
    Exynos5420 SoCs.

    Fixes: 1bce6fb3edf1 ("[media] s5p-mfc: Rework clock handling")
    Signed-off-by: Marek Szyprowski <m.szyprowski@samsung.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 d53145e8f38c0621f46dccec7c9974d781fd899e
Author: Julian Anastasov <ja@ssi.bg>
Date:   Tue Jun 4 21:56:35 2019 +0300

    ipvs: defer hook registration to avoid leaks

    [ Upstream commit cf47a0b882a4e5f6b34c7949d7b293e9287f1972 ]

    syzkaller reports for memory leak when registering hooks [1]

    As we moved the nf_unregister_net_hooks() call into
    __ip_vs_dev_cleanup(), defer the nf_register_net_hooks()
    call, so that hooks are allocated and freed from same
    pernet_operations (ipvs_core_dev_ops).
```

```
    [1]
    BUG: memory leak
    unreferenced object 0xffff88810acd8a80 (size 96):
     comm "syz-executor073", pid 7254, jiffies 4294950560 (age 22.250s)
     hex dump (first 32 bytes):
       02 00 00 00 00 00 00 00 50 8b bb 82 ff ff ff ff  ........P.......
       00 00 00 00 00 00 00 00 00 77 bb 82 ff ff ff ff  .........w......
     backtrace:
       [<0000000013db61f1>] kmemleak_alloc_recursive include/linux/kmemleak.h:55 [inline]
       [<0000000013db61f1>] slab_post_alloc_hook mm/slab.h:439 [inline]
       [<0000000013db61f1>] slab_alloc_node mm/slab.c:3269 [inline]
       [<0000000013db61f1>] kmem_cache_alloc_node_trace+0x15b/0x2a0 mm/slab.c:3597
       [<000000001a27307d>] __do_kmalloc_node mm/slab.c:3619 [inline]
       [<000000001a27307d>] __kmalloc_node+0x38/0x50 mm/slab.c:3627
       [<0000000025054add>] kmalloc_node include/linux/slab.h:590 [inline]
       [<0000000025054add>] kvmalloc_node+0x4a/0xd0 mm/util.c:431
       [<0000000050d1bc00>] kvmalloc include/linux/mm.h:637 [inline]
       [<0000000050d1bc00>] kvzalloc include/linux/mm.h:645 [inline]
       [<0000000050d1bc00>] allocate_hook_entries_size+0x3b/0x60 net/netfilter/core.c:61
       [<00000000e8abe142>] nf_hook_entries_grow+0xae/0x270 net/netfilter/core.c:128
       [<000000004b94797c>] __nf_register_net_hook+0x9a/0x170 net/netfilter/core.c:337
       [<00000000d1545cbc>] nf_register_net_hook+0x34/0xc0 net/netfilter/core.c:464
       [<00000000876c9b55>] nf_register_net_hooks+0x53/0xc0 net/netfilter/core.c:480
       [<000000002ea868e0>] __ip_vs_init+0xe8/0x170 net/netfilter/ipvs/ip_vs_core.c:2280
       [<000000002eb2d451>] ops_init+0x4c/0x140 net/core/net_namespace.c:130
       [<000000000284ec48>] setup_net+0xde/0x230 net/core/net_namespace.c:316
       [<00000000a70600fa>] copy_net_ns+0xf0/0x1e0 net/core/net_namespace.c:439
       [<00000000ff26c15e>] create_new_namespaces+0x141/0x2a0 kernel/nsproxy.c:107
       [<00000000b103dc79>] copy_namespaces+0xa1/0xe0 kernel/nsproxy.c:165
       [<000000007cc008a2>] copy_process.part.0+0x11fd/0x2150 kernel/fork.c:2035
       [<00000000c344af7c>] copy_process kernel/fork.c:1800 [inline]
       [<00000000c344af7c>] _do_fork+0x121/0x4f0 kernel/fork.c:2369

    Reported-by: syzbot+722da59ccb264bc19910@syzkaller.appspotmail.com
    Fixes: 719c7d563c17 ("ipvs: Fix use-after-free in ip_vs_in")
    Signed-off-by: Julian Anastasov <ja@ssi.bg>
    Acked-by: Simon Horman <horms@verge.net.au>
    Signed-off-by: Pablo Neira Ayuso <pablo@netfilter.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 556ae9c9a3ef9d7e4a0bdb4991f43c60c5b2492c
Author: Colin Ian King <colin.king@canonical.com>
Date:   Sat Jun 8 07:27:25 2019 -0400

    media: staging: davinci: fix memory leaks and check for allocation failure

    [ Upstream commit a84e355ecd3ed9759d7aaa40170aab78e2a68a06 ]

    There are three error return paths that don't kfree params causing a
    memory leak.  Fix this by adding an error return path that kfree's
    params before returning.  Also add a check to see params failed to
    be allocated.

    Signed-off-by: Colin Ian King <colin.king@canonical.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 813c009e9b2920f57fcfaa42d8c76f1152bebdd0
Author: Arnd Bergmann <arnd@arndb.de>
Date:   Tue Jun 18 13:22:13 2019 +0200

    ipsec: select crypto ciphers for xfrm_algo
```

```
    [ Upstream commit 597179b0ba550bd83fab1a9d57c42a9343c58514 ]

    kernelci.org reports failed builds on arc because of what looks
    like an old missed 'select' statement:

    net/xfrm/xfrm_algo.o: In function `xfrm_probe_algs':
    xfrm_algo.c:(.text+0x1e8): undefined reference to `crypto_has_ahash'

    I don't see this in randconfig builds on other architectures, but
    it's fairly clear we want to select the hash code for it, like we
    do for all its other users. As Herbert points out, CRYPTO_BLKCIPHER
    is also required even though it has not popped up in build tests.

    Fixes: 17bc19702221 ("ipsec: Use skcipher and ahash when probing algorithms")
    Signed-off-by: Arnd Bergmann <arnd@arndb.de>
    Acked-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Steffen Klassert <steffen.klassert@secunet.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b7a9b93f91aaf0436d06618b3db9c985899c9afe
Author: Julien Thierry <julien.thierry@arm.com>
Date:   Tue Jun 11 10:38:06 2019 +0100

    arm64: Do not enable IRQs for ct_user_exit

    [ Upstream commit 9034f6251572a4744597c51dea5ab73a55f2b938 ]

    For el0_dbg and el0_error, DAIF bits get explicitly cleared before
    calling ct_user_exit.

    When context tracking is disabled, DAIF gets set (almost) immediately
    after. When context tracking is enabled, among the first things done
    is disabling IRQs.

    What is actually needed is:
    - PSR.D = 0 so the system can be debugged (should be already the case)
    - PSR.A = 0 so async error can be handled during context tracking

    Do not clear PSR.I in those two locations.

    Reviewed-by: Marc Zyngier <marc.zyngier@arm.com>
    Acked-by: Mark Rutland <mark.rutland@arm.com>
    Reviewed-by: James Morse <james.morse@arm.com>
    Cc: Will Deacon <will.deacon@arm.com>
    Signed-off-by: Julien Thierry <julien.thierry@arm.com>
    Signed-off-by: Catalin Marinas <catalin.marinas@arm.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8f0a41cd10af64a199f59e48ffe95a687b2d286a
Author: Minwoo Im <minwoo.im.dev@gmail.com>
Date:   Sun Jun 9 03:02:19 2019 +0900

    nvme-pci: adjust irq max_vector using num_possible_cpus()

    [ Upstream commit dad77d63903e91a2e97a0c984cabe5d36e91ba60 ]

    If the "irq_queues" are greater than num_possible_cpus(),
    nvme_calc_irq_sets() can have irq set_size for HCTX_TYPE_DEFAULT greater
    than it can be afforded.
    2039         affd->set_size[HCTX_TYPE_DEFAULT] = nrirqs - nr_read_queues;

    It might cause a WARN() from the irq_build_affinity_masks() like [1]:
    220         if (nr_present < numvecs)
    221                 WARN_ON(nr_present + nr_others < numvecs);
```

```
    This patch prevents it from the WARN() by adjusting the max_vector value
    from the nvme_setup_irqs().

    [1] WARN messages when modprobe nvme write_queues=32 poll_queues=0:
    root@target:~/nvme# nproc
    8
    root@target:~/nvme# modprobe nvme write_queues=32 poll_queues=0
    [   17.925326] nvme nvme0: pci function 0000:00:04.0
    [   17.940601] WARNING: CPU: 3 PID: 1030 at kernel/irq/affinity.c:221
irq_create_affinity_masks+0x222/0x330
    [   17.940602] Modules linked in: nvme nvme_core [last unloaded: nvme]
    [   17.940605] CPU: 3 PID: 1030 Comm: kworker/u17:4 Tainted: G        W         5.1.0+
#156
    [   17.940605] Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS rel-1.12.1-0-
ga5cab58e9a3f-prebuilt.qemu.org 04/01/2014
    [   17.940608] Workqueue: nvme-reset-wq nvme_reset_work [nvme]
    [   17.940609] RIP: 0010:irq_create_affinity_masks+0x222/0x330
    [   17.940611] Code: 4c 8d 4c 24 28 4c 8d 44 24 30 e8 c9 fa ff ff 89 44 24 18 e8 c0 38
fa ff 8b 44 24 18 44 8b 54 24 1c 5a 44 01 d0 41 39 c4 76 02 <0f> 0b 48 89 df 44 01 e5 e8 f1
ce 10 00 48 8b 34 24 44 89 f0 44 01
    [   17.940611] RSP: 0018:ffffc90002277c50 EFLAGS: 00010216
    [   17.940612] RAX: 0000000000000008 RBX: ffff88807ca48860 RCX: 0000000000000000
    [   17.940612] RDX: ffff88807bc03800 RSI: 0000000000000020 RDI: 0000000000000000
    [   17.940613] RBP: 0000000000000001 R08: ffffc90002277c78 R09: ffffc90002277c70
    [   17.940613] R10: 0000000000000008 R11: 0000000000000001 R12: 0000000000000020
    [   17.940614] R13: 0000000000025d08 R14: 0000000000000001 R15: ffff88807bc03800
    [   17.940614] FS:  0000000000000000(0000) GS:ffff88807db80000(0000)
knlGS:0000000000000000
    [   17.940616] CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    [   17.940617] CR2: 00005635e583f790 CR3: 000000000240a000 CR4: 00000000000006e0
    [   17.940617] DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000
    [   17.940618] DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400
    [   17.940618] Call Trace:
    [   17.940622]  __pci_enable_msix_range+0x215/0x540
    [   17.940623]  ? kernfs_put+0x117/0x160
    [   17.940625]  pci_alloc_irq_vectors_affinity+0x74/0x110
    [   17.940626]  nvme_reset_work+0xc30/0x1397 [nvme]
    [   17.940628]  ? __switch_to_asm+0x34/0x70
    [   17.940628]  ? __switch_to_asm+0x40/0x70
    [   17.940629]  ? __switch_to_asm+0x34/0x70
    [   17.940630]  ? __switch_to_asm+0x40/0x70
    [   17.940630]  ? __switch_to_asm+0x34/0x70
    [   17.940631]  ? __switch_to_asm+0x40/0x70
    [   17.940632]  ? nvme_irq_check+0x30/0x30 [nvme]
    [   17.940633]  process_one_work+0x20b/0x3e0
    [   17.940634]  worker_thread+0x1f9/0x3d0
    [   17.940635]  ? cancel_delayed_work+0xa0/0xa0
    [   17.940636]  kthread+0x117/0x120
    [   17.940637]  ? kthread_stop+0xf0/0xf0
    [   17.940638]  ret_from_fork+0x3a/0x50
    [   17.940639] ---[ end trace aca8a131361cd42a ]---
    [   17.942124] nvme nvme0: 7/1/0 default/read/poll queues

    Signed-off-by: Minwoo Im <minwoo.im.dev@gmail.com>
    Signed-off-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cf1d95daf12cb83eaf57044e55a5ab8d82da9535
Author: Geert Uytterhoeven <geert@linux-m68k.org>
Date:   Fri Jun 21 11:12:00 2019 +0200

    lightnvm: fix uninitialized pointer in nvm_remove_tgt()

    [ Upstream commit 2f5af4ab7de14bd35f3435e6a47300276bbb6c17 ]
```

```
    With gcc 4.1:

        drivers/lightnvm/core.c: In function ‘nvm_remove_tgt’:
        drivers/lightnvm/core.c:510: warning: ‘t’ is used uninitialized in this function

    Indeed, if no NVM devices have been registered, t will be an
    uninitialized pointer, and may be dereferenced later.  A call to
    nvm_remove_tgt() can be triggered from userspace by issuing the
    NVM_DEV_REMOVE ioctl on the lightnvm control device.

    Fix this by preinitializing t to NULL.

    Fixes: 843f2edbdde085b4 ("lightnvm: do not remove instance under global lock")
    Signed-off-by: Geert Uytterhoeven <geert@linux-m68k.org>
    Signed-off-by: Matias Bjørling <mb@lightnvm.io>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c8a410cd3cfba1aabae6bfd4a38d395c4ebd9df6
Author: Heiner Litz <hlitz@ucsc.edu>
Date:   Fri Jun 21 11:11:59 2019 +0200

    lightnvm: pblk: fix freeing of merged pages

    [ Upstream commit 510fd8ea98fcb586c01aef93d87c060a159ac30a ]

    bio_add_pc_page() may merge pages when a bio is padded due to a flush.
    Fix iteration over the bio to free the correct pages in case of a merge.

    Signed-off-by: Heiner Litz <hlitz@ucsc.edu>
    Reviewed-by: Javier González <javier@javigon.com>
    Signed-off-by: Matias Bjørling <mb@lightnvm.io>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ad99783841ff35c728c745e645c309a95959a09f
Author: Chaitanya Kulkarni <chaitanya.kulkarni@wdc.com>
Date:   Sat Jun 8 13:01:02 2019 -0700

    nvme-pci: set the errno on ctrl state change error

    [ Upstream commit e71afda49335620e3d9adf56015676db33a3bd86 ]

    This patch removes the confusing assignment of the variable result at
    the time of declaration and sets the value in error cases next to the
    places where the actual error is happening.

    Here we also set the result value to -ENODEV when we fail at the final
    ctrl state transition in nvme_reset_work(). Without this assignment
    result will hold 0 from nvme_setup_io_queue() and on failure 0 will be
    passed to he nvme_remove_dead_ctrl() from final state transition.

    Signed-off-by: Chaitanya Kulkarni <chaitanya.kulkarni@wdc.com>
    Signed-off-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7cb611ce5b09485acb3bc8323d811539ff037e90
Author: Minwoo Im <minwoo.im.dev@gmail.com>
Date:   Sun Jun 9 03:35:20 2019 +0900

    nvme-pci: properly report state change failure in nvme_reset_work

    [ Upstream commit cee6c269b016ba89c62e34d6bccb103ee2c7de4f ]

    If the state change to NVME_CTRL_CONNECTING fails, the dmesg is going to
```

```
    be like:

      [  293.689160] nvme nvme0: failed to mark controller CONNECTING
      [  293.689160] nvme nvme0: Removing after probe failure status: 0

    Even it prints the first line to indicate the situation, the second line
    is not proper because the status is 0 which means normally success of
    the previous operation.

    This patch makes it indicate the proper error value when it fails.
      [   25.932367] nvme nvme0: failed to mark controller CONNECTING
      [   25.932369] nvme nvme0: Removing after probe failure status: -16

    This situation is able to be easily reproduced by:
      root@target:~# rmmod nvme && modprobe nvme && rmmod nvme

    Signed-off-by: Minwoo Im <minwoo.im.dev@gmail.com>
    Reviewed-by: Chaitanya Kulkarni <chaitanya.kulkarni@wdc.com>
    Signed-off-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6a64b52a5fb316aa76ef62ee2945aa3ce757eeeb
Author: Anton Eidelman <anton@lightbitslabs.com>
Date:   Thu Jun 20 08:48:10 2019 +0200

    nvme: fix possible io failures when removing multipathed ns

    [ Upstream commit 2181e455612a8db2761eabbf126640552a451e96 ]

    When a shared namespace is removed, we call blk_cleanup_queue()
    when the device can still be accessed as the current path and this can
    result in submission to a dying queue. Hence, direct_make_request()
    called by our mpath device may fail (propagating the failure to userspace).
    Instead, we want to failover this I/O to a different path if one exists.
    Thus, before we cleanup the request queue, we make sure that the device is
    cleared from the current path nor it can be selected again as such.

    Fix this by:
    - clear the ns from the head->list and synchronize rcu to make sure there is
      no concurrent path search that restores it as the current path
    - clear the mpath current path in order to trigger a subsequent path search
      and sync srcu to wait for any ongoing request submissions
    - safely continue to namespace removal and blk_cleanup_queue

    Signed-off-by: Anton Eidelman <anton@lightbitslabs.com>
    Signed-off-by: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4e0fff95afa4e4d9d87092fee0d139fd8b8e31cc
Author: Pan Bian <bianpan2016@163.com>
Date:   Thu Apr 18 10:27:18 2019 +0800

    EDAC/sysfs: Fix memory leak when creating a csrow object

    [ Upstream commit 585fb3d93d32dbe89e718b85009f9c322cc554cd ]

    In edac_create_csrow_object(), the reference to the object is not
    released when adding the device to the device hierarchy fails
    (device_add()). This may result in a memory leak.

    Signed-off-by: Pan Bian <bianpan2016@163.com>
    Signed-off-by: Borislav Petkov <bp@suse.de>
    Reviewed-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Cc: James Morse <james.morse@arm.com>
```

```
    Cc: Mauro Carvalho Chehab <mchehab@kernel.org>
    Cc: linux-edac <linux-edac@vger.kernel.org>
    Link: https://lkml.kernel.org/r/1555554438-103953-1-git-send-email-bianpan2016@163.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e39f0772b94a9bd471561e50e2bed8350616387b
Author: Greg KH <gregkh@linuxfoundation.org>
Date:   Wed May 8 12:40:42 2019 +0200

    EDAC/sysfs: Drop device references properly

    [ Upstream commit 7adc05d2dc3af95e4e1534841d58f736262142cd ]

    Do put_device() if device_add() fails.

     [ bp: do device_del() for the successfully created devices in
       edac_create_csrow_objects(), on the unwind path. ]

    Signed-off-by: Greg KH <gregkh@linuxfoundation.org>
    Signed-off-by: Borislav Petkov <bp@suse.de>
    Link: https://lkml.kernel.org/r/20190427214925.GE16338@kroah.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f96a5d4b1b525541cc7ea116192d768cf62f8331
Author: Tudor Ambarus <tudor.ambarus@microchip.com>
Date:   Wed Jun 19 14:38:28 2019 +0000

    spi: fix ctrl->num_chipselect constraint

    [ Upstream commit f9481b08220d7dc1ff21e296a330ee8b721b44e4 ]

    at91sam9g25ek showed the following error at probe:
    atmel_spi f0000000.spi: Using dma0chan2 (tx) and dma0chan3 (rx)
    for DMA transfers
    atmel_spi: probe of f0000000.spi failed with error -22

    Commit 0a919ae49223 ("spi: Don't call spi_get_gpio_descs() before device name is set")
    moved the calling of spi_get_gpio_descs() after ctrl->dev is set,
    but didn't move the !ctrl->num_chipselect check. When there are
    chip selects in the device tree, the spi-atmel driver lets the
    SPI core discover them when registering the SPI master.
    The ctrl->num_chipselect is thus expected to be set by
    spi_get_gpio_descs().

    Move the !ctlr->num_chipselect after spi_get_gpio_descs() as it was
    before the aforementioned commit. While touching this block, get rid
    of the explicit comparison with 0 and update the commenting style.

    Fixes: 0a919ae49223 ("spi: Don't call spi_get_gpio_descs() before device name is set")
    Signed-off-by: Tudor Ambarus <tudor.ambarus@microchip.com>
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a257222d849bed1da57deab4cf394e2794d9d5db
Author: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
Date:   Mon Jun 17 13:31:45 2019 +0200

    ACPICA: Clear status of GPEs on first direct enable

    [ Upstream commit 44758bafa53602f2581a6857bb20b55d4d8ad5b2 ]

    ACPI GPEs (other than the EC one) can be enabled in two situations.
    First, the GPEs with existing _Lxx and _Exx methods are enabled
    implicitly by ACPICA during system initialization.  Second, the
    GPEs without these methods (like GPEs listed by _PRW objects for
```

```
    wakeup devices) need to be enabled directly by the code that is
    going to use them (e.g. ACPI power management or device drivers).

    In the former case, if the status of a given GPE is set to start
    with, its handler method (either _Lxx or _Exx) needs to be invoked
    to take care of the events (possibly) signaled before the GPE was
    enabled.  In the latter case, however, the first caller of
    acpi_enable_gpe() for a given GPE should not be expected to care
    about any events that might be signaled through it earlier.  In
    that case, it is better to clear the status of the GPE before
    enabling it, to prevent stale events from triggering unwanted
    actions (like spurious system resume, for example).

    For this reason, modify acpi_ev_add_gpe_reference() to take an
    additional boolean argument indicating whether or not the GPE
    status needs to be cleared when its reference counter changes from
    zero to one and make acpi_enable_gpe() pass TRUE to it through
    that new argument.

    Fixes: 18996f2db918 ("ACPICA: Events: Stop unconditionally clearing ACPI IRQs during
suspend/resume")
    Reported-by: Furquan Shaikh <furquan@google.com>
    Tested-by: Furquan Shaikh <furquan@google.com>
    Tested-by: Mika Westerberg <mika.westerberg@linux.intel.com>
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ffebee147bbea7dd627b8aba8da2ecb72dd34288
Author: Dennis Zhou <dennis@kernel.org>
Date:   Thu May 23 16:10:18 2019 -0400

    blk-iolatency: only account submitted bios

    [ Upstream commit a3fb01ba5af066521f3f3421839e501bb2c71805 ]

    As is, iolatency recognizes done_bio and cleanup as ending paths. If a
    request is marked REQ_NOWAIT and fails to get a request, the bio is
    cleaned up via rq_qos_cleanup() and ended in bio_wouldblock_error().
    This results in underflowing the inflight counter. Fix this by only
    accounting bios that were actually submitted.

    Signed-off-by: Dennis Zhou <dennis@kernel.org>
    Cc: Josef Bacik <josef@toxicpanda.com>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3bd2baa72c12822efe9fc3d20f265ffcb1cd14c2
Author: Qian Cai <cai@lca.pw>
Date:   Wed Jun 19 10:32:53 2019 -0400

    x86/cacheinfo: Fix a -Wtype-limits warning

    [ Upstream commit 1b7aebf0487613033aff26420e32fa2076d52846 ]

    cpuinfo_x86.x86_model is an unsigned type, so comparing against zero
    will generate a compilation warning:

      arch/x86/kernel/cpu/cacheinfo.c: In function 'cacheinfo_amd_init_llc_id':
      arch/x86/kernel/cpu/cacheinfo.c:662:19: warning: comparison is always true \
        due to limited range of data type [-Wtype-limits]

    Remove the unnecessary lower bound check.

     [ bp: Massage. ]
```

```
      Fixes: 68091ee7ac3c ("x86/CPU/AMD: Calculate last level cache ID from number of sharing
  threads")
      Signed-off-by: Qian Cai <cai@lca.pw>
      Signed-off-by: Borislav Petkov <bp@suse.de>
      Reviewed-by: Sean Christopherson <sean.j.christopherson@intel.com>
      Cc: "Gustavo A. R. Silva" <gustavo@embeddedor.com>
      Cc: "H. Peter Anvin" <hpa@zytor.com>
      Cc: Ingo Molnar <mingo@redhat.com>
      Cc: Masami Hiramatsu <mhiramat@kernel.org>
      Cc: Pu Wen <puwen@hygon.cn>
      Cc: Suravee Suthikulpanit <suravee.suthikulpanit@amd.com>
      Cc: Thomas Gleixner <tglx@linutronix.de>
      Cc: x86-ml <x86@kernel.org>
      Link: https://lkml.kernel.org/r/1560954773-11967-1-git-send-email-cai@lca.pw
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit a294b5786e78fcf7409fddb0ccb6e647e178f4a3
  Author: Ilias Apalodimas <ilias.apalodimas@linaro.org>
  Date:   Wed Jun 19 13:04:00 2019 +0300

      net: netsec: initialize tx ring on ndo_open

      [ Upstream commit 39e3622edeffa63c2871153d8743c5825b139968 ]

      Since we changed the Tx ring handling and now depends on bit31 to figure
      out the owner of the descriptor, we should initialize this every time
      the device goes down-up instead of doing it once on driver init. If the
      value is not correctly initialized the device won't have any available
      descriptors

      Changes since v1:
      - Typo fixes

      Fixes: 35e07d234739 ("net: socionext: remove mmio reads on Tx")
      Signed-off-by: Ilias Apalodimas <ilias.apalodimas@linaro.org>
      Acked-by: Ard Biesheuvel <ard.biesheuvel@linaro.org>
      Signed-off-by: David S. Miller <davem@davemloft.net>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 5817d78eba34f6c86f5462ae2c5212f80a013357
  Author: Mika Westerberg <mika.westerberg@linux.intel.com>
  Date:   Wed Jun 12 13:57:38 2019 +0300

      PCI: Add missing link delays required by the PCIe spec

      [ Upstream commit c2bf1fc212f7e6f25ace1af8f0b3ac061ea48ba5 ]

      Currently Linux does not follow PCIe spec regarding the required delays
      after reset. A concrete example is a Thunderbolt add-in-card that
      consists of a PCIe switch and two PCIe endpoints:

        +-1b.0-[01-6b]----00.0-[02-6b]--+-00.0-[03]----00.0 TBT controller
                                        +-01.0-[04-36]-- DS hotplug port
                                        +-02.0-[37]----00.0 xHCI controller
                                        \-04.0-[38-6b]-- DS hotplug port

      The root port (1b.0) and the PCIe switch downstream ports are all PCIe
      gen3 so they support 8GT/s link speeds.

      We wait for the PCIe hierarchy to enter D3cold (runtime):

        pcieport 0000:00:1b.0: power state changed by ACPI to D3cold

      When it wakes up from D3cold, according to the PCIe 4.0 section 5.8 the
      PCIe switch is put to reset and its power is re-applied. This means that
```

```
    we must follow the rules in PCIe 4.0 section 6.6.1.

    For the PCIe gen3 ports we are dealing with here, the following applies:

      With a Downstream Port that supports Link speeds greater than 5.0
      GT/s, software must wait a minimum of 100 ms after Link training
      completes before sending a Configuration Request to the device
      immediately below that Port. Software can determine when Link training
      completes by polling the Data Link Layer Link Active bit or by setting
      up an associated interrupt (see Section 6.7.3.3).

    Translating this into the above topology we would need to do this (DLLLA
    stands for Data Link Layer Link Active):

      pcieport 0000:00:1b.0: wait for 100ms after DLLLA is set before access to 0000:01:00.0
      pcieport 0000:02:00.0: wait for 100ms after DLLLA is set before access to 0000:03:00.0
      pcieport 0000:02:02.0: wait for 100ms after DLLLA is set before access to 0000:37:00.0

    I've instrumented the kernel with additional logging so we can see the
    actual delays the kernel performs:

      pcieport 0000:00:1b.0: power state changed by ACPI to D0
      pcieport 0000:00:1b.0: waiting for D3cold delay of 100 ms
      pcieport 0000:00:1b.0: waking up bus
      pcieport 0000:00:1b.0: waiting for D3hot delay of 10 ms
      pcieport 0000:00:1b.0: restoring config space at offset 0x2c (was 0x60, writing 0x60)
      ...
      pcieport 0000:00:1b.0: PME# disabled
      pcieport 0000:01:00.0: restoring config space at offset 0x3c (was 0x1ff, writing
0x201ff)
      ...
      pcieport 0000:01:00.0: PME# disabled
      pcieport 0000:02:00.0: restoring config space at offset 0x3c (was 0x1ff, writing
0x201ff)
      ...
      pcieport 0000:02:00.0: PME# disabled
      pcieport 0000:02:01.0: restoring config space at offset 0x3c (was 0x1ff, writing
0x201ff)
      ...
      pcieport 0000:02:01.0: restoring config space at offset 0x4 (was 0x100000, writing
0x100407)
      pcieport 0000:02:01.0: PME# disabled
      pcieport 0000:02:02.0: restoring config space at offset 0x3c (was 0x1ff, writing
0x201ff)
      ...
      pcieport 0000:02:02.0: PME# disabled
      pcieport 0000:02:04.0: restoring config space at offset 0x3c (was 0x1ff, writing
0x201ff)
      ...
      pcieport 0000:02:04.0: PME# disabled
      pcieport 0000:02:01.0: PME# enabled
      pcieport 0000:02:01.0: waiting for D3hot delay of 10 ms
      pcieport 0000:02:04.0: PME# enabled
      pcieport 0000:02:04.0: waiting for D3hot delay of 10 ms
      thunderbolt 0000:03:00.0: restoring config space at offset 0x14 (was 0x0, writing
0x8a040000)
      ...
      thunderbolt 0000:03:00.0: PME# disabled
      xhci_hcd 0000:37:00.0: restoring config space at offset 0x10 (was 0x0, writing
0x73f00000)
      ...
      xhci_hcd 0000:37:00.0: PME# disabled

    For the switch upstream port (01:00.0) we wait for 100ms but not taking
    into account the DLLLA requirement. We then wait 10ms for D3hot -> D0
```

```
    transition of the root port and the two downstream hotplug ports. This
    means that we deviate from what the spec requires.

    Performing the same check for system sleep (s2idle) transitions we can
    see following when resuming from s2idle:

      pcieport 0000:00:1b.0: power state changed by ACPI to D0
      pcieport 0000:00:1b.0: restoring config space at offset 0x2c (was 0x60, writing 0x60)
      ...
      pcieport 0000:01:00.0: restoring config space at offset 0x3c (was 0x1ff, writing
0x201ff)
      ...
      pcieport 0000:02:02.0: restoring config space at offset 0x3c (was 0x1ff, writing
0x201ff)
      pcieport 0000:02:02.0: restoring config space at offset 0x2c (was 0x0, writing 0x0)
      pcieport 0000:02:01.0: restoring config space at offset 0x3c (was 0x1ff, writing
0x201ff)
      pcieport 0000:02:04.0: restoring config space at offset 0x3c (was 0x1ff, writing
0x201ff)
      pcieport 0000:02:02.0: restoring config space at offset 0x28 (was 0x0, writing 0x0)
      pcieport 0000:02:00.0: restoring config space at offset 0x3c (was 0x1ff, writing
0x201ff)
      pcieport 0000:02:02.0: restoring config space at offset 0x24 (was 0x10001, writing
0x1fff1)
      pcieport 0000:02:01.0: restoring config space at offset 0x2c (was 0x0, writing 0x60)
      pcieport 0000:02:02.0: restoring config space at offset 0x20 (was 0x0, writing
0x73f073f0)
      pcieport 0000:02:04.0: restoring config space at offset 0x2c (was 0x0, writing 0x60)
      pcieport 0000:02:01.0: restoring config space at offset 0x28 (was 0x0, writing 0x60)
      pcieport 0000:02:00.0: restoring config space at offset 0x2c (was 0x0, writing 0x0)
      pcieport 0000:02:02.0: restoring config space at offset 0x1c (was 0x101, writing
0x1f1)
      pcieport 0000:02:04.0: restoring config space at offset 0x28 (was 0x0, writing 0x60)
      pcieport 0000:02:01.0: restoring config space at offset 0x24 (was 0x10001, writing
0x1ff10001)
      pcieport 0000:02:00.0: restoring config space at offset 0x28 (was 0x0, writing 0x0)
      pcieport 0000:02:02.0: restoring config space at offset 0x18 (was 0x0, writing
0x373702)
      pcieport 0000:02:04.0: restoring config space at offset 0x24 (was 0x10001, writing
0x49f12001)
      pcieport 0000:02:01.0: restoring config space at offset 0x20 (was 0x0, writing
0x73e05c00)
      pcieport 0000:02:00.0: restoring config space at offset 0x24 (was 0x10001, writing
0x1fff1)
      pcieport 0000:02:04.0: restoring config space at offset 0x20 (was 0x0, writing
0x89f07400)
      pcieport 0000:02:01.0: restoring config space at offset 0x1c (was 0x101, writing
0x5151)
      pcieport 0000:02:00.0: restoring config space at offset 0x20 (was 0x0, writing
0x8a008a00)
      pcieport 0000:02:02.0: restoring config space at offset 0xc (was 0x10000, writing
0x10020)
      pcieport 0000:02:04.0: restoring config space at offset 0x1c (was 0x101, writing
0x6161)
      pcieport 0000:02:01.0: restoring config space at offset 0x18 (was 0x0, writing
0x360402)
      pcieport 0000:02:00.0: restoring config space at offset 0x1c (was 0x101, writing
0x1f1)
      pcieport 0000:02:04.0: restoring config space at offset 0x18 (was 0x0, writing
0x6b3802)
      pcieport 0000:02:02.0: restoring config space at offset 0x4 (was 0x100000, writing
0x100407)
      pcieport 0000:02:00.0: restoring config space at offset 0x18 (was 0x0, writing
0x30302)
      pcieport 0000:02:01.0: restoring config space at offset 0xc (was 0x10000, writing
```

```
0x10020)
      pcieport 0000:02:04.0: restoring config space at offset 0xc (was 0x10000, writing
0x10020)
      pcieport 0000:02:00.0: restoring config space at offset 0xc (was 0x10000, writing
0x10020)
      pcieport 0000:02:01.0: restoring config space at offset 0x4 (was 0x100000, writing
0x100407)
      pcieport 0000:02:04.0: restoring config space at offset 0x4 (was 0x100000, writing
0x100407)
      pcieport 0000:02:00.0: restoring config space at offset 0x4 (was 0x100000, writing
0x100407)
      xhci_hcd 0000:37:00.0: restoring config space at offset 0x10 (was 0x0, writing
0x73f00000)
      ...
      thunderbolt 0000:03:00.0: restoring config space at offset 0x14 (was 0x0, writing
0x8a040000)

    This is even worse. None of the mandatory delays are performed. If this
    would be S3 instead of s2idle then according to PCI FW spec 3.2 section
    4.6.8.  there is a specific _DSM that allows the OS to skip the delays
    but this platform does not provide the _DSM and does not go to S3 anyway
    so no firmware is involved that could already handle these delays.

    In this particular Intel Coffee Lake platform these delays are not
    actually needed because there is an additional delay as part of the ACPI
    power resource that is used to turn on power to the hierarchy but since
    that additional delay is not required by any of standards (PCIe, ACPI)
    it is not present in the Intel Ice Lake, for example where missing the
    mandatory delays causes pciehp to start tearing down the stack too early
    (links are not yet trained).

    For this reason, change the PCIe portdrv PM resume hooks so that they
    perform the mandatory delays before the downstream component gets
    resumed. We perform the delays before port services are resumed because
    otherwise pciehp might find that the link is not up (even if it is just
    training) and tears-down the hierarchy.

    Signed-off-by: Mika Westerberg <mika.westerberg@linux.intel.com>
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b1d218908e4670c300bc22a0840072f0401af5f3
Author: Arnaldo Carvalho de Melo <acme@redhat.com>
Date:   Tue Jun 18 17:48:12 2019 -0300

    perf build: Handle slang being in /usr/include and in /usr/include/slang/

    [ Upstream commit 78d6ccce03e86de34c7000bcada493ed0679e350 ]

    In some distros slang.h may be in a /usr/include 'slang' subdir, so use
    the if slang is not explicitely disabled (by using NO_SLANG=1) and its
    feature test for the common case (having /usr/include/slang.h) failed,
    use the results for the test that checks if it is in slang/slang.h.

    Change the only file in perf that includes slang.h to use
    HAVE_SLANG_INCLUDE_SUBDIR and forget about this for good.

    On a rhel6 system now we have:

      $ /tmp/build/perf/perf -vv | grep slang
                    libslang: [ on  ]  # HAVE_SLANG_SUPPORT
      $ ldd /tmp/build/perf/perf | grep libslang
            libslang.so.2 => /usr/lib64/libslang.so.2 (0x00007fa2d5a8d000)
      $ grep slang /tmp/build/perf/FEATURE-DUMP
      feature-libslang=0
```

```
       feature-libslang-include-subdir=1
       $ cat /etc/redhat-release
       CentOS release 6.10 (Final)
       $

     While on fedora:29:

       $ /tmp/build/perf/perf -vv | grep slang
                     libslang: [ on  ]  # HAVE_SLANG_SUPPORT
       $ ldd /tmp/build/perf/perf | grep slang
             libslang.so.2 => /lib64/libslang.so.2 (0x00007f8eb11a7000)
       $ grep slang /tmp/build/perf/FEATURE-DUMP
       feature-libslang=1
       feature-libslang-include-subdir=1
       $
       $ cat /etc/fedora-release
       Fedora release 29 (Twenty Nine)
       $

     The feature-libslang-include-subdir=1 line is because the 'gettid()'
     test was added to test-all.c as the new glibc has an implementation for
     that, so we soon should have it not failing, i.e. should be the common
     case soon. Perhaps I should move it out till it becomes the norm...

     Cc: Adrian Hunter <adrian.hunter@intel.com>
     Cc: Florian Fainelli <f.fainelli@gmail.com>
     Cc: Jiri Olsa <jolsa@kernel.org>
     Cc: Namhyung Kim <namhyung@kernel.org>
     Fixes: 1955c8cf5e26 ("perf tools: Don't hardcode host include path for libslang")
     Link: https://lkml.kernel.org/n/tip-bkgtpsu3uit821fuwsdhj9gd@git.kernel.org
     Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit c407f8770e9dc8b852893d0a74ecaa7b88a122ac
  Author: Alexei Starovoitov <ast@kernel.org>
  Date:   Sat Jun 15 12:12:21 2019 -0700

     bpf: fix callees pruning callers

     [ Upstream commit eea1c227b9e9bad295e8ef984004a9acf12bb68c ]

     The commit 7640ead93924 partially resolved the issue of callees
     incorrectly pruning the callers.
     With introduction of bounded loops and jmps_processed heuristic
     single verifier state may contain multiple branches and calls.
     It's possible that new verifier state (for future pruning) will be
     allocated inside callee. Then callee will exit (still within the same
     verifier state). It will go back to the caller and there R6-R9 registers
     will be read and will trigger mark_reg_read. But the reg->live for all frames
     but the top frame is not set to LIVE_NONE. Hence mark_reg_read will fail
     to propagate liveness into parent and future walking will incorrectly
     conclude that the states are equivalent because LIVE_READ is not set.
     In other words the rule for parent/live should be:
     whenever register parentage chain is set the reg->live should be set to LIVE_NONE.
     is_state_visited logic already follows this rule for spilled registers.

     Fixes: 7640ead93924 ("bpf: verifier: make sure callees don't prune with caller
  differences")
     Fixes: f4d7e40a5b71 ("bpf: introduce function calls (verification)")
     Signed-off-by: Alexei Starovoitov <ast@kernel.org>
     Signed-off-by: Daniel Borkmann <daniel@iogearbox.net>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 3c0f17e28504bdcd243c656a09c674ebee4dfe9b
  Author: Arnaldo Carvalho de Melo <acme@redhat.com>
```

```
Date:   Tue Jun 18 17:59:16 2019 -0300

    tools build: Fix the zstd test in the test-all.c common case feature test

    [ Upstream commit 3469fa84c1631face938efc42b3f488a2c2504e0 ]

    We were renanimg 'main' to 'main_zstd' but then using 'main_libzstd();'
    in the main() for test-all.c, causing this:

      $ cat /tmp/build/perf/feature/test-all.make.output
      test-all.c: In function ‘main’:
      test-all.c:236:2: error: implicit declaration of function ‘main_test_libzstd’; did you
mean ‘main_test_zstd’? [-Werror=implicit-function-declaration]
        main_test_libzstd();
        ^~~~~~~~~~~~~~~~~
        main_test_zstd
      cc1: all warnings being treated as errors
      $

    I.e. what was supposed to be the fast path feature test was _always_
    failing, duh, fix it.

    Cc: Adrian Hunter <adrian.hunter@intel.com>
    Cc: Jiri Olsa <jolsa@kernel.org>
    Cc: Namhyung Kim <namhyung@kernel.org>
    Cc: Alexey Budankov <alexey.budankov@linux.intel.com>
    Fixes: 3b1c5d965971 ("tools build: Implement libzstd feature check, LIBZSTD_DIR and
NO_LIBZSTD defines")
    Link: https://lkml.kernel.org/n/tip-ma4abk0utroiw4mwpmvnjlru@git.kernel.org
    Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 38bded12152dfd85e98528940bf3e1f46544bf89
Author: Nilkanth Ahirrao <anilkanth@jp.adit-jv.com>
Date:   Tue Jun 18 14:19:53 2019 +0900

    ASoC: rsnd: fixup mod ID calculation in rsnd_ctu_probe_

    [ Upstream commit ac28ec07ae1c5c1e18ed6855eb105a328418da88 ]

    commit c16015f36cc1 ("ASoC: rsnd: add .get_id/.get_id_sub")
    introduces rsnd_ctu_id which calcualates and gives
    the main Device id of the CTU by dividing the id by 4.
    rsnd_mod_id uses this interface to get the CTU main
    Device id. But this commit forgets to revert the main
    Device id calcution previously done in rsnd_ctu_probe_
    which also divides the id by 4. This path corrects the
    same to get the correct main Device id.

    The issue is observered when rsnd_ctu_probe_ is done for CTU1

    Fixes: c16015f36cc1 ("ASoC: rsnd: add .get_id/.get_id_sub")

    Signed-off-by: Nilkanth Ahirrao <anilkanth@jp.adit-jv.com>
    Signed-off-by: Suresh Udipi <sudipi@jp.adit-jv.com>
    Signed-off-by: Jiada Wang <jiada_wang@mentor.com>
    Acked-by: Kuninori Morimoto <kuninori.morimoto.gx@renesas.com>
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 255333f141535e728b9c213dc8859798a7615c22
Author: Denis Kirjanov <kda@linux-powerpc.org>
Date:   Mon Jun 17 10:53:40 2019 +0200

    ipoib: correcly show a VF hardware address
```

```
    [ Upstream commit 64d701c608fea362881e823b666327f5d28d7ffd ]

    in the case of IPoIB with SRIOV enabled hardware
    ip link show command incorrecly prints
    0 instead of a VF hardware address.

    Before:
    11: ib1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 2044 qdisc pfifo_fast
    state UP mode DEFAULT group default qlen 256
        link/infiniband
    80:00:00:66:fe:80:00:00:00:00:00:00:00:24:8a:07:03:00:a4:3e:7c brd
    00:ff:ff:ff:ff:12:40:1b:ff:ff:00:00:00:00:00:00:ff:ff:ff:ff
        vf 0 MAC 00:00:00:00:00:00, spoof checking off, link-state disable,
    trust off, query_rss off
    ...
    After:
    11: ib1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 2044 qdisc pfifo_fast
    state UP mode DEFAULT group default qlen 256
        link/infiniband
    80:00:00:66:fe:80:00:00:00:00:00:00:00:24:8a:07:03:00:a4:3e:7c brd
    00:ff:ff:ff:ff:12:40:1b:ff:ff:00:00:00:00:00:00:ff:ff:ff:ff
        vf 0     link/infiniband
    80:00:00:66:fe:80:00:00:00:00:00:00:00:24:8a:07:03:00:a4:3e:7c brd
    00:ff:ff:ff:ff:12:40:1b:ff:ff:00:00:00:00:00:00:ff:ff:ff:ff, spoof
    checking off, link-state disable, trust off, query_rss off

    v1->v2: just copy an address without modifing ifla_vf_mac
    v2->v3: update the changelog

    Signed-off-by: Denis Kirjanov <kda@linux-powerpc.org>
    Acked-by: Doug Ledford <dledford@redhat.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 833577ecf3451c7306abb48f221d365c2ee4cc1b
Author: Mitch Williams <mitch.a.williams@intel.com>
Date:   Tue May 14 10:37:09 2019 -0700

    iavf: allow null RX descriptors

    [ Upstream commit efa14c3985828da3163f5372137cb64d992b0f79 ]

    In some circumstances, the hardware can hand us a null receive
    descriptor, with no data attached but otherwise valid. Unfortunately,
    the driver was ill-equipped to handle such an event, and would stop
    processing packets at that point.

    To fix this, use the Descriptor Done bit instead of the size to
    determine whether or not a descriptor is ready to be processed. Add some
    checks to allow for unused buffers.

    Signed-off-by: Mitch Williams <mitch.a.williams@intel.com>
    Tested-by: Andrew Bowers <andrewx.bowers@intel.com>
    Signed-off-by: Jeff Kirsher <jeffrey.t.kirsher@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9d28094f3cf3b315e784f04ba617de8c8d8978fa
Author: Jason Wang <jasowang@redhat.com>
Date:   Mon Jun 17 05:20:54 2019 -0400

    vhost_net: disable zerocopy by default

    [ Upstream commit 098eadce3c622c07b328d0a43dda379b38cf7c5e ]
```

```
    Vhost_net was known to suffer from HOL[1] issues which is not easy to
    fix. Several downstream disable the feature by default. What's more,
    the datapath was split and datacopy path got the support of batching
    and XDP support recently which makes it faster than zerocopy part for
    small packets transmission.

    It looks to me that disable zerocopy by default is more
    appropriate. It cold be enabled by default again in the future if we
    fix the above issues.

    [1] https://patchwork.kernel.org/patch/3787671/

    Signed-off-by: Jason Wang <jasowang@redhat.com>
    Acked-by: Michael S. Tsirkin <mst@redhat.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9c88708a3ba987d03250f20a04599979b15f96a4
Author: Arnaldo Carvalho de Melo <acme@redhat.com>
Date:   Mon Jun 17 14:32:53 2019 -0300

    perf evsel: Make perf_evsel__name() accept a NULL argument

    [ Upstream commit fdbdd7e8580eac9bdafa532746c865644d125e34 ]

    In which case it simply returns "unknown", like when it can't figure out
    the evsel->name value.

    This makes this code more robust and fixes a problem in 'perf trace'
    where a NULL evsel was being passed to a routine that only used the
    evsel for printing its name when a invalid syscall id was passed.

    Reported-by: Leo Yan <leo.yan@linaro.org>
    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-f30ztaasku3z935cn3ak3h53@git.kernel.org
    Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2175aa3f1a89cb1bb15b23d31898ff359f62952e
Author: Peter Zijlstra <peterz@infradead.org>
Date:   Wed Apr 24 13:38:23 2019 +0200

    x86/atomic: Fix smp_mb__{before,after}_atomic()

    [ Upstream commit 69d927bba39517d0980462efc051875b7f4db185 ]

    Recent probing at the Linux Kernel Memory Model uncovered a
    'surprise'. Strongly ordered architectures where the atomic RmW
    primitive implies full memory ordering and
    smp_mb__{before,after}_atomic() are a simple barrier() (such as x86)
    fail for:

            *x = 1;
            atomic_inc(u);
            smp_mb__after_atomic();
            r0 = *y;

    Because, while the atomic_inc() implies memory order, it
    (surprisingly) does not provide a compiler barrier. This then allows
    the compiler to re-order like so:

            atomic_inc(u);
            *x = 1;
```

```
             smp_mb__after_atomic();
             r0 = *y;

     Which the CPU is then allowed to re-order (under TSO rules) like:

             atomic_inc(u);
             r0 = *y;
             *x = 1;

     And this very much was not intended. Therefore strengthen the atomic
     RmW ops to include a compiler barrier.

     NOTE: atomic_{or,and,xor} and the bitops already had the compiler
     barrier.

     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>
     Signed-off-by: Ingo Molnar <mingo@kernel.org>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit 179cdf4ff36f2d2e02c1b5afd925133ec1767bb5
 Author: Geert Uytterhoeven <geert@linux-m68k.org>
 Date:   Mon Jun 17 09:44:52 2019 +0200

     integrity: Fix __integrity_init_keyring() section mismatch

     [ Upstream commit 8c655784e2cf59cb6140759b8b546d98261d1ad9 ]

     With gcc-4.6.3:

         WARNING: vmlinux.o(.text.unlikely+0x24c64): Section mismatch in reference from the
 function __integrity_init_keyring() to the function .init.text:set_platform_trusted_keys()
         The function __integrity_init_keyring() references
         the function __init set_platform_trusted_keys().
         This is often because __integrity_init_keyring lacks a __init
         annotation or the annotation of set_platform_trusted_keys is wrong.

     Indeed, if the compiler decides not to inline __integrity_init_keyring(),
     a warning is issued.

     Fix this by adding the missing __init annotation.

     Fixes: 9dc92c45177ab70e ("integrity: Define a trusted platform keyring")
     Signed-off-by: Geert Uytterhoeven <geert@linux-m68k.org>
     Reviewed-by: Nayna Jain <nayna@linux.ibm.com>
     Reviewed-by: James Morris <jamorris@linux.microsoft.com>
     Signed-off-by: Mimi Zohar <zohar@linux.ibm.com>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit 11113029ac7eb9f2f7d8da2726d30fb6b40c3074
 Author: Kan Liang <kan.liang@linux.intel.com>
 Date:   Tue Apr 30 17:53:43 2019 -0700

     perf/x86/intel/uncore: Handle invalid event coding for free-running counter

     [ Upstream commit 543ac280b3576c0009e8c0fcd4d6bfc9978d7bd0 ]

     Counting with invalid event coding for free-running counter may cause
     OOPs, e.g. uncore_iio_free_running_0/event=1/.

     Current code only validate the event with free-running event format,
     event=0xff,umask=0xXY. Non-free-running event format never be checked
     for the PMU with free-running counters.
```

```
    Add generic hw_config() to check and reject the invalid event coding
    for free-running PMU.

    Signed-off-by: Kan Liang <kan.liang@linux.intel.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: acme@kernel.org
    Cc: eranian@google.com
    Fixes: 0f519f0352e3 ("perf/x86/intel/uncore: Support IIO free-running counters on SKX")
    Link: https://lkml.kernel.org/r/1556672028-119221-2-git-send-email-
kan.liang@linux.intel.com
    Signed-off-by: Ingo Molnar <mingo@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit eb3a8802d38f3e559bbf54595cb63205520cb732
Author: Jiri Olsa <jolsa@redhat.com>
Date:   Sun Jun 16 16:13:13 2019 +0200

    perf/x86/intel: Disable check_msr for real HW

    [ Upstream commit d0e1a507bdc761a14906f03399d933ea639a1756 ]

    Tom Vaden reported false failure of the check_msr() function, because
    some servers can do POST tracing and enable LBR tracing during
    bootup.

    Kan confirmed that check_msr patch was to fix a bug report in
    guest, so it's ok to disable it for real HW.

    Reported-by: Tom Vaden <tom.vaden@hpe.com>
    Signed-off-by: Jiri Olsa <jolsa@kernel.org>
    Signed-off-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Acked-by: Tom Vaden <tom.vaden@hpe.com>
    Cc: Alexander Shishkin <alexander.shishkin@linux.intel.com>
    Cc: Arnaldo Carvalho de Melo <acme@kernel.org>
    Cc: Liang Kan <kan.liang@linux.intel.com>
    Cc: Linus Torvalds <torvalds@linux-foundation.org>
    Cc: Namhyung Kim <namhyung@kernel.org>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Cc: Thomas Gleixner <tglx@linutronix.de>
    Link: https://lkml.kernel.org/r/20190616141313.GD2500@krava
    [ Readability edits. ]
    Signed-off-by: Ingo Molnar <mingo@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 41a811e161217fbe3998d8a4f415b19b59280b90
Author: Qian Cai <cai@lca.pw>
Date:   Mon Jun 3 17:11:44 2019 -0400

    sched/fair: Fix "runnable_avg_yN_inv" not used warnings

    [ Upstream commit 509466b7d480bc5d22e90b9fbe6122ae0e2fbe39 ]

    runnable_avg_yN_inv[] is only used in kernel/sched/pelt.c but was
    included in several other places because they need other macros all
    came from kernel/sched/sched-pelt.h which was generated by
    Documentation/scheduler/sched-pelt. As the result, it causes compilation
    a lot of warnings,

      kernel/sched/sched-pelt.h:4:18: warning: 'runnable_avg_yN_inv' defined but not used [-
Wunused-const-variable=]
      kernel/sched/sched-pelt.h:4:18: warning: 'runnable_avg_yN_inv' defined but not used [-
```

```
  Wunused-const-variable=]
        kernel/sched/sched-pelt.h:4:18: warning: 'runnable_avg_yN_inv' defined but not used [-
  Wunused-const-variable=]
        ...

      Silence it by appending the __maybe_unused attribute for it, so all
      generated variables and macros can still be kept in the same file.

      Signed-off-by: Qian Cai <cai@lca.pw>
      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/1559596304-31581-1-git-send-email-cai@lca.pw
      Signed-off-by: Ingo Molnar <mingo@kernel.org>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 161c926ba6f0bb779c0fb860d3cf390eb314d345
  Author: Kan Liang <kan.liang@linux.intel.com>
  Date:   Mon Jun 3 06:41:21 2019 -0700

      perf/x86/intel: Add more Icelake CPUIDs

      [ Upstream commit faaeff98666c24376cebd0b106504d05a36881d1 ]

      Add new model number for Icelake desktop and server to perf.

      The data source encoding for Icelake server is the same as Skylake
      server.

      Signed-off-by: Kan Liang <kan.liang@linux.intel.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: bp@alien8.de
      Cc: qiuxu.zhuo@intel.com
      Cc: rui.zhang@intel.com
      Cc: tony.luck@intel.com
      Link: https://lkml.kernel.org/r/20190603134122.13853-2-kan.liang@linux.intel.com
      Signed-off-by: Ingo Molnar <mingo@kernel.org>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 29c36efe71e493c537f11ba86c95b1208891993b
  Author: Gao Xiang <gaoxiang25@huawei.com>
  Date:   Mon Jun 3 17:13:38 2019 +0800

      sched/core: Add __sched tag for io_schedule()

      [ Upstream commit e3b929b0a184edb35531153c5afcaebb09014f9d ]

      Non-inline io_schedule() was introduced in:

        commit 10ab56434f2f ("sched/core: Separate out io_schedule_prepare() and
  io_schedule_finish()")

      Keep in line with io_schedule_timeout(), otherwise "/proc/<pid>/wchan" will
      report io_schedule() rather than its callers when waiting for IO.

      Reported-by: Jilong Kou <koujilong@huawei.com>
      Signed-off-by: Gao Xiang <gaoxiang25@huawei.com>
      Signed-off-by: Peter Zijlstra (Intel) <peterz@infradead.org>
      Acked-by: Tejun Heo <tj@kernel.org>
      Cc: Andrew Morton <akpm@linux-foundation.org>
      Cc: Linus Torvalds <torvalds@linux-foundation.org>
```

```
    Cc: Miao Xie <miaoxie@huawei.com>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Cc: Thomas Gleixner <tglx@linutronix.de>
    Fixes: 10ab56434f2f ("sched/core: Separate out io_schedule_prepare() and
io_schedule_finish()")
    Link: https://lkml.kernel.org/r/20190603091338.2695-1-gaoxiang25@huawei.com
    Signed-off-by: Ingo Molnar <mingo@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cddcfccf8fd1b0608201d2265a585b736654a7c2
Author: Nicolas Dichtel <nicolas.dichtel@6wind.com>
Date:   Fri Jun 14 11:13:55 2019 +0200

    xfrm: fix sa selector validation

    [ Upstream commit b8d6d0079757cbd1b69724cfd1c08e2171c68cee ]

    After commit b38ff4075a80, the following command does not work anymore:
    $ ip xfrm state add src 10.125.0.2 dst 10.125.0.1 proto esp spi 34 reqid 1 \
      mode tunnel enc 'cbc(aes)' 0xb0abdba8b782ad9d364ec81e3a7d82a1 auth-trunc \
      'hmac(sha1)' 0xe26609ebd00acb6a4d51fca13e49ea78a72c73e6 96 flag align4

    In fact, the selector is not mandatory, allow the user to provide an empty
    selector.

    Fixes: b38ff4075a80 ("xfrm: Fix xfrm sel prefix length validation")
    CC: Anirudh Gupta <anirudh.gupta@sophos.com>
    Signed-off-by: Nicolas Dichtel <nicolas.dichtel@6wind.com>
    Acked-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Steffen Klassert <steffen.klassert@secunet.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f92c955af769a5fffbab3a55a277374920c0f7fc
Author: Tejun Heo <tj@kernel.org>
Date:   Thu Jun 13 15:30:41 2019 -0700

    blkcg, writeback: dead memcgs shouldn't contribute to writeback ownership arbitration

    [ Upstream commit 6631142229005e1b1c311a09efe9fb3cfdac8559 ]

    wbc_account_io() collects information on cgroup ownership of writeback
    pages to determine which cgroup should own the inode.  Pages can stay
    associated with dead memcgs but we want to avoid attributing IOs to
    dead blkcgs as much as possible as the association is likely to be
    stale.  However, currently, pages associated with dead memcgs
    contribute to the accounting delaying and/or confusing the
    arbitration.

    Fix it by ignoring pages associated with dead memcgs.

    Signed-off-by: Tejun Heo <tj@kernel.org>
    Cc: Jan Kara <jack@suse.cz>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c538dfa07348744b52a0a81190cce6e544a22a6c
Author: Bob Liu <bob.liu@oracle.com>
Date:   Sat Jun 15 01:43:48 2019 -0600

    block: null_blk: fix race condition for null_del_dev

    [ Upstream commit 7602843fd873cae43a444b83b14dfdd114a9659c ]

    Dulicate call of null_del_dev() will trigger null pointer error like below.
    The reason is a race condition between nullb_device_power_store() and
```

```
    nullb_group_drop_item().

      CPU#0                          CPU#1
      ----------------               -----------------
      do_rmdir()
       >configfs_rmdir()
        >client_drop_item()
         >nullb_group_drop_item()
                                     nullb_device_power_store()
                                     >null_del_dev()

          >test_and_clear_bit(NULLB_DEV_FL_UP
           >null_del_dev()
           ^^^^^
           Duplicated null_dev_dev() triger null pointer error

                                     >clear_bit(NULLB_DEV_FL_UP

    The fix could be keep the sequnce of clear NULLB_DEV_FL_UP and null_del_dev().

    [  698.613600] BUG: unable to handle kernel NULL pointer dereference at 0000000000000018
    [  698.613608] #PF error: [normal kernel read fault]
    [  698.613611] PGD 0 P4D 0
    [  698.613619] Oops: 0000 [#1] SMP PTI
    [  698.613627] CPU: 3 PID: 6382 Comm: rmdir Not tainted 5.0.0+ #35
    [  698.613631] Hardware name: LENOVO 20LJS2EV08/20LJS2EV08, BIOS R0SET33W (1.17 )
07/18/2018
    [  698.613644] RIP: 0010:null_del_dev+0xc/0x110 [null_blk]
    [  698.613649] Code: 00 00 00 5b 41 5c 41 5d 41 5e 41 5f 5d c3 0f 0b eb 97 e8 47 bb 2a
e8 0f 1f 80 00 00 00 00 0f 1f 44 00 00 55 48 89 e5 41 54 53 <8b> 77 18 48 89 fb 4c 8b 27 48
c7 c7 40 57 1e c1 e8 bf c7 cb e8 48
    [  698.613654] RSP: 0018:ffffb887888bfde0 EFLAGS: 00010286
    [  698.613659] RAX: 0000000000000000 RBX: ffff9d436d92bc00 RCX: ffff9d43a9184681
    [  698.613663] RDX: ffffffffc11e5c30 RSI: 0000000068be6540 RDI: 0000000000000000
    [  698.613667] RBP: ffffb887888bfdf0 R08: 0000000000000001 R09: 0000000000000000
    [  698.613671] R10: ffffb887888bfdd8 R11: 0000000000000f16 R12: ffff9d436d92bc08
    [  698.613675] R13: ffff9d436d94e630 R14: ffffffffc11e5088 R15: ffffffffc11e5000
    [  698.613680] FS:  00007faa68be6540(0000) GS:ffff9d43d14c0000(0000)
knlGS:0000000000000000
    [  698.613685] CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    [  698.613689] CR2: 0000000000000018 CR3: 000000042f70c002 CR4: 00000000003606e0
    [  698.613693] DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000
    [  698.613697] DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400
    [  698.613700] Call Trace:
    [  698.613712]  nullb_group_drop_item+0x50/0x70 [null_blk]
    [  698.613722]  client_drop_item+0x29/0x40
    [  698.613728]  configfs_rmdir+0x1ed/0x300
    [  698.613738]  vfs_rmdir+0xb2/0x130
    [  698.613743]  do_rmdir+0x1c7/0x1e0
    [  698.613750]  __x64_sys_rmdir+0x17/0x20
    [  698.613759]  do_syscall_64+0x5a/0x110
    [  698.613768]  entry_SYSCALL_64_after_hwframe+0x44/0xa9

    Signed-off-by: Bob Liu <bob.liu@oracle.com>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 64db92d1cca390e5564eca90cb45cda9626ce5a0
Author: Yunsheng Lin <linyunsheng@huawei.com>
Date:   Thu Jun 13 17:12:31 2019 +0800

    net: hns3: delay ring buffer clearing during reset

    [ Upstream commit 3a30964a2eef6aabd3ab18b979ea0eacf1147731 ]
```

```
    The driver may not be able to disable the ring through firmware
    when downing the netdev during reset process, which may cause
    hardware accessing freed buffer problem.

    This patch delays the ring buffer clearing to reset uninit
    process because hardware will not access the ring buffer after
    hardware reset is completed.

    Fixes: bb6b94a896d4 ("net: hns3: Add reset interface implementation in client")
    Signed-off-by: Yunsheng Lin <linyunsheng@huawei.com>
    Signed-off-by: Peng Li <lipeng321@huawei.com>
    Signed-off-by: Huazhong Tan <tanhuazhong@huawei.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 648e4b6b43e3a982634d6fc63542f380663424a5
Author: Yunsheng Lin <linyunsheng@huawei.com>
Date:   Thu Jun 13 17:12:30 2019 +0800

    net: hns3: fix for skb leak when doing selftest

    [ Upstream commit 8f9eed1a8791b83eb1c54c261d68424717e4111e ]

    If hns3_nic_net_xmit does not return NETDEV_TX_BUSY when doing
    a loopback selftest, the skb is not freed in hns3_clean_tx_ring
    or hns3_nic_net_xmit, which causes skb not freed problem.

    This patch fixes it by freeing skb when hns3_nic_net_xmit does
    not return NETDEV_TX_OK.

    Fixes: c39c4d98dc65 ("net: hns3: Add mac loopback selftest support in hns3 driver")

    Signed-off-by: Yunsheng Lin <linyunsheng@huawei.com>
    Signed-off-by: Peng Li <lipeng321@huawei.com>
    Signed-off-by: Huazhong Tan <tanhuazhong@huawei.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 03d5cc280629171d09573aebf0b8dd6a7d1c04af
Author: Yunsheng Lin <linyunsheng@huawei.com>
Date:   Thu Jun 13 17:12:29 2019 +0800

    net: hns3: fix for dereferencing before null checking

    [ Upstream commit 757188005f905664b0186b88cf26a7e844190a63 ]

    The netdev is dereferenced before null checking in the function
    hns3_setup_tc.

    This patch moves the dereferencing after the null checking.

    Fixes: 76ad4f0ee747 ("net: hns3: Add support of HNS3 Ethernet Driver for hip08 SoC")

    Signed-off-by: Yunsheng Lin <linyunsheng@huawei.com>
    Signed-off-by: Peng Li <lipeng321@huawei.com>
    Signed-off-by: Huazhong Tan <tanhuazhong@huawei.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e2a7aa231b0bb35f31381dceb32a78d9241a1e56
Author: Michal Kalderon <michal.kalderon@marvell.com>
Date:   Thu Jun 13 11:29:42 2019 +0300

    qed: iWARP - Fix tc for MPA ll2 connection
```

```
    [ Upstream commit cb94d52b93c74fe1f2595734fabeda9f8ae891ee ]

    The driver needs to assign a lossless traffic class for the MPA ll2
    connection to ensure no packets are dropped when returning from the
    driver as they will never be re-transmitted by the peer.

    Fixes: ae3488ff37dc ("qed: Add ll2 connection for processing unaligned MPA packets")
    Signed-off-by: Ariel Elior <ariel.elior@marvell.com>
    Signed-off-by: Michal Kalderon <michal.kalderon@marvell.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2826cc1782c8acaf98698233d934aa0ff3e7b4fe
Author: Aaron Lewis <aaronlewis@google.com>
Date:   Wed Jun 5 15:02:52 2019 -0700

    x86/cpufeatures: Add FDP_EXCPTN_ONLY and ZERO_FCS_FDS

    [ Upstream commit cbb99c0f588737ec98c333558922ce47e9a95827 ]

    Add the CPUID enumeration for Intel's de-feature bits to accommodate
    passing these de-features through to kvm guests.

    These de-features are (from SDM vol 1, section 8.1.8):
     - X86_FEATURE_FDP_EXCPTN_ONLY: If CPUID.(EAX=07H,ECX=0H):EBX[bit 6] = 1, the
       data pointer (FDP) is updated only for the x87 non-control instructions that
       incur unmasked x87 exceptions.
     - X86_FEATURE_ZERO_FCS_FDS: If CPUID.(EAX=07H,ECX=0H):EBX[bit 13] = 1, the
       processor deprecates FCS and FDS; it saves each as 0000H.

    Signed-off-by: Aaron Lewis <aaronlewis@google.com>
    Signed-off-by: Borislav Petkov <bp@suse.de>
    Reviewed-by: Jim Mattson <jmattson@google.com>
    Cc: Fenghua Yu <fenghua.yu@intel.com>
    Cc: Frederic Weisbecker <frederic@kernel.org>
    Cc: "H. Peter Anvin" <hpa@zytor.com>
    Cc: Ingo Molnar <mingo@redhat.com>
    Cc: Konrad Rzeszutek Wilk <konrad.wilk@oracle.com>
    Cc: marcorr@google.com
    Cc: Peter Feiner <pfeiner@google.com>
    Cc: pshier@google.com
    Cc: Robert Hoo <robert.hu@linux.intel.com>
    Cc: Thomas Gleixner <tglx@linutronix.de>
    Cc: Thomas Lendacky <Thomas.Lendacky@amd.com>
    Cc: x86-ml <x86@kernel.org>
    Link: https://lkml.kernel.org/r/20190605220252.103406-1-aaronlewis@google.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 412121857fe45e05c6844bfaf7d12e622d8cd94d
Author: Rajneesh Bhardwaj <rajneesh.bhardwaj@linux.intel.com>
Date:   Fri Jun 14 13:47:01 2019 +0530

    perf/x86: Add Intel Ice Lake NNPI uncore support

    [ Upstream commit 5f4318c1b1d23a9290e4def78ee76017c288bf60 ]

    Intel Ice Lake uncore support already included IMC PCI ID but ICL-NNPI
    CPUID is missing so add it to fix the probe function.

    Fixes: e39875d15ad6 ("perf/x86: add Intel Icelake uncore support")
    Signed-off-by: Rajneesh Bhardwaj <rajneesh.bhardwaj@linux.intel.com>
    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Acked-by: Peter Zijlstra <peterz@infradead.org>
    Cc: alexander.shishkin@linux.intel.com
    Cc: Dave Hansen <dave.hansen@linux.intel.com>
```

```
    Cc: Andy Shevchenko <andriy.shevchenko@linux.intel.com>
    Cc: "H. Peter Anvin" <hpa@zytor.com>
    Cc: Kan Liang <kan.liang@linux.intel.com>
    Cc: Qiuxu Zhuo <qiuxu.zhuo@intel.com>
    Cc: Srinivas Pandruvada <srinivas.pandruvada@linux.intel.com>
    Cc: Len Brown <lenb@kernel.org>
    Cc: Linux PM <linux-pm@vger.kernel.org>
    Link: https://lkml.kernel.org/r/20190614081701.13828-1-rajneesh.bhardwaj@linux.intel.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 865dcf44bda2185c431f7bae5700e9419bc12f12
Author: Waiman Long <longman@redhat.com>
Date:   Tue May 21 16:48:43 2019 -0400

    rcu: Force inlining of rcu_read_lock()

    [ Upstream commit 6da9f775175e516fc7229ceaa9b54f8f56aa7924 ]

    When debugging options are turned on, the rcu_read_lock() function
    might not be inlined. This results in lockdep's print_lock() function
    printing "rcu_read_lock+0x0/0x70" instead of rcu_read_lock()'s caller.
    For example:

    [   10.579995] ============================
    [   10.584033] WARNING: suspicious RCU usage
    [   10.588074] 4.18.0.memcg_v2+ #1 Not tainted
    [   10.593162] ----------------------------
    [   10.597203] include/linux/rcupdate.h:281 Illegal context switch in
    RCU read-side critical section!
    [   10.606220]
    [   10.606220] other info that might help us debug this:
    [   10.606220]
    [   10.614280]
    [   10.614280] rcu_scheduler_active = 2, debug_locks = 1
    [   10.620853] 3 locks held by systemd/1:
    [   10.624632]  #0: (____ptrval____) (&type->i_mutex_dir_key#5){.+.+}, at:
lookup_slow+0x42/0x70
    [   10.633232]  #1: (____ptrval____) (rcu_read_lock){....}, at: rcu_read_lock+0x0/0x70
    [   10.640954]  #2: (____ptrval____) (rcu_read_lock){....}, at: rcu_read_lock+0x0/0x70

    These "rcu_read_lock+0x0/0x70" strings are not providing any useful
    information.  This commit therefore forces inlining of the rcu_read_lock()
    function so that rcu_read_lock()'s caller is instead shown.

    Signed-off-by: Waiman Long <longman@redhat.com>
    Signed-off-by: Paul E. McKenney <paulmck@linux.ibm.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e05a270a707a9e67bb41a38eeaab840fd3df6ffa
Author: Jerome Brunet <jbrunet@baylibre.com>
Date:   Thu Jun 13 13:42:32 2019 +0200

    ASoC: meson: axg-tdm: fix sample clock inversion

    [ Upstream commit cb36ff785e868992e96e8b9e5a0c2822b680a9e2 ]

    The content of SND_SOC_DAIFMT_FORMAT_MASK is a number, not a bitfield,
    so the test to check if the format is i2s is wrong. Because of this the
    clock setting may be wrong. For example, the sample clock gets inverted
    in DSP B mode, when it should not.

    Fix the lrclk invert helper function

    Fixes: 1a11d88f499c ("ASoC: meson: add tdm formatter base driver")
    Signed-off-by: Jerome Brunet <jbrunet@baylibre.com>
```

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

commit 80e78141245be288f1125be8bbb319d22d7c5fa2
Author: Rajneesh Bhardwaj <rajneesh.bhardwaj@linux.intel.com>
Date:   Thu Jun 6 06:54:19 2019 +0530

    x86/cpu: Add Ice Lake NNPI to Intel family

    [ Upstream commit e32d045cd4ba06b59878323e434bad010e78e658 ]

    Add the CPUID model number of Ice Lake Neural Network Processor for Deep
    Learning Inference (ICL-NNPI) to the Intel family list. Ice Lake NNPI uses
    model number 0x9D and this will be documented in a future version of Intel
    Software Development Manual.

    Signed-off-by: Rajneesh Bhardwaj <rajneesh.bhardwaj@linux.intel.com>
    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Cc: bp@suse.de
    Cc: Borislav Petkov <bp@alien8.de>
    Cc: Dave Hansen <dave.hansen@linux.intel.com>
    Cc: Andy Shevchenko <andriy.shevchenko@linux.intel.com>
    Cc: "H. Peter Anvin" <hpa@zytor.com>
    Cc: Kan Liang <kan.liang@linux.intel.com>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Cc: platform-driver-x86@vger.kernel.org
    Cc: Qiuxu Zhuo <qiuxu.zhuo@intel.com>
    Cc: Srinivas Pandruvada <srinivas.pandruvada@linux.intel.com>
    Cc: Len Brown <lenb@kernel.org>
    Cc: Linux PM <linux-pm@vger.kernel.org>
    Link: https://lkml.kernel.org/r/20190606012419.13250-1-rajneesh.bhardwaj@linux.intel.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d349f0ecd7e8d96eb74e14c7698d69fbcc907a9d
Author: Eric Biggers <ebiggers@google.com>
Date:   Sun Jun 2 22:42:33 2019 -0700

    crypto: testmgr - add some more preemption points

    [ Upstream commit e63e1b0dd0003dc31f73d875907432be3a2abe5d ]

    Call cond_resched() after each fuzz test iteration.  This avoids stall
    warnings if fuzz_iterations is set very high for testing purposes.

    While we're at it, also call cond_resched() after finishing testing each
    test vector.

    Signed-off-by: Eric Biggers <ebiggers@google.com>
    Acked-by: Ard Biesheuvel <ard.biesheuvel@linaro.org>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 09b9995f12f88ab80a1a3feb5ad784d0921bcb2b
Author: Ondrej Mosnacek <omosnace@redhat.com>
Date:   Wed Jun 12 10:12:26 2019 +0200

    selinux: fix empty write to keycreate file

    [ Upstream commit 464c258aa45b09f16aa0f05847ed8895873262d9 ]

    When sid == 0 (we are resetting keycreate_sid to the default value), we
    should skip the KEY__CREATE check.

    Before this patch, doing a zero-sized write to /proc/self/keycreate
    would check if the current task can create unlabeled keys (which would
```

```
    usually fail with -EACCESS and generate an AVC). Now it skips the check
    and correctly sets the task's keycreate_sid to 0.

    Bug report: https://bugzilla.redhat.com/show_bug.cgi?id=1719067

    Tested using the reproducer from the report above.

    Fixes: 4eb582cf1fbd ("[PATCH] keys: add a way to store the appropriate context for
newly-created keys")
    Reported-by: Kir Kolyshkin <kir@sacred.ru>
    Signed-off-by: Ondrej Mosnacek <omosnace@redhat.com>
    Signed-off-by: Paul Moore <paul@paul-moore.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4cb2edfa241099a3e2feb0e3457626fdc5836662
Author: Marek Szyprowski <m.szyprowski@samsung.com>
Date:   Wed Jun 12 09:57:57 2019 -0400

    media: s5p-mfc: fix reading min scratch buffer size on MFC v6/v7

    [ Upstream commit be22203aec440c1761ce8542c2636ac6c8951e3a ]

    MFC v6 and v7 has no register to read min scratch buffer size, so it has
    to be read conditionally only if hardware supports it. This fixes following
    NULL pointer exception on SoCs with MFC v6/v7:

    8<--- cut here ---
    Unable to handle kernel NULL pointer dereference at virtual address 00000000
    pgd = f25837f9
    [00000000] *pgd=bd93d835
    Internal error: Oops: 17 [#1] PREEMPT SMP ARM
    Modules linked in: btmrvl_sdio btmrvl bluetooth mwifiex_sdio mwifiex ecdh_generic ecc
    Hardware name: SAMSUNG EXYNOS (Flattened Device Tree)
    PC is at s5p_mfc_get_min_scratch_buf_size+0x30/0x3c
    LR is at s5p_mfc_get_min_scratch_buf_size+0x28/0x3c
    ...
    [<c074f998>] (s5p_mfc_get_min_scratch_buf_size) from [<c0745bc0>]
(s5p_mfc_irq+0x814/0xa5c)
    [<c0745bc0>] (s5p_mfc_irq) from [<c019a218>] (__handle_irq_event_percpu+0x64/0x3f8)
    [<c019a218>] (__handle_irq_event_percpu) from [<c019a5d8>]
(handle_irq_event_percpu+0x2c/0x7c)
    [<c019a5d8>] (handle_irq_event_percpu) from [<c019a660>] (handle_irq_event+0x38/0x5c)
    [<c019a660>] (handle_irq_event) from [<c019ebc4>] (handle_fasteoi_irq+0xc4/0x180)
    [<c019ebc4>] (handle_fasteoi_irq) from [<c0199270>] (generic_handle_irq+0x24/0x34)
    [<c0199270>] (generic_handle_irq) from [<c0199888>] (__handle_domain_irq+0x7c/0xec)
    [<c0199888>] (__handle_domain_irq) from [<c04ac298>] (gic_handle_irq+0x58/0x9c)
    [<c04ac298>] (gic_handle_irq) from [<c0101ab0>] (__irq_svc+0x70/0xb0)
    Exception stack(0xe73ddc60 to 0xe73ddca8)
    ...
    [<c0101ab0>] (__irq_svc) from [<c01967d8>] (console_unlock+0x5a8/0x6a8)
    [<c01967d8>] (console_unlock) from [<c01981d0>] (vprintk_emit+0x118/0x2d8)
    [<c01981d0>] (vprintk_emit) from [<c01983b0>] (vprintk_default+0x20/0x28)
    [<c01983b0>] (vprintk_default) from [<c01989b4>] (printk+0x30/0x54)
    [<c01989b4>] (printk) from [<c07500b8>] (s5p_mfc_init_decode_v6+0x1d4/0x284)
    [<c07500b8>] (s5p_mfc_init_decode_v6) from [<c07230d0>] (vb2_start_streaming+0x24/0x150)
    [<c07230d0>] (vb2_start_streaming) from [<c0724e4c>] (vb2_core_streamon+0x11c/0x15c)
    [<c0724e4c>] (vb2_core_streamon) from [<c07478b8>] (vidioc_streamon+0x64/0xa0)
    [<c07478b8>] (vidioc_streamon) from [<c0709640>] (__video_do_ioctl+0x28c/0x45c)
    [<c0709640>] (__video_do_ioctl) from [<c0709bc8>] (video_usercopy+0x260/0x8a4)
    [<c0709bc8>] (video_usercopy) from [<c02b3820>] (do_vfs_ioctl+0xb0/0x9fc)
    [<c02b3820>] (do_vfs_ioctl) from [<c02b41a0>] (ksys_ioctl+0x34/0x58)
    [<c02b41a0>] (ksys_ioctl) from [<c0101000>] (ret_fast_syscall+0x0/0x28)
    Exception stack(0xe73ddfa8 to 0xe73ddff0)
    ...
    ---[ end trace 376cf5ba6e0bee93 ]---
```

```
    Signed-off-by: Marek Szyprowski <m.szyprowski@samsung.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 e6164c5cf38b87b2a71620216d6cb7c09159ff50
Author: Valdis Klētnieks <valdis.kletnieks@vt.edu>
Date:   Thu Jun 6 22:39:27 2019 -0400

    bpf: silence warning messages in core

    [ Upstream commit aee450cbe482a8c2f6fa5b05b178ef8b8ff107ca ]

    Compiling kernel/bpf/core.c with W=1 causes a flood of warnings:

    kernel/bpf/core.c:1198:65: warning: initialized field overwritten [-Woverride-init]
     1198 | #define BPF_INSN_3_TBL(x, y, z) [BPF_##x | BPF_##y | BPF_##z] = true
          |                                                                 ^~~~
    kernel/bpf/core.c:1087:2: note: in expansion of macro 'BPF_INSN_3_TBL'
     1087 |  INSN_3(ALU, ADD,  X),   \
          |  ^~~~~
    kernel/bpf/core.c:1202:3: note: in expansion of macro 'BPF_INSN_MAP'
     1202 |   BPF_INSN_MAP(BPF_INSN_2_TBL, BPF_INSN_3_TBL),
          |   ^~~~~~~~~~~~
    kernel/bpf/core.c:1198:65: note: (near initialization for 'public_insntable[12]')
     1198 | #define BPF_INSN_3_TBL(x, y, z) [BPF_##x | BPF_##y | BPF_##z] = true
          |                                                                 ^~~~
    kernel/bpf/core.c:1087:2: note: in expansion of macro 'BPF_INSN_3_TBL'
     1087 |  INSN_3(ALU, ADD,  X),   \
          |  ^~~~~
    kernel/bpf/core.c:1202:3: note: in expansion of macro 'BPF_INSN_MAP'
     1202 |   BPF_INSN_MAP(BPF_INSN_2_TBL, BPF_INSN_3_TBL),
          |   ^~~~~~~~~~~~

    98 copies of the above.

    The attached patch silences the warnings, because we *know* we're overwriting
    the default initializer. That leaves bpf/core.c with only 6 other warnings,
    which become more visible in comparison.

    Signed-off-by: Valdis Kletnieks <valdis.kletnieks@vt.edu>
    Acked-by: Andrii Nakryiko <andriin@fb.com>
    Signed-off-by: Daniel Borkmann <daniel@iogearbox.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9ee7f7b3be04d8ad50e7677d650a8e03b4b646e3
Author: Young Xiao <92siuyang@gmail.com>
Date:   Tue Jun 4 08:26:33 2019 -0400

    media: davinci: vpif_capture: fix memory leak in vpif_probe()

    [ Upstream commit 64f883cd98c6d43013fb0cea788b63e50ebc068c ]

    If vpif_probe() fails on v4l2_device_register() and vpif_probe_complete(),
    then memory allocated at initialize_vpif() for global vpif_obj.dev[i]
    become unreleased.

    The patch adds deallocation of vpif_obj.dev[i] on the error path.

    Signed-off-by: Young Xiao <92siuyang@gmail.com>
    Acked-by: Lad, Prabhakar <prabhakar.csengg@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 36dd2bcba50835d81b1adcb171e9437316020ca6
Author: Tony Lindgren <tony@atomide.com>
Date:   Tue Jun 11 23:33:52 2019 -0700

    gpio: omap: Fix lost edge wake-up interrupts

    [ Upstream commit a522f1d0c381c42f3ace13b8bbeeccabdd6d2e5c ]

    If an edge interrupt triggers while entering idle just before we save
    GPIO datain register to saved_datain, the triggered GPIO will not be
    noticed on wake-up. This is because the saved_datain and GPIO datain
    are the same on wake-up in omap_gpio_unidle(). Let's fix this by
    ignoring any pending edge interrupts for saved_datain.

    This issue affects only idle states where the GPIO module internal
    wake-up path is operational. For deeper idle states where the GPIO
    module gets powered off, Linux generic wakeirqs must be used for
    the padconf wake-up events with pinctrl-single driver. For examples,
    please see "interrupts-extended" dts usage in many drivers.

    This issue can be somewhat easily reproduced by pinging an idle system
    with smsc911x Ethernet interface configured IRQ_TYPE_EDGE_FALLING. At
    some point the smsc911x interrupts will just stop triggering. Also if
    WLCORE WLAN is used with EDGE interrupt like it's documentation specifies,
    we can see lost interrupts without this patch.

    Note that in the long run we may be able to cancel entering idle by
    returning an error in gpio_omap_cpu_notifier() on pending interrupts.
    But let's fix the bug first.

    Also note that because of the recent clean-up efforts this patch does
    not apply directly to older kernels. This does fix a long term issue
    though, and can be backported as needed.

    Cc: Aaro Koskinen <aaro.koskinen@iki.fi>
    Cc: Grygorii Strashko <grygorii.strashko@ti.com>
    Cc: Keerthy <j-keerthy@ti.com>
    Cc: Ladislav Michl <ladis@linux-mips.org>
    Cc: Peter Ujfalusi <peter.ujfalusi@ti.com>
    Cc: Russell King <rmk+kernel@armlinux.org.uk>
    Cc: Tero Kristo <t-kristo@ti.com>
    Signed-off-by: Tony Lindgren <tony@atomide.com>
    Signed-off-by: Linus Walleij <linus.walleij@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4651fcc68315a60d7db2d22ffcc842f9b5638e23
Author: Srinivas Kandagatla <srinivas.kandagatla@linaro.org>
Date:   Wed Jun 12 12:03:43 2019 +0100

    regmap: fix bulk writes on paged registers

    [ Upstream commit db057679de3e9e6a03c1bcd5aee09b0d25fd9f5b ]

    On buses like SlimBus and SoundWire which does not support
    gather_writes yet in regmap, A bulk write on paged register
    would be silently ignored after programming page.
    This is because local variable 'ret' value in regmap_raw_write_impl()
    gets reset to 0 once page register is written successfully and the
    code below checks for 'ret' value to be -ENOTSUPP before linearising
    the write buffer to send to bus->write().

    Fix this by resetting the 'ret' value to -ENOTSUPP in cases where
    gather_writes() is not supported or single register write is
    not possible.
```

```
    Signed-off-by: Srinivas Kandagatla <srinivas.kandagatla@linaro.org>
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4bd220bd998ea21c5dda2583bd0b25c35330897c
Author: Russell King <rmk+kernel@armlinux.org.uk>
Date:   Mon Jun 10 20:10:44 2019 +0300

    gpio: omap: ensure irq is enabled before wakeup

    [ Upstream commit c859e0d479b3b4f6132fc12637c51e01492f31f6 ]

    Documentation states:

      NOTE: There must be a correlation between the wake-up enable and
      interrupt-enable registers. If a GPIO pin has a wake-up configured
      on it, it must also have the corresponding interrupt enabled (on
      one of the two interrupt lines).

    Ensure that this condition is always satisfied by enabling the detection
    events after enabling the interrupt, and disabling the detection before
    disabling the interrupt.  This ensures interrupt/wakeup events can not
    happen until both the wakeup and interrupt enables correlate.

    If we do any clearing, clear between the interrupt enable/disable and
    trigger setting.

    Signed-off-by: Russell King <rmk+kernel@armlinux.org.uk>
    Signed-off-by: Grygorii Strashko <grygorii.strashko@ti.com>
    Tested-by: Tony Lindgren <tony@atomide.com>
    Signed-off-by: Linus Walleij <linus.walleij@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8bc9c0dadb61e11b9ddcca68c161481a41a62ba3
Author: Russell King <rmk+kernel@armlinux.org.uk>
Date:   Mon Jun 10 20:10:45 2019 +0300

    gpio: omap: fix lack of irqstatus_raw0 for OMAP4

    [ Upstream commit 64ea3e9094a1f13b96c33244a3fb3a0f45690bd2 ]

    Commit 384ebe1c2849 ("gpio/omap: Add DT support to GPIO driver") added
    the register definition tables to the gpio-omap driver. Subsequently to
    that commit, commit 4e962e8998cc ("gpio/omap: remove cpu_is_omapxxxx()
    checks from *_runtime_resume()") added definitions for irqstatus_raw*
    registers to the legacy OMAP4 definitions, but missed the DT
    definitions.

    This causes an unintentional change of behaviour for the 1.101 errata
    workaround on OMAP4 platforms. Fix this oversight.

    Fixes: 4e962e8998cc ("gpio/omap: remove cpu_is_omapxxxx() checks from
*_runtime_resume()")
    Signed-off-by: Russell King <rmk+kernel@armlinux.org.uk>
    Signed-off-by: Grygorii Strashko <grygorii.strashko@ti.com>
    Tested-by: Tony Lindgren <tony@atomide.com>
    Signed-off-by: Linus Walleij <linus.walleij@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 725fb018531f72dff4fc0a4a01e191451db1e538
Author: Eric Auger <eric.auger@redhat.com>
Date:   Mon Jun 3 08:53:30 2019 +0200

    iommu: Fix a leak in iommu_insert_resv_region
```

```
    [ Upstream commit ad0834dedaa15c3a176f783c0373f836e44b4700 ]

    In case we expand an existing region, we unlink
    this latter and insert the larger one. In
    that case we should free the original region after
    the insertion. Also we can immediately return.

    Fixes: 6c65fb318e8b ("iommu: iommu_get_group_resv_regions")

    Signed-off-by: Eric Auger <eric.auger@redhat.com>
    Signed-off-by: Joerg Roedel <jroedel@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0a1c9f326f894791b7bccec3d8113800cd47c265
Author: Kieran Bingham <kieran.bingham+renesas@ideasonboard.com>
Date:   Wed May 15 11:39:12 2019 -0400

    media: fdp1: Support M3N and E3 platforms

    [ Upstream commit 4e8c120de9268fc26f583268b9d22e7d37c4595f ]

    New Gen3 R-Car platforms incorporate the FDP1 with an updated version
    register. No code change is required to support these targets, but they
    will currently report an error stating that the device can not be
    identified.

    Update the driver to match against the new device types.

    Signed-off-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 b3a2e4cf4e21e5b8b0a0def56a316395642d8cf0
Author: Oliver Neukum <oneukum@suse.com>
Date:   Tue Apr 30 08:28:14 2019 -0400

    media: uvcvideo: Fix access to uninitialized fields on probe error

    [ Upstream commit 11a087f484bf15ff65f0a9f277aa5a61fd07ed2a ]

    We need to check whether this work we are canceling actually is
    initialized.

    Signed-off-by: Oliver Neukum <oneukum@suse.com>
    Reported-by: syzbot+2e1ef9188251d9cc7944@syzkaller.appspotmail.com
    Signed-off-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 16e08707043a7e1c4f021309f56c89b5215ec517
Author: Xingyu Chen <xingyu.chen@amlogic.com>
Date:   Sat Jun 8 21:04:10 2019 +0200

    irqchip/meson-gpio: Add support for Meson-G12A SoC

    [ Upstream commit c64a9e804ccf86eb202bfd1c6a8c5233c75a0431 ]

    The Meson-G12A SoC uses the same GPIO interrupt controller IP block as the
    other Meson SoCs, A totle of 100 pins can be spied on, which is the sum of:

    - 223:100 undefined (no interrupt)
    - 99:97   3 pins on bank GPIOE
    - 96:77   20 pins on bank GPIOX
```

```
      - 76:61   16 pins on bank GPIOA
      - 60:53   8 pins on bank GPIOC
      - 52:37   16 pins on bank BOOT
      - 36:28   9 pins on bank GPIOH
      - 27:12   16 pins on bank GPIOZ
      - 11:0    12 pins in the AO domain

      Signed-off-by: Xingyu Chen <xingyu.chen@amlogic.com>
      Signed-off-by: Jianxin Pan <jianxin.pan@amlogic.com>
      Signed-off-by: Martin Blumenstingl <martin.blumenstingl@googlemail.com>
      Signed-off-by: Marc Zyngier <marc.zyngier@arm.com>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 833ead63413a4f36d64d7f66471b36a7ec4bf402
  Author: Hechao Li <hechaol@fb.com>
  Date:   Mon Jun 10 17:43:07 2019 -0700

      selftests/bpf : clean up feature/ when make clean

      [ Upstream commit 89cceaa939171fafa153d4bf637b39e396bbd785 ]

      An error "implicit declaration of function 'reallocarray'" can be thrown
      with the following steps:

      $ cd tools/testing/selftests/bpf
      $ make clean && make CC=<Path to GCC 4.8.5>
      $ make clean && make CC=<Path to GCC 7.x>

      The cause is that the feature folder generated by GCC 4.8.5 is not
      removed, leaving feature-reallocarray being 1, which causes reallocarray
      not defined when re-compliing with GCC 7.x. This diff adds feature
      folder to EXTRA_CLEAN to avoid this problem.

      v2: Rephrase the commit message.

      Signed-off-by: Hechao Li <hechaol@fb.com>
      Acked-by: Andrii Nakryiko <andriin@fb.com>
      Signed-off-by: Daniel Borkmann <daniel@iogearbox.net>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 5cd8950672dafa7f3fff678b3b69d65016e8480d
  Author: Thomas Richter <tmricht@linux.ibm.com>
  Date:   Thu May 23 10:25:21 2019 +0200

      perf report: Fix OOM error in TUI mode on s390

      [ Upstream commit 8a07aa4e9b7b0222129c07afff81634a884b2866 ]

      Debugging a OOM error using the TUI interface revealed this issue
      on s390:

      [tmricht@m83lp54 perf]$ cat /proc/kallsyms |sort
      ....
      00000001119b7158 B radix_tree_node_cachep
      00000001119b8000 B __bss_stop
      00000001119b8000 B _end
      000003ff80002850 t autofs_mount [autofs4]
      000003ff80002868 t autofs_show_options  [autofs4]
      000003ff80002a98 t autofs_evict_inode   [autofs4]
      ....

      There is a huge gap between the last kernel symbol
      __bss_stop/_end and the first kernel module symbol
      autofs_mount (from autofs4 module).
```

```
After reading the kernel symbol table via functions:

 dso__load()
 +--> dso__load_kernel_sym()
      +--> dso__load_kallsyms()
           +--> __dso_load_kallsyms()
                +--> symbols__fixup_end()

the symbol __bss_stop has a start address of 1119b8000 and
an end address of 3ff80002850, as can be seen by this debug statement:

  symbols__fixup_end __bss_stop start:0x1119b8000 end:0x3ff80002850

The size of symbol __bss_stop is 0x3fe6e64a850 bytes!
It is the last kernel symbol and fills up the space until
the first kernel module symbol.

This size kills the TUI interface when executing the following
code:

  process_sample_event()
    hist_entry_iter__add()
      hist_iter__report_callback()
        hist_entry__inc_addr_samples()
          symbol__inc_addr_samples(symbol = __bss_stop)
            symbol__cycles_hist()
               annotated_source__alloc_histograms(...,
                                                  symbol__size(sym),
                                                  ...)

This function allocates memory to save sample histograms.
The symbol_size() marco is defined as sym->end - sym->start, which
results in above value of 0x3fe6e64a850 bytes and
the call to calloc() in annotated_source__alloc_histograms() fails.

The histgram memory allocation might fail, make this failure
no-fatal and continue processing.

Output before:
[tmricht@m83lp54 perf]$ ./perf --debug stderr=1 report -vvvvv \
                                                  -i ~/slow.data 2>/tmp/2
[tmricht@m83lp54 perf]$ tail -5 /tmp/2
  __symbol__inc_addr_samples(875): ENOMEM! sym->name=__bss_stop,
               start=0x1119b8000, addr=0x2aa0005eb08, end=0x3ff80002850,
               func: 0
problem adding hist entry, skipping event
0x938b8 [0x8]: failed to process type: 68 [Cannot allocate memory]
[tmricht@m83lp54 perf]$

Output after:
[tmricht@m83lp54 perf]$ ./perf --debug stderr=1 report -vvvvv \
                                                  -i ~/slow.data 2>/tmp/2
[tmricht@m83lp54 perf]$ tail -5 /tmp/2
   symbol__inc_addr_samples map:0x1597830 start:0x110730000 end:0x3ff80002850
   symbol__hists notes->src:0x2aa2a70 nr_hists:1
   symbol__inc_addr_samples sym:unlink_anon_vmas src:0x2aa2a70
   __symbol__inc_addr_samples: addr=0x11094c69e
   0x11094c670 unlink_anon_vmas: period++ [addr: 0x11094c69e, 0x2e, evidx=0]
        => nr_samples: 1, period: 526008
[tmricht@m83lp54 perf]$

There is no error about failed memory allocation and the TUI interface
shows all entries.

Signed-off-by: Thomas Richter <tmricht@linux.ibm.com>
```

```
    Reviewed-by: Hendrik Brueckner <brueckner@linux.ibm.com>
    Cc: Heiko Carstens <heiko.carstens@de.ibm.com>
    Cc: Hendrik Brueckner <brueckner@linux.vnet.ibm.com>
    Link: http://lkml.kernel.org/r/90cb5607-3e12-5167-682d-978eba7dafa8@linux.ibm.com
    Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9f13cb2d5847df56cc55eba5af5ea766f0e46ae9
Author: Thomas Richter <tmricht@linux.ibm.com>
Date:   Tue Jun 4 07:35:04 2019 +0200

    perf test 6: Fix missing kvm module load for s390

    [ Upstream commit 53fe307dfd309e425b171f6272d64296a54f4dff ]

    Command

       # perf test -Fv 6

    fails with error

       running test 100 'kvm-s390:kvm_s390_create_vm' failed to parse
        event 'kvm-s390:kvm_s390_create_vm', err -1, str 'unknown tracepoint'
        event syntax error: 'kvm-s390:kvm_s390_create_vm'
                             \___ unknown tracepoint

    when the kvm module is not loaded or not built in.

    Fix this by adding a valid function which tests if the module
    is loaded. Loaded modules (or builtin KVM support) have a
    directory named
      /sys/kernel/debug/tracing/events/kvm-s390
    for this tracepoint.

    Check for existence of this directory.

    Signed-off-by: Thomas Richter <tmricht@linux.ibm.com>
    Reviewed-by: Christian Borntraeger <borntraeger@de.ibm.com>
    Cc: Heiko Carstens <heiko.carstens@de.ibm.com>
    Cc: Hendrik Brueckner <brueckner@linux.vnet.ibm.com>
    Link: http://lkml.kernel.org/r/20190604053504.43073-1-tmricht@linux.ibm.com
    Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e70d9ff97ab0a080177450fd7238125ea6351e07
Author: Mathieu Poirier <mathieu.poirier@linaro.org>
Date:   Wed Jun 5 10:16:33 2019 -0600

    perf cs-etm: Properly set the value of 'old' and 'head' in snapshot mode

    [ Upstream commit e45c48a9a4d20ebc7b639a62c3ef8f4b08007027 ]

    This patch adds the necessary intelligence to properly compute the value
    of 'old' and 'head' when operating in snapshot mode.  That way we can
    get the latest information in the AUX buffer and be compatible with the
    generic AUX ring buffer mechanic.

    Tester notes:

    > Leo, have you had the chance to test/review this one? Suzuki?

    Sure.  I applied this patch on the perf/core branch (with latest
    commit 3e4fbf36c1e3 'perf augmented_raw_syscalls: Move reading
    filename to the loop') and passed testing with below steps:
```

```
       # perf record -e cs_etm/@tmc_etr0/ -S -m,64 --per-thread ./sort &
       [1] 19097
       Bubble sorting array of 30000 elements

       # kill -USR2 19097
       # kill -USR2 19097
       # kill -USR2 19097
       [ perf record: Woken up 4 times to write data ]
       [ perf record: Captured and wrote 0.753 MB perf.data ]
```

```
    Signed-off-by: Mathieu Poirier <mathieu.poirier@linaro.org>
    Tested-by: Leo Yan <leo.yan@linaro.org>
    Cc: Alexander Shishkin <alexander.shishkin@linux.intel.com>
    Cc: Jiri Olsa <jolsa@redhat.com>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Cc: Suzuki Poulouse <suzuki.poulose@arm.com>
    Cc: linux-arm-kernel@lists.infradead.org
    Link: http://lkml.kernel.org/r/20190605161633.12245-1-mathieu.poirier@linaro.org
    Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 20d293588c5f4bcd0f6eb830a6744b6a9e30b311
Author: Stefano Brivio <sbrivio@redhat.com>
Date:   Sun May 26 23:14:06 2019 +0200

    ipset: Fix memory accounting for hash types on resize

    [ Upstream commit 11921796f4799ca9c61c4b22cc54d84aa69f8a35 ]

    If a fresh array block is allocated during resize, the current in-memory
    set size should be increased by the size of the block, not replaced by it.

    Before the fix, adding entries to a hash set type, leading to a table
    resize, caused an inconsistent memory size to be reported. This becomes
    more obvious when swapping sets with similar sizes:

      # cat hash_ip_size.sh
      #!/bin/sh
      FAIL_RETRIES=10

      tries=0
      while [ ${tries} -lt ${FAIL_RETRIES} ]; do
            ipset create t1 hash:ip
            for i in `seq 1 4345`; do
                    ipset add t1 1.2.$((i / 255)).$((i % 255))
            done
            t1_init="$(ipset list t1|sed -n 's/Size in memory: \(.*\)/\1/p')"

            ipset create t2 hash:ip
            for i in `seq 1 4360`; do
                    ipset add t2 1.2.$((i / 255)).$((i % 255))
            done
            t2_init="$(ipset list t2|sed -n 's/Size in memory: \(.*\)/\1/p')"

            ipset swap t1 t2
            t1_swap="$(ipset list t1|sed -n 's/Size in memory: \(.*\)/\1/p')"
            t2_swap="$(ipset list t2|sed -n 's/Size in memory: \(.*\)/\1/p')"

            ipset destroy t1
            ipset destroy t2
            tries=$((tries + 1))

            if [ ${t1_init} -lt 10000 ] || [ ${t2_init} -lt 10000 ]; then
                    echo "FAIL after ${tries} tries:"
                    echo "T1 size ${t1_init}, after swap ${t1_swap}"
```

```
                    echo "T2 size ${t2_init}, after swap ${t2_swap}"
                    exit 1
            fi
      done
      echo "PASS"
      # echo -n 'func hash_ip4_resize +p' > /sys/kernel/debug/dynamic_debug/control
      # ./hash_ip_size.sh
      [ 2035.018673] attempt to resize set t1 from 10 to 11, t 00000000fe6551fa
      [ 2035.078583] set t1 resized from 10 (00000000fe6551fa) to 11 (00000000172a0163)
      [ 2035.080353] Table destroy by resize 00000000fe6551fa
      FAIL after 4 tries:
      T1 size 9064, after swap 71128
      T2 size 71128, after swap 9064

    Reported-by: NOYB <JunkYardMail1@Frontier.com>
    Fixes: 9e41f26a505c ("netfilter: ipset: Count non-static extension memory for
userspace")
    Signed-off-by: Stefano Brivio <sbrivio@redhat.com>
    Signed-off-by: Jozsef Kadlecsik <kadlec@blackhole.kfki.hu>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit acce347db830923f5458f52ff4b91e15129b1b3e
Author: Aditya Pakki <pakki001@umn.edu>
Date:   Mon Jun 10 12:47:37 2019 +0200

    netfilter: ipset: fix a missing check of nla_parse

    [ Upstream commit f4f5748bfec94cf418e49bf05f0c81a1b9ebc950 ]

    When nla_parse fails, we should not use the results (the first
    argument). The fix checks if it fails, and if so, returns its error code
    upstream.

    Signed-off-by: Aditya Pakki <pakki001@umn.edu>
    Signed-off-by: Jozsef Kadlecsik <kadlec@blackhole.kfki.hu>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4c48018f786d72acc6f76d7bb4a2e69a4d01ab09
Author: Robert Hancock <hancock@sedsystems.ca>
Date:   Fri Jun 7 10:42:36 2019 -0600

    net: sfp: add mutex to prevent concurrent state checks

    [ Upstream commit 2158e856f56bb762ef90f3ec244d41a519826f75 ]

    sfp_check_state can potentially be called by both a threaded IRQ handler
    and delayed work. If it is concurrently called, it could result in
    incorrect state management. Add a st_mutex to protect the state - this
    lock gets taken outside of code that checks and handle state changes, and
    the existing sm_mutex nests inside of it.

    Suggested-by: Russell King <rmk+kernel@armlinux.org.uk>
    Signed-off-by: Robert Hancock <hancock@sedsystems.ca>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1277ef7670b6305c4c6d9456100da8abde94b9eb
Author: Borislav Petkov <bp@suse.de>
Date:   Sat Apr 20 12:53:05 2019 +0200

    RAS/CEC: Fix pfn insertion

    [ Upstream commit 6d8e294bf5f0e85c34e8b14b064e2965f53f38b0 ]

    When inserting random PFNs for debugging the CEC through
```

```
    (debugfs)/ras/cec/pfn, depending on the return value of pfn_set(),
    multiple values get inserted per a single write.

    That is because simple_attr_write() interprets a retval of 0 as
    success and claims the whole input. However, pfn_set() returns the
    cec_add_elem() value, which, if > 0 and smaller than the whole input
    length, makes glibc continue issuing the write syscall until there's
    input left:

      pfn_set
      simple_attr_write
      debugfs_attr_write
      full_proxy_write
      vfs_write
      ksys_write
      do_syscall_64
      entry_SYSCALL_64_after_hwframe

    leading to those repeated calls.

    Return 0 to fix that.

    Signed-off-by: Borislav Petkov <bp@suse.de>
    Cc: Tony Luck <tony.luck@intel.com>
    Cc: linux-edac <linux-edac@vger.kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 95d0b4bff4d4e68a1dee15322af240dfb74f645f
Author: Julian Wiedmann <jwi@linux.ibm.com>
Date:   Mon Jun 3 07:47:04 2019 +0200

    s390/qdio: handle PENDING state for QEBSM devices

    [ Upstream commit 04310324c6f482921c071444833e70fe861b73d9 ]

    When a CQ-enabled device uses QEBSM for SBAL state inspection,
    get_buf_states() can return the PENDING state for an Output Queue.
    get_outbound_buffer_frontier() isn't prepared for this, and any PENDING
    buffer will permanently stall all further completion processing on this
    Queue.

    This isn't a concern for non-QEBSM devices, as get_buf_states() for such
    devices will manually turn PENDING buffers into EMPTY ones.

    Fixes: 104ea556ee7f ("qdio: support asynchronous delivery of storage blocks")
    Signed-off-by: Julian Wiedmann <jwi@linux.ibm.com>
    Signed-off-by: Heiko Carstens <heiko.carstens@de.ibm.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 506c6ccedf71dad2a8637fb4599243513e1484a7
Author: Robert Hancock <hancock@sedsystems.ca>
Date:   Thu Jun 6 16:28:17 2019 -0600

    net: axienet: Fix race condition causing TX hang

    [ Upstream commit 7de44285c1f69ccfbe8be1d6a16fcd956681fee6 ]

    It is possible that the interrupt handler fires and frees up space in
    the TX ring in between checking for sufficient TX ring space and
    stopping the TX queue in axienet_start_xmit. If this happens, the
    queue wake from the interrupt handler will occur before the queue is
    stopped, causing a lost wakeup and the adapter's transmit hanging.

    To avoid this, after stopping the queue, check again whether there is
    sufficient space in the TX ring. If so, wake up the queue again.
```

```
    Signed-off-by: Robert Hancock <hancock@sedsystems.ca>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cfc2d88848b3b60dbb543df503038f3f2d5dbbbe
Author: Fabio Estevam <festevam@gmail.com>
Date:   Thu Jun 6 09:40:33 2019 -0300

    net: fec: Do not use netdev messages too early

    [ Upstream commit a19a0582363b9a5f8ba812f34f1b8df394898780 ]

    When a valid MAC address is not found the current messages
    are shown:

    fec 2188000.ethernet (unnamed net_device) (uninitialized): Invalid MAC address:
00:00:00:00:00:00
    fec 2188000.ethernet (unnamed net_device) (uninitialized): Using random MAC address:
aa:9f:25:eb:7e:aa

    Since the network device has not been registered at this point, it is better
    to use dev_err()/dev_info() instead, which will provide cleaner log
    messages like these:

    fec 2188000.ethernet: Invalid MAC address: 00:00:00:00:00:00
    fec 2188000.ethernet: Using random MAC address: aa:9f:25:eb:7e:aa

    Tested on a imx6dl-pico-pi board.

    Signed-off-by: Fabio Estevam <festevam@gmail.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a5b94181aa976ccb4a2a0611169af36b6c89c368
Author: Antoine Tenart <antoine.tenart@bootlin.com>
Date:   Mon May 27 16:51:06 2019 +0200

    crypto: inside-secure - do not rely on the hardware last bit for result descriptors

    [ Upstream commit 893325904272356802 36b9470e851afc49b3caa1 ]

    When performing a transformation the hardware is given result
    descriptors to save the result data. Those result descriptors are
    batched using a 'first' and a 'last' bit. There are cases were more
    descriptors than needed are given to the engine, leading to the engine
    only using some of them, and not setting the last bit on the last
    descriptor we gave. This causes issues were the driver and the hardware
    aren't in sync anymore about the number of result descriptors given (as
    the driver do not give a pool of descriptor to use for any
    transformation, but a pool of descriptors to use *per* transformation).

    This patch fixes it by attaching the number of given result descriptors
    to the requests, and by using this number instead of the 'last' bit
    found on the descriptors to process them.

    Signed-off-by: Antoine Tenart <antoine.tenart@bootlin.com>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit faac7e78c489cd3debb9881a1bd40d4c5e2bb156
Author: Biao Huang <biao.huang@mediatek.com>
Date:   Mon Jun 3 09:58:05 2019 +0800

    net: stmmac: modify default value of tx-frames
```

```
    [ Upstream commit d2facb4b3983425f6776c24dd678a82dbe673773 ]

    the default value of tx-frames is 25, it's too late when
    passing tstamp to stack, then the ptp4l will fail:

    ptp4l -i eth0 -f gPTP.cfg -m
    ptp4l: selected /dev/ptp0 as PTP clock
    ptp4l: port 1: INITIALIZING to LISTENING on INITIALIZE
    ptp4l: port 0: INITIALIZING to LISTENING on INITIALIZE
    ptp4l: port 1: link up
    ptp4l: timed out while polling for tx timestamp
    ptp4l: increasing tx_timestamp_timeout may correct this issue,
           but it is likely caused by a driver bug
    ptp4l: port 1: send peer delay response failed
    ptp4l: port 1: LISTENING to FAULTY on FAULT_DETECTED (FT_UNSPECIFIED)

    ptp4l tests pass when changing the tx-frames from 25 to 1 with
    ethtool -C option.
    It should be fine to set tx-frames default value to 1, so ptp4l will pass
    by default.

    Signed-off-by: Biao Huang <biao.huang@mediatek.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 037f641e406d026dcb36031142a2bc834da5084d
Author: Biao Huang <biao.huang@mediatek.com>
Date:   Mon Jun 3 09:58:06 2019 +0800

    net: stmmac: dwmac4: fix flow control issue

    [ Upstream commit ee326fd01e79dfa42014d55931260b68b9fa3273 ]

    Current dwmac4_flow_ctrl will not clear
    GMAC_RX_FLOW_CTRL_RFE/GMAC_RX_FLOW_CTRL_RFE bits,
    so MAC hw will keep flow control on although expecting
    flow control off by ethtool. Add codes to fix it.

    Fixes: 477286b53f55 ("stmmac: add GMAC4 core support")
    Signed-off-by: Biao Huang <biao.huang@mediatek.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f5d06ac24d9689a0e22a529f86287361a65d673b
Author: Jae Hyun Yoo <jae.hyun.yoo@linux.intel.com>
Date:   Fri May 31 18:15:39 2019 -0400

    media: aspeed: fix a kernel warning on clk control

    [ Upstream commit 9698ed4d4a2993ce54b9f7d71a2891e972caa117 ]

    Video engine clock control can be double disabled and eventually
    it causes a kernel warning with stack dump printing out like below:

    [  515.540498] ------------[ cut here ]------------
    [  515.545174] WARNING: CPU: 0 PID: 1310 at drivers/clk/clk.c:684
clk_core_unprepare+0x13c/0x170
    [  515.553806] vclk-gate already unprepared
    [  515.557841] CPU: 0 PID: 1310 Comm: obmc-ikvm Tainted: G        W         5.0.6-
df66fbc97853fbba90a0bfa44de32f3d5f7602b4 #1
    [  515.568973] Hardware name: Generic DT based system
    [  515.573777] Backtrace:
    [  515.576272] [<80107cdc>] (dump_backtrace) from [<80107f10>] (show_stack+0x20/0x24)
    [  515.583930]  r7:803a5614 r6:00000009 r5:00000000 r4:9d88fe1c
```

```
     [  515.589712] [<80107ef0>] (show_stack) from [<80690184>] (dump_stack+0x20/0x28)
     [  515.597053] [<80690164>] (dump_stack) from [<80116044>] (__warn.part.3+0xb4/0xdc)
     [  515.604557] [<80115f90>] (__warn.part.3) from [<801160d8>]
 (warn_slowpath_fmt+0x6c/0x90)
     [  515.612734]  r6:000002ac r5:8080befc r4:80a07008
     [  515.617463] [<80116070>] (warn_slowpath_fmt) from [<803a5614>]
 (clk_core_unprepare+0x13c/0x170)
     [  515.626167]  r3:8080cdf4 r2:8080bfc0
     [  515.629834]  r7:98d682a8 r6:9d8a9200 r5:9e5151a0 r4:97abd620
     [  515.635530] [<803a54d8>] (clk_core_unprepare) from [<803a76a4>]
 (clk_unprepare+0x34/0x3c)
     [  515.643812]  r5:9e5151a0 r4:97abd620
     [  515.647529] [<803a7670>] (clk_unprepare) from [<804f36ec>]
 (aspeed_video_off+0x38/0x50)
     [  515.655539]  r5:9e5151a0 r4:9e504000
     [  515.659242] [<804f36b4>] (aspeed_video_off) from [<804f4358>]
 (aspeed_video_release+0x90/0x114)
     [  515.668036]  r5:9e5044b0 r4:9e504000
     [  515.671643] [<804f42c8>] (aspeed_video_release) from [<804d302c>]
 (v4l2_release+0xd4/0xe8)
     [  515.679999]  r7:98d682a8 r6:9d087810 r5:9d8a9200 r4:9e504318
     [  515.685695] [<804d2f58>] (v4l2_release) from [<80236454>] (__fput+0x98/0x1c4)
     [  515.692914]  r5:9e51b608 r4:9d8a9200
     [  515.696597] [<802363bc>] (__fput) from [<802365e8>] (____fput+0x18/0x1c)
     [  515.703315]  r9:80a0700c r8:801011e4 r7:00000000 r6:80a64b9c r5:9d8e35a0 r4:9d8e38dc
     [  515.711167] [<802365d0>] (____fput) from [<80131ca4>] (task_work_run+0x7c/0xa0)
     [  515.718596] [<80131c28>] (task_work_run) from [<80106884>]
 (do_work_pending+0x4a8/0x578)
     [  515.726777]  r7:801011e4 r6:80a07008 r5:9d88ffb0 r4:ffffe000
     [  515.732466] [<801063dc>] (do_work_pending) from [<8010106c>]
 (slow_work_pending+0xc/0x20)
     [  515.740727] Exception stack(0x9d88ffb0 to 0x9d88fff8)
     [  515.745840] ffa0:                                     00000000 76f18094 00000000
 00000000
     [  515.754122] ffc0: 00000007 00176778 7eda4c20 00000006 00000000 00000000 48e20fa4
 00000000
     [  515.762386] ffe0: 00000002 7eda4b08 00000000 48f91efc 80000010 00000007
     [  515.769097]  r10:00000000 r9:9d88e000 r8:801011e4 r7:00000006 r6:7eda4c20 r5:00176778
     [  515.777006]  r4:00000007
     [  515.779558] ---[ end trace 12c04aadef8afbbb ]---
     [  515.784176] ------------[ cut here ]------------
     [  515.788817] WARNING: CPU: 0 PID: 1310 at drivers/clk/clk.c:825
 clk_core_disable+0x18c/0x204
     [  515.797161] eclk-gate already disabled
     [  515.800916] CPU: 0 PID: 1310 Comm: obmc-ikvm Tainted: G        W         5.0.6-
 df66fbc97853fbba90a0bfa44de32f3d5f7602b4 #1
     [  515.811945] Hardware name: Generic DT based system
     [  515.816730] Backtrace:
     [  515.819210] [<80107cdc>] (dump_backtrace) from [<80107f10>] (show_stack+0x20/0x24)
     [  515.826782]  r7:803a5900 r6:00000009 r5:00000000 r4:9d88fe04
     [  515.832454] [<80107ef0>] (show_stack) from [<80690184>] (dump_stack+0x20/0x28)
     [  515.839687] [<80690164>] (dump_stack) from [<80116044>] (__warn.part.3+0xb4/0xdc)
     [  515.847170] [<80115f90>] (__warn.part.3) from [<801160d8>]
 (warn_slowpath_fmt+0x6c/0x90)
     [  515.855247]  r6:00000339 r5:8080befc r4:80a07008
     [  515.859868] [<80116070>] (warn_slowpath_fmt) from [<803a5900>]
 (clk_core_disable+0x18c/0x204)
     [  515.868385]  r3:8080cdd0 r2:8080c00c
     [  515.871957]  r7:98d682a8 r6:9d8a9200 r5:97abd560 r4:97abd560
     [  515.877615] [<803a5774>] (clk_core_disable) from [<803a59a0>]
 (clk_core_disable_lock+0x28/0x34)
     [  515.886301]  r7:98d682a8 r6:9d8a9200 r5:97abd560 r4:a0000013
     [  515.891960] [<803a5978>] (clk_core_disable_lock) from [<803a7714>]
 (clk_disable+0x2c/0x30)
     [  515.900216]  r5:9e5151a0 r4:9e515f60
```

```
      [  515.903816] [<803a76e8>] (clk_disable) from [<804f36f8>] (aspeed_video_off+0x44/0x50)
      [  515.911656] [<804f36b4>] (aspeed_video_off) from [<804f4358>]
  (aspeed_video_release+0x90/0x114)
      [  515.920341]  r5:9e5044b0 r4:9e504000
      [  515.923921] [<804f42c8>] (aspeed_video_release) from [<804d302c>]
  (v4l2_release+0xd4/0xe8)
      [  515.932184]  r7:98d682a8 r6:9d087810 r5:9d8a9200 r4:9e504318
      [  515.937851] [<804d2f58>] (v4l2_release) from [<80236454>] (__fput+0x98/0x1c4)
      [  515.944980]  r5:9e51b608 r4:9d8a9200
      [  515.948559] [<802363bc>] (__fput) from [<802365e8>] (____fput+0x18/0x1c)
      [  515.955257]  r9:80a0700c r8:801011e4 r7:00000000 r6:80a64b9c r5:9d8e35a0 r4:9d8e38dc
      [  515.963008] [<802365d0>] (____fput) from [<80131ca4>] (task_work_run+0x7c/0xa0)
      [  515.970333] [<80131c28>] (task_work_run) from [<80106884>]
  (do_work_pending+0x4a8/0x578)
      [  515.978421]  r7:801011e4 r6:80a07008 r5:9d88ffb0 r4:ffffe000
      [  515.984086] [<801063dc>] (do_work_pending) from [<8010106c>]
  (slow_work_pending+0xc/0x20)
      [  515.992247] Exception stack(0x9d88ffb0 to 0x9d88fff8)
      [  515.997296] ffa0:                                     00000000 76f18094 00000000
  00000000
      [  516.005473] ffc0: 00000007 00176778 7eda4c20 00000006 00000000 00000000 48e20fa4
  00000000
      [  516.013642] ffe0: 00000002 7eda4b08 00000000 48f91efc 80000010 00000007
      [  516.020257]  r10:00000000 r9:9d88e000 r8:801011e4 r7:00000006 r6:7eda4c20 r5:00176778
      [  516.028072]  r4:00000007
      [  516.030606] ---[ end trace 12c04aadef8afbbc ]---

      To prevent this issue, this commit adds clock status checking
      logic into the Aspeed video engine driver.

      Signed-off-by: Jae Hyun Yoo <jae.hyun.yoo@linux.intel.com>
      Reviewed-by: Eddie James <eajames@linux.ibm.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 33f5c9308a554c8fcda6e7cc5fd6a2b7e3e65370
  Author: Jae Hyun Yoo <jae.hyun.yoo@linux.intel.com>
  Date:   Fri May 31 18:15:41 2019 -0400

      media: aspeed: change irq to threaded irq

      [ Upstream commit 12ae1c1bf5db2f33fcd9092a96f630291c4b181a ]

      Differently from other Aspeed drivers, this driver calls clock
      control APIs in interrupt context. Since ECLK is coupled with a
      reset bit in clk-aspeed module, aspeed_clk_enable will make 10ms of
      busy waiting delay for triggering the reset and it will eventually
      disturb other drivers' interrupt handling. To fix this issue, this
      commit changes this driver's irq to threaded irq so that the delay
      can be happened in a thread context.

      Signed-off-by: Jae Hyun Yoo <jae.hyun.yoo@linux.intel.com>
      Reviewed-by: Eddie James <eajames@linux.ibm.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 cf16cff4c04901f9a9370300c3ba3f0437c44860
  Author: Jiri Olsa <jolsa@redhat.com>
  Date:   Fri May 31 15:13:21 2019 +0200

      perf jvmti: Address gcc string overflow warning for strncpy()

      [ Upstream commit 279ab04dbea1370d2eac0f854270369ccaef8a44 ]
```

```
    We are getting false positive gcc warning when we compile with gcc9 (9.1.1):

         CC       jvmti/libjvmti.o
       In file included from /usr/include/string.h:494,
                        from jvmti/libjvmti.c:5:
       In function ‘strncpy’,
           inlined from ‘copy_class_filename.constprop’ at jvmti/libjvmti.c:166:3:
       /usr/include/bits/string_fortified.h:106:10: error: ‘__builtin_strncpy’ specified
 bound depends on the length of the source argument [-Werror=stringop-overflow=]
         106 |   return __builtin___strncpy_chk (__dest, __src, __len, __bos (__dest));
             |          ^~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
       jvmti/libjvmti.c: In function ‘copy_class_filename.constprop’:
       jvmti/libjvmti.c:165:26: note: length computed here
         165 |   size_t file_name_len = strlen(file_name);
             |                          ^~~~~~~~~~~~~~~~~
       cc1: all warnings being treated as errors

    As per Arnaldo's suggestion use strlcpy(), which does the same thing and keeps
    gcc silent.

    Suggested-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Signed-off-by: Jiri Olsa <jolsa@kernel.org>
    Cc: Alexander Shishkin <alexander.shishkin@linux.intel.com>
    Cc: Ben Gainey <ben.gainey@arm.com>
    Cc: Namhyung Kim <namhyung@kernel.org>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Cc: Stephane Eranian <eranian@google.com>
    Link: http://lkml.kernel.org/r/20190531131321.GB1281@krava
    Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit bdfd587ab363a3eabc29d082a1d027e44462b92f
Author: Fabio Estevam <festevam@gmail.com>
Date:   Fri May 31 13:45:04 2019 -0400

    media: imx7-mipi-csis: Propagate the error if clock enabling fails

    [ Upstream commit 2b393f91c651c16d5c09f5c7aa689e58a79df34e ]

    Currently the return value from clk_bulk_prepare_enable() is checked,
    but it is not propagate it in the case of failure.

    Fix it and also move the error message to the caller of
    mipi_csis_clk_enable().

    Signed-off-by: Fabio Estevam <festevam@gmail.com>
    Reviewed-by: Rui Miguel Silva <rmfrfs@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 694bc18a64c47849deb49e01e33123af07cadc33
Author: Miles Chen <miles.chen@mediatek.com>
Date:   Wed May 29 00:08:20 2019 +0800

    arm64: mm: make CONFIG_ZONE_DMA32 configurable

    [ Upstream commit 0c1f14ed12262f45a3af1d588e4d7bd12438b8f5 ]

    This change makes CONFIG_ZONE_DMA32 defuly y and allows users
    to overwrite it only when CONFIG_EXPERT=y.

    For the SoCs that do not need CONFIG_ZONE_DMA32, this is the
    first step to manage all available memory by a single
```

```
    zone(normal zone) to reduce the overhead of multiple zones.

    The change also fixes a build error when CONFIG_NUMA=y and
    CONFIG_ZONE_DMA32=n.

    arch/arm64/mm/init.c:195:17: error: use of undeclared identifier 'ZONE_DMA32'
                    max_zone_pfns[ZONE_DMA32] = PFN_DOWN(max_zone_dma_phys());

    Change since v1:
    1. only expose CONFIG_ZONE_DMA32 when CONFIG_EXPERT=y
    2. remove redundant IS_ENABLED(CONFIG_ZONE_DMA32)

    Cc: Robin Murphy <robin.murphy@arm.com>
    Signed-off-by: Miles Chen <miles.chen@mediatek.com>
    Signed-off-by: Catalin Marinas <catalin.marinas@arm.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4d5a27a47a4e8a00689cbab2fdf5bd10a932bf06
Author: Abhishek Goel <huntbag@linux.vnet.ibm.com>
Date:   Wed May 29 04:30:33 2019 -0500

    cpupower : frequency-set -r option misses the last cpu in related cpu list

    [ Upstream commit 04507c0a9385cc8280f794a36bfff567c8cc1042 ]

    To set frequency on specific cpus using cpupower, following syntax can
    be used :
    cpupower -c #i frequency-set -f #f -r

    While setting frequency using cpupower frequency-set command, if we use
    '-r' option, it is expected to set frequency for all cpus related to
    cpu #i. But it is observed to be missing the last cpu in related cpu
    list. This patch fixes the problem.

    Signed-off-by: Abhishek Goel <huntbag@linux.vnet.ibm.com>
    Reviewed-by: Thomas Renninger <trenn@suse.de>
    Signed-off-by: Shuah Khan <skhan@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 852b5e1fa70868b5fc88be7299fecba274a1df09
Author: Weihang Li <liweihang@hisilicon.com>
Date:   Mon Jun 3 10:09:18 2019 +0800

    net: hns3: set ops to null when unregister ad_dev

    [ Upstream commit 594a81b39525f0a17e92c2e0b167ae1400650380 ]

    The hclge/hclgevf and hns3 module can be unloaded independently,
    when hclge/hclgevf unloaded firstly, the ops of ae_dev should
    be set to NULL, otherwise it will cause an use-after-free problem.

    Fixes: 38caee9d3ee8 ("net: hns3: Add support of the HNAE3 framework")
    Signed-off-by: Weihang Li <liweihang@hisilicon.com>
    Signed-off-by: Peng Li <lipeng321@huawei.com>
    Signed-off-by: Huazhong Tan <tanhuazhong@huawei.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 44f7be43d1399251f3c9ad00582364cdae2c7be0
Author: Weihang Li <liweihang@hisilicon.com>
Date:   Mon Jun 3 10:09:17 2019 +0800

    net: hns3: add a check to pointer in error_detected and slot_reset

    [ Upstream commit 661262bc3e0ecc9a1aed39c6b2a99766da2c22e2 ]
```

```
    If we add a VF without loading hclgevf.ko and then there is a RAS error
    occurs, PCIe AER will call error_detected and slot_reset of all functions,
    and will get a NULL pointer when we check ad_dev->ops->handle_hw_ras_error.
    This will cause a call trace and failures on handling of follow-up RAS
    errors.

    This patch check ae_dev and ad_dev->ops at first to solve above issues.

    Signed-off-by: Weihang Li <liweihang@hisilicon.com>
    Signed-off-by: Peng Li <lipeng321@huawei.com>
    Signed-off-by: Huazhong Tan <tanhuazhong@huawei.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8d9765e899b4e527f23d43a0abfb997db0f85f82
Author: Kefeng Wang <wangkefeng.wang@huawei.com>
Date:   Thu May 30 03:25:49 2019 -0400

    media: wl128x: Fix some error handling in fm_v4l2_init_video_device()

    [ Upstream commit 69fbb3f47327d959830c94bf31893972b8c8f700 ]

    X-Originating-IP: [10.175.113.25]
    X-CFilter-Loop: Reflected
    The fm_v4l2_init_video_device() forget to unregister v4l2/video device
    in the error path, it could lead to UAF issue, eg,

      BUG: KASAN: use-after-free in atomic64_read include/asm-generic/atomic-
instrumented.h:836 [inline]
      BUG: KASAN: use-after-free in atomic_long_read include/asm-generic/atomic-long.h:28
[inline]
      BUG: KASAN: use-after-free in __mutex_unlock_slowpath+0x92/0x690
kernel/locking/mutex.c:1206
      Read of size 8 at addr ffff8881e84a7c70 by task v4l_id/3659

      CPU: 1 PID: 3659 Comm: v4l_id Not tainted 5.1.0 #8
      Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS 1.10.2-1ubuntu1 04/01/2014
      Call Trace:
       __dump_stack lib/dump_stack.c:77 [inline]
       dump_stack+0xa9/0x10e lib/dump_stack.c:113
       print_address_description+0x65/0x270 mm/kasan/report.c:187
       kasan_report+0x149/0x18d mm/kasan/report.c:317
       atomic64_read include/asm-generic/atomic-instrumented.h:836 [inline]
       atomic_long_read include/asm-generic/atomic-long.h:28 [inline]
       __mutex_unlock_slowpath+0x92/0x690 kernel/locking/mutex.c:1206
       fm_v4l2_fops_open+0xac/0x120 [fm_drv]
       v4l2_open+0x191/0x390 [videodev]
       chrdev_open+0x20d/0x570 fs/char_dev.c:417
       do_dentry_open+0x700/0xf30 fs/open.c:777
       do_last fs/namei.c:3416 [inline]
       path_openat+0x7c4/0x2a90 fs/namei.c:3532
       do_filp_open+0x1a5/0x2b0 fs/namei.c:3563
       do_sys_open+0x302/0x490 fs/open.c:1069
       do_syscall_64+0x9f/0x450 arch/x86/entry/common.c:290
       entry_SYSCALL_64_after_hwframe+0x49/0xbe
      RIP: 0033:0x7f8180c17c8e
      ...
      Allocated by task 3642:
       set_track mm/kasan/common.c:87 [inline]
       __kasan_kmalloc.constprop.3+0xa0/0xd0 mm/kasan/common.c:497
       fm_drv_init+0x13/0x1000 [fm_drv]
       do_one_initcall+0xbc/0x47d init/main.c:901
       do_init_module+0x1b5/0x547 kernel/module.c:3456
       load_module+0x6405/0x8c10 kernel/module.c:3804
```

```
        __do_sys_finit_module+0x162/0x190 kernel/module.c:3898
        do_syscall_64+0x9f/0x450 arch/x86/entry/common.c:290
        entry_SYSCALL_64_after_hwframe+0x49/0xbe

       Freed by task 3642:
        set_track mm/kasan/common.c:87 [inline]
        __kasan_slab_free+0x130/0x180 mm/kasan/common.c:459
        slab_free_hook mm/slub.c:1429 [inline]
        slab_free_freelist_hook mm/slub.c:1456 [inline]
        slab_free mm/slub.c:3003 [inline]
        kfree+0xe1/0x270 mm/slub.c:3958
        fm_drv_init+0x1e6/0x1000 [fm_drv]
        do_one_initcall+0xbc/0x47d init/main.c:901
        do_init_module+0x1b5/0x547 kernel/module.c:3456
        load_module+0x6405/0x8c10 kernel/module.c:3804
        __do_sys_finit_module+0x162/0x190 kernel/module.c:3898
        do_syscall_64+0x9f/0x450 arch/x86/entry/common.c:290
        entry_SYSCALL_64_after_hwframe+0x49/0xbe

    Add relevant unregister functions to fix it.

    Cc: Hans Verkuil <hans.verkuil@cisco.com>
    Reported-by: Hulk Robot <hulkci@huawei.com>
    Signed-off-by: Kefeng Wang <wangkefeng.wang@huawei.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e8a4563973a49ff82280aaeb13f88f4b4c89ac3e
Author: Neil Armstrong <narmstrong@baylibre.com>
Date:   Fri May 31 06:33:15 2019 -0400

    media: platform: ao-cec-g12a: disable regmap fast_io for cec bus regmap

    [ Upstream commit 9f7406d6b56b4b71a12480b68221755ea7b3e0ee ]

    With fast_io enabled, spinlock_irq is used for read/write operations,
    thus leading to :
    BUG: sleeping function called from invalid context at [snip]/ao-cec-g12a.c:379
     in_atomic(): 1, irqs_disabled(): 128, pid: 1451, name: irq/14-ff800280
    [snip]
    Call trace:
     dump_backtrace+0x0/0x180
     show_stack+0x14/0x1c
     dump_stack+0xa8/0xe0
     ___might_sleep+0xf4/0x104
     __might_sleep+0x4c/0x80
     meson_ao_cec_g12a_read+0x7c/0x164
     regmap_read+0x16c/0x1b0
     meson_ao_cec_g12a_irq_thread+0xcc/0x200
     irq_thread_fn+0x2c/0x60
     irq_thread+0x14c/0x1fc
     kthread+0x11c/0x12c
     ret_from_fork+0x10/0x18

    Simply remove fast_io to use mutexes instead.

    Fixes: b7778c46683c ("media: platform: meson: Add Amlogic Meson G12A AO CEC Controller
driver")

    Signed-off-by: Neil Armstrong <narmstrong@baylibre.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 e26b15def40a6f93cc29738c22e33f2d6fdda330
```

```
Author: Imre Deak <imre.deak@intel.com>
Date:   Fri May 24 23:15:09 2019 +0300

    locking/lockdep: Fix merging of hlocks with non-zero references

    [ Upstream commit d9349850e188b8b59e5322fda17ff389a1c0cd7d ]

    The sequence

            static DEFINE_WW_CLASS(test_ww_class);

            struct ww_acquire_ctx ww_ctx;
            struct ww_mutex ww_lock_a;
            struct ww_mutex ww_lock_b;
            struct ww_mutex ww_lock_c;
            struct mutex lock_c;

            ww_acquire_init(&ww_ctx, &test_ww_class);

            ww_mutex_init(&ww_lock_a, &test_ww_class);
            ww_mutex_init(&ww_lock_b, &test_ww_class);
            ww_mutex_init(&ww_lock_c, &test_ww_class);

            mutex_init(&lock_c);

            ww_mutex_lock(&ww_lock_a, &ww_ctx);

            mutex_lock(&lock_c);

            ww_mutex_lock(&ww_lock_b, &ww_ctx);
            ww_mutex_lock(&ww_lock_c, &ww_ctx);

            mutex_unlock(&lock_c);  (*)

            ww_mutex_unlock(&ww_lock_c);
            ww_mutex_unlock(&ww_lock_b);
            ww_mutex_unlock(&ww_lock_a);

            ww_acquire_fini(&ww_ctx); (**)

    will trigger the following error in __lock_release() when calling
    mutex_release() at **:

            DEBUG_LOCKS_WARN_ON(depth <= 0)

    The problem is that the hlock merging happening at * updates the
    references for test_ww_class incorrectly to 3 whereas it should've
    updated it to 4 (representing all the instances for ww_ctx and
    ww_lock_[abc]).

    Fix this by updating the references during merging correctly taking into
    account that we can have non-zero references (both for the hlock that we
    merge into another hlock or for the hlock we are merging into).

    Signed-off-by: Imre Deak <imre.deak@intel.com>
    Signed-off-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Cc: =?UTF-8?q?Ville=20Syrj=C3=A4l=C3=A4?= <ville.syrjala@linux.intel.com>
    Cc: Linus Torvalds <torvalds@linux-foundation.org>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Cc: Thomas Gleixner <tglx@linutronix.de>
    Cc: Will Deacon <will.deacon@arm.com>
    Link: https://lkml.kernel.org/r/20190524201509.9199-2-imre.deak@intel.com
    Signed-off-by: Ingo Molnar <mingo@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 74e248d9f983288cca74f02e132827d9d3ee56dc
Author: Imre Deak <imre.deak@intel.com>
Date:   Fri May 24 23:15:08 2019 +0300

    locking/lockdep: Fix OOO unlock when hlocks need merging

    [ Upstream commit 8c8889d8eaf4501ae4aaf870b6f8f55db5d5109a ]

    The sequence

            static DEFINE_WW_CLASS(test_ww_class);

            struct ww_acquire_ctx ww_ctx;
            struct ww_mutex ww_lock_a;
            struct ww_mutex ww_lock_b;
            struct mutex lock_c;
            struct mutex lock_d;

            ww_acquire_init(&ww_ctx, &test_ww_class);

            ww_mutex_init(&ww_lock_a, &test_ww_class);
            ww_mutex_init(&ww_lock_b, &test_ww_class);

            mutex_init(&lock_c);

            ww_mutex_lock(&ww_lock_a, &ww_ctx);

            mutex_lock(&lock_c);

            ww_mutex_lock(&ww_lock_b, &ww_ctx);

            mutex_unlock(&lock_c);          (*)

            ww_mutex_unlock(&ww_lock_b);
            ww_mutex_unlock(&ww_lock_a);

            ww_acquire_fini(&ww_ctx);

    triggers the following WARN in __lock_release() when doing the unlock at *:

            DEBUG_LOCKS_WARN_ON(curr->lockdep_depth != depth - 1);

    The problem is that the WARN check doesn't take into account the merging
    of ww_lock_a and ww_lock_b which results in decreasing curr->lockdep_depth
    by 2 not only 1.

    Note that the following sequence doesn't trigger the WARN, since there
    won't be any hlock merging.

            ww_acquire_init(&ww_ctx, &test_ww_class);

            ww_mutex_init(&ww_lock_a, &test_ww_class);
            ww_mutex_init(&ww_lock_b, &test_ww_class);

            mutex_init(&lock_c);
            mutex_init(&lock_d);

            ww_mutex_lock(&ww_lock_a, &ww_ctx);

            mutex_lock(&lock_c);
            mutex_lock(&lock_d);

            ww_mutex_lock(&ww_lock_b, &ww_ctx);

            mutex_unlock(&lock_d);
```

```
            ww_mutex_unlock(&ww_lock_b);
            ww_mutex_unlock(&ww_lock_a);

            mutex_unlock(&lock_c);

            ww_acquire_fini(&ww_ctx);
```

    In general both of the above two sequences are valid and shouldn't
    trigger any lockdep warning.

    Fix this by taking the decrement due to the hlock merging into account
    during lock release and hlock class re-setting. Merging can't happen
    during lock downgrading since there won't be a new possibility to merge
    hlocks in that case, so add a WARN if merging still happens then.

    Signed-off-by: Imre Deak <imre.deak@intel.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: Will Deacon <will.deacon@arm.com>
    Cc: ville.syrjala@linux.intel.com
    Link: https://lkml.kernel.org/r/20190524201509.9199-1-imre.deak@intel.com
    Signed-off-by: Ingo Molnar <mingo@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a6091a0b64e3e6225b6eb32122f3b0d658ac0f64
Author: Sven Eckelmann <sven@narfation.org>
Date: Sun Jun 2 10:57:31 2019 +0200

    batman-adv: Fix duplicated OGMs on NETDEV_UP

    [Upstream commit 9e6b5648bbc4cd48fab62cecbb81e9cc3c6e7e88]

    The state of slave interfaces are handled differently depending on whether
    the interface is up or not. All active interfaces (IFF_UP) will transmit
    OGMs. But for B.A.T.M.A.N. IV, also non-active interfaces are scheduling
    (low TTL) OGMs on active interfaces. The code which setups and schedules
    the OGMs must therefore already be called when the interfaces gets added as
    slave interface and the transmit function must then check whether it has to
    send out the OGM or not on the specific slave interface.

    But the commit f0d97253fb5f ("batman-adv: remove ogm_emit and ogm_schedule
    API calls") moved the setup code from the enable function to the activate
    function. The latter is called either when the added slave was already up
    when batadv_hardif_enable_interface processed the new interface or when a
    NETDEV_UP event was received for this slave interfac. As result, each
    NETDEV_UP would schedule a new OGM worker for the interface and thus OGMs
    would be send a lot more than expected.

    Fixes: f0d97253fb5f ("batman-adv: remove ogm_emit and ogm_schedule API calls")
    Reported-by: Linus Lüssing <linus.luessing@c0d3.blue>
    Tested-by: Linus Lüssing <linus.luessing@c0d3.blue>
    Acked-by: Marek Lindner <mareklindner@neomailbox.ch>
    Signed-off-by: Sven Eckelmann <sven@narfation.org>
    Signed-off-by: Simon Wunderlich <sw@simonwunderlich.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1390dacc3681d213546e4f5cbe18aa4572fad900
Author: David S. Miller <davem@davemloft.net>
Date: Thu May 30 11:36:15 2019 -0700

    tua6100: Avoid build warnings.

```
    [ Upstream commit 621ccc6cc5f8d6730b740d31d4818227866c93c9 ]

    Rename _P to _P_VAL and _R to _R_VAL to avoid global
    namespace conflicts:

    drivers/media/dvb-frontends/tua6100.c: In function ‘tua6100_set_params’:
    drivers/media/dvb-frontends/tua6100.c:79: warning: "_P" redefined
     #define _P 32

    In file included from ./include/acpi/platform/aclinux.h:54,
                     from ./include/acpi/platform/acenv.h:152,
                     from ./include/acpi/acpi.h:22,
                     from ./include/linux/acpi.h:34,
                     from ./include/linux/i2c.h:17,
                     from drivers/media/dvb-frontends/tua6100.h:30,
                     from drivers/media/dvb-frontends/tua6100.c:32:
    ./include/linux/ctype.h:14: note: this is the location of the previous definition
     #define _P 0x10 /* punct */

    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 52265017e39e97df962a1721d47ad494f1357c9f
Author: Christophe Leroy <christophe.leroy@c-s.fr>
Date:   Tue May 21 13:34:18 2019 +0000

    crypto: talitos - Align SEC1 accesses to 32 bits boundaries.

    [ Upstream commit c9cca7034b34a2d82e9a03b757de2485c294851c ]

    The MPC885 reference manual states:

    SEC Lite-initiated 8xx writes can occur only on 32-bit-word boundaries, but
    reads can occur on any byte boundary. Writing back a header read from a
    non-32-bit-word boundary will yield unpredictable results.

    In order to ensure that, cra_alignmask is set to 3 for SEC1.

    Signed-off-by: Christophe Leroy <christophe.leroy@c-s.fr>
    Fixes: 9c4a79653b35 ("crypto: talitos - Freescale integrated security engine (SEC)
driver")
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ed6e14dcb82028a292353ee6fbc659344662f4e8
Author: Christophe Leroy <christophe.leroy@c-s.fr>
Date:   Tue May 21 13:34:17 2019 +0000

    crypto: talitos - properly handle split ICV.

    [ Upstream commit eae55a586c3c8b50982bad3c3426e9c9dd7a0075 ]

    The driver assumes that the ICV is as a single piece in the last
    element of the scatterlist. This assumption is wrong.

    This patch ensures that the ICV is properly handled regardless of
    the scatterlist layout.

    Fixes: 9c4a79653b35 ("crypto: talitos - Freescale integrated security engine (SEC)
driver")
    Signed-off-by: Christophe Leroy <christophe.leroy@c-s.fr>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d9918f4c0e54f9aae2be661fef12302b2655cbb3
```

```
Author: Vladimir Oltean <olteanv@gmail.com>
Date:   Tue May 28 20:38:17 2019 +0300

    net: dsa: sja1105: Fix broken fixed-link interfaces on user ports

    [ Upstream commit af7cd0366ee994e8b35985d407261dc0ed9dfb4d ]

    PHYLIB and PHYLINK handle fixed-link interfaces differently. PHYLIB
    wraps them in a software PHY ("pseudo fixed link") phydev construct such
    that .adjust_link driver callbacks see an unified API. Whereas PHYLINK
    simply creates a phylink_link_state structure and passes it to
    .mac_config.

    At the time the driver was introduced, DSA was using PHYLIB for the
    CPU/cascade ports (the ones with no net devices) and PHYLINK for
    everything else.

    As explained below:

    commit aab9c4067d2389d0adfc9c53806437df7b0fe3d5
    Author: Florian Fainelli <f.fainelli@gmail.com>
    Date:   Thu May 10 13:17:36 2018 -0700

      net: dsa: Plug in PHYLINK support

      Drivers that utilize fixed links for user-facing ports (e.g: bcm_sf2)
      will need to implement phylink_mac_ops from now on to preserve
      functionality, since PHYLINK *does not* create a phy_device instance
      for fixed links.

    In the above patch, DSA guards the .phylink_mac_config callback against
    a NULL phydev pointer.  Therefore, .adjust_link is not called in case of
    a fixed-link user port.

    This patch fixes the situation by converting the driver from using
    .adjust_link to .phylink_mac_config.  This can be done now in a unified
    fashion for both slave and CPU/cascade ports because DSA now uses
    PHYLINK for all ports.

    Signed-off-by: Vladimir Oltean <olteanv@gmail.com>
    Signed-off-by: Ioana Ciornei <ioana.ciornei@nxp.com>
    Reviewed-by: Florian Fainelli <f.fainelli@gmail.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a977ec50691d6ad12fc9eadd18e3d7f2ed0812f1
Author: Ioana Ciornei <ioana.ciornei@nxp.com>
Date:   Tue May 28 20:38:09 2019 +0300

    net: phy: Check against net_device being NULL

    [ Upstream commit 82c76aca81187b3d28a6fb3062f6916450ce955e ]

    In general, we don't want MAC drivers calling phy_attach_direct with the
    net_device being NULL. Add checks against this in all the functions
    calling it: phy_attach() and phy_connect_direct().

    Signed-off-by: Ioana Ciornei <ioana.ciornei@nxp.com>
    Suggested-by: Andrew Lunn <andrew@lunn.ch>
    Reviewed-by: Andrew Lunn <andrew@lunn.ch>
    Reviewed-by: Florian Fainelli <f.fainelli@gmail.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ba3a612f8559dee674657dee2870405f6f1f6973
```

```
Author: Shailendra Verma <shailendra.v@samsung.com>
Date:   Thu Nov 24 23:57:34 2016 -0500

    media: staging: media: davinci_vpfe: - Fix for memory leak if decoder initialization
fails.

    [ Upstream commit 6995a659101bd4effa41cebb067f9dc18d77520d ]

    Fix to avoid possible memory leak if the decoder initialization
    got failed.Free the allocated memory for file handle object
    before return in case decoder initialization fails.

    Signed-off-by: Shailendra Verma <shailendra.v@samsung.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 175955ac858c396b2a8ec3da4ca20677da900560
Author: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com>
Date:   Tue May 28 11:28:22 2019 -0500

    ASoC: Intel: sof-rt5682: fix undefined references with Baytrail-only support

    [ Upstream commit 17fc24875da1bef4650cf007edae3b2e26d2fa4e ]

    The sof-rt5682 machine driver supports both legacy Baytrail devices
    and more recent ApolloLake/CometLake platforms. When only Baytrail is
    selected, the compilation fails with the following errors:

    ERROR: "hdac_hdmi_jack_port_init"
    [sound/soc/intel/boards/snd-soc-sof_rt5682.ko] undefined!

    ERROR: "hdac_hdmi_jack_init"
    [sound/soc/intel/boards/snd-soc-sof_rt5682.ko] undefined!

    Fix by selecting SND_SOC_HDAC_HDMI unconditionally. The code for HDMI
    support is not reachable on Baytrail so this change has no functional
    impact.

    Fixes: f70abd75b7c6 ("ASoC: Intel: add sof-rt5682 machine driver")
    Reported-by: kbuild test robot <lkp@intel.com>
    Signed-off-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 fea9869ff4c2cc39d2e8e86868d6d47fdf543b36
Author: Kefeng Wang <wangkefeng.wang@huawei.com>
Date:   Mon May 27 08:14:55 2019 -0400

    media: saa7164: fix remove_proc_entry warning

    [ Upstream commit 50710eeefbc1ed25375942aad0c4d1eb4af0f330 ]

    if saa7164_proc_create() fails, saa7164_fini() will trigger a warning,

    name 'saa7164'
    WARNING: CPU: 1 PID: 6311 at fs/proc/generic.c:672 remove_proc_entry+0x1e8/0x3a0
      ? remove_proc_entry+0x1e8/0x3a0
      ? try_stop_module+0x7b/0x240
      ? proc_readdir+0x70/0x70
      ? rcu_read_lock_sched_held+0xd7/0x100
      saa7164_fini+0x13/0x1f [saa7164]
      __x64_sys_delete_module+0x30c/0x480
      ? __ia32_sys_delete_module+0x480/0x480
      ? __x64_sys_clock_gettime+0x11e/0x1c0
      ? __x64_sys_timer_create+0x1a0/0x1a0
```

```
       ? trace_hardirqs_off_caller+0x40/0x180
       ? do_syscall_64+0x18/0x450
       do_syscall_64+0x9f/0x450
       entry_SYSCALL_64_after_hwframe+0x49/0xbe

    Fix it by checking the return of proc_create_single() before
    calling remove_proc_entry().

    Signed-off-by: Kefeng Wang <wangkefeng.wang@huawei.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    [hverkuil-cisco@xs4all.nl: use 0444 instead of S_IRUGO]
    [hverkuil-cisco@xs4all.nl: use pr_info instead of KERN_INFO]
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 717cba2f103a791ff33e7cdfac4841535a9d69e7
Author: Hans Verkuil <hverkuil@xs4all.nl>
Date:   Mon May 27 05:31:13 2019 -0400

    media: mc-device.c: don't memset __user pointer contents

    [ Upstream commit 518fa4e0e0da97ea2e17c95ab57647ce748a96e2 ]

    You can't memset the contents of a __user pointer. Instead, call copy_to_user to
    copy links.reserved (which is zeroed) to the user memory.

    This fixes this sparse warning:

    SPARSE:drivers/media/mc/mc-device.c drivers/media/mc/mc-device.c:521:16:  warning:
incorrect type in argument 1 (different address spaces)

    Fixes: f49308878d720 ("media: media_device_enum_links32: clean a reserved field")

    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Reviewed-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 d2bc45756df51c00ac8a60fab0a5c45bcb665193
Author: Mitch Williams <mitch.a.williams@intel.com>
Date:   Tue Apr 16 10:30:45 2019 -0700

    ice: Check all VFs for MDD activity, don't disable

    [ Upstream commit 23c0112246b454e408fb0579b3f9089353d4d327 ]

    Don't use the mdd_detected variable as an exit condition for this loop;
    the first VF to NOT have an MDD event will cause the loop to terminate.

    Instead just look at all of the VFs, but don't disable them. This
    prevents proper release of resources if the VFs are rebooted or the VF
    driver reloaded. Instead, just log a message and call out repeat
    offenders.

    To make it clear what we are doing, use a differently-named variable in
    the loop.

    Signed-off-by: Mitch Williams <mitch.a.williams@intel.com>
    Signed-off-by: Anirudh Venkataramanan <anirudh.venkataramanan@intel.com>
    Tested-by: Andrew Bowers <andrewx.bowers@intel.com>
    Signed-off-by: Jeff Kirsher <jeffrey.t.kirsher@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ce217d5f6dc4f9de56d4b5e95d68d121490b9b36
Author: Arnaldo Carvalho de Melo <acme@redhat.com>
```

```
Date:   Tue May 28 16:02:56 2019 -0300

    perf annotate TUI browser: Do not use member from variable within its own initialization

    [ Upstream commit da2019633f0b5c105ce658aada333422d8cb28fe ]

    Some compilers will complain when using a member of a struct to
    initialize another member, in the same struct initialization.

    For instance:

      debian:8      Debian clang version 3.5.0-10 (tags/RELEASE_350/final) (based on LLVM
3.5.0)
      oraclelinux:7 clang version 3.4.2 (tags/RELEASE_34/dot2-final)

    Produce:

      ui/browsers/annotate.c:104:12: error: variable 'ops' is uninitialized when used within
its own initialization [-Werror,-Wuninitialized]
                                                 (!ops.current_entry ||
                                                   ^~~
      1 error generated.

    So use an extra variable, initialized just before that struct, to have
    the value used in the expressions used to init two of the struct
    members.

    Cc: Adrian Hunter <adrian.hunter@intel.com>
    Cc: Jiri Olsa <jolsa@kernel.org>
    Cc: Namhyung Kim <namhyung@kernel.org>
    Fixes: c298304bd747 ("perf annotate: Use a ops table for annotation_line__write()")
    Link: https://lkml.kernel.org/n/tip-f9nexro58q62l3o9hez8hr0i@git.kernel.org
    Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 702fc0f88dc5d6124594abafc678d7c3d6022863
Author: Vandana BN <bnvandana@gmail.com>
Date:   Wed May 22 04:34:15 2019 -0400

    media: usb:zr364xx:Fix KASAN:null-ptr-deref Read in zr364xx_vidioc_querycap

    [ Upstream commit 5d2e73a5f80a5b5aff3caf1ec6d39b5b3f54b26e ]

    SyzKaller hit the null pointer deref while reading from uninitialized
    udev->product in zr364xx_vidioc_querycap().

    ==================================================================
    BUG: KASAN: null-ptr-deref in read_word_at_a_time+0xe/0x20
    include/linux/compiler.h:274
    Read of size 1 at addr 0000000000000000 by task v4l_id/5287

    CPU: 1 PID: 5287 Comm: v4l_id Not tainted 5.1.0-rc3-319004-g43151d6 #6
    Hardware name: Google Google Compute Engine/Google Compute Engine, BIOS
    Google 01/01/2011
    Call Trace:
      __dump_stack lib/dump_stack.c:77 [inline]
      dump_stack+0xe8/0x16e lib/dump_stack.c:113
      kasan_report.cold+0x5/0x3c mm/kasan/report.c:321
      read_word_at_a_time+0xe/0x20 include/linux/compiler.h:274
      strscpy+0x8a/0x280 lib/string.c:207
      zr364xx_vidioc_querycap+0xb5/0x210 drivers/media/usb/zr364xx/zr364xx.c:706
      v4l_querycap+0x12b/0x340 drivers/media/v4l2-core/v4l2-ioctl.c:1062
      __video_do_ioctl+0x5bb/0xb40 drivers/media/v4l2-core/v4l2-ioctl.c:2874
      video_usercopy+0x44e/0xf00 drivers/media/v4l2-core/v4l2-ioctl.c:3056
      v4l2_ioctl+0x14e/0x1a0 drivers/media/v4l2-core/v4l2-dev.c:364
```

```
       vfs_ioctl fs/ioctl.c:46 [inline]
       file_ioctl fs/ioctl.c:509 [inline]
       do_vfs_ioctl+0xced/0x12f0 fs/ioctl.c:696
       ksys_ioctl+0xa0/0xc0 fs/ioctl.c:713
       __do_sys_ioctl fs/ioctl.c:720 [inline]
       __se_sys_ioctl fs/ioctl.c:718 [inline]
       __x64_sys_ioctl+0x74/0xb0 fs/ioctl.c:718
       do_syscall_64+0xcf/0x4f0 arch/x86/entry/common.c:290
       entry_SYSCALL_64_after_hwframe+0x49/0xbe
     RIP: 0033:0x7f3b56d8b347
     Code: 90 90 90 48 8b 05 f1 fa 2a 00 64 c7 00 26 00 00 00 48 c7 c0 ff ff ff
     ff c3 90 90 90 90 90 90 90 90 90 b8 10 00 00 00 0f 05 <48> 3d 01 f0 ff
     ff 73 01 c3 48 8b 0d c1 fa 2a 00 31 d2 48 29 c2 64
     RSP: 002b:00007ffe005d5d68 EFLAGS: 00000202 ORIG_RAX: 0000000000000010
     RAX: ffffffffffffffda RBX: 0000000000000003 RCX: 00007f3b56d8b347
     RDX: 00007ffe005d5d70 RSI: 0000000080685600 RDI: 0000000000000003
     RBP: 0000000000000000 R08: 0000000000000000 R09: 0000000000000000
     R10: 0000000000000000 R11: 0000000000000202 R12: 0000000000400884
     R13: 00007ffe005d5ec0 R14: 0000000000000000 R15: 0000000000000000
     ==================================================================
```

    For this device udev->product is not initialized and accessing it causes a NULL pointer
  deref.

    The fix is to check for NULL before strscpy() and copy empty string, if
    product is NULL

    Reported-by: syzbot+66010012fd4c531a1a96@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 ef039427b9a611215ff0566c810bfb7aab0583fd
  Author: Eric Biggers <ebiggers@google.com>
  Date: Mon May 20 09:29:42 2019 -0700

    fscrypt: clean up some BUG_ON()s in block encryption/decryption

    [Upstream commit eeacfdc68a104967162dfcba60f53f6f5b62a334]

    Replace some BUG_ON()s with WARN_ON_ONCE() and returning an error code,
    and move the check for len divisible by FS_CRYPTO_BLOCK_SIZE into
    fscrypt_crypt_block() so that it's done for both encryption and
    decryption, not just encryption.

    Reviewed-by: Chandan Rajendra <chandan@linux.ibm.com>
    Signed-off-by: Eric Biggers <ebiggers@google.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit bb50e4a36b33427fc8f066c227b641abca9452a9
  Author: sumitg <sumitg@nvidia.com>
  Date: Fri May 17 09:53:42 2019 -0400

    media: v4l2-core: fix use-after-free error

    [Upstream commit 3e0f724346e96daae7792262c6767449795ac3b5]

    Fixing use-after-free within __v4l2_ctrl_handler_setup().
    Memory is being freed with kfree(new_ref) for duplicate
    control reference entry but ctrl->cluster pointer is still
    referring to freed duplicate entry resulting in error on
    access. Change done to update cluster pointer only when new
    control reference is added.

```
      ==================================================================
      BUG: KASAN: use-after-free in __v4l2_ctrl_handler_setup+0x388/0x428
      Read of size 8 at addr ffffffc324e78618 by task systemd-udevd/312

      Allocated by task 312:

      Freed by task 312:

      The buggy address belongs to the object at ffffffc324e78600
       which belongs to the cache kmalloc-64 of size 64
      The buggy address is located 24 bytes inside of
       64-byte region [ffffffc324e78600, ffffffc324e78640)
      The buggy address belongs to the page:
      page:fffffffbf0c939e00 count:1 mapcount:0 mapping:
                                             (null) index:0xffffffc324e78f80
      flags: 0x4000000000000100(slab)
      raw: 4000000000000100 0000000000000000 ffffffc324e78f80 000000018020001a
      raw: 0000000000000000 0000000100000001 ffffffc37040fb80 0000000000000000
      page dumped because: kasan: bad access detected

      Memory state around the buggy address:
       ffffffc324e78500: fb fb fb fb fb fb fb fb fc fc fc fc fc fc fc fc
       ffffffc324e78580: fb fb fb fb fb fb fb fb fc fc fc fc fc fc fc fc
      >ffffffc324e78600: fb fb fb fb fb fb fb fb fc fc fc fc fc fc fc fc
                                ^
       ffffffc324e78680: 00 00 00 00 00 00 00 00 fc fc fc fc fc fc fc fc
       ffffffc324e78700: 00 00 00 00 00 fc fc fc fc fc fc fc fc fc fc fc
      ==================================================================

    Suggested-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Sumit Gupta <sumitg@nvidia.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 a1b01aeb1d1e5470f906809df67b71acfee78de6
Author: Kefeng Wang <wangkefeng.wang@huawei.com>
Date:   Mon May 13 03:18:29 2019 -0400

    media: vim2m: fix two double-free issues

    [ Upstream commit 20059cbbf981ca954be56f7963ae494d18e2dda1 ]

    vim2m_device_release() will be called by video_unregister_device() to release
    various objects.

    There are two double-free issue,
    1. dev->m2m_dev will be freed twice in error_m2m path/vim2m_device_release
    2. the error_v4l2 and error_free path in vim2m_probe() will release
       same objects, since vim2m_device_release has done.

    Fixes: ea6c7e34f3b2 ("media: vim2m: replace devm_kzalloc by kzalloc")

    Cc: Laurent Pinchart <laurent.pinchart@ideasonboard.com>
    Reported-by: Hulk Robot <hulkci@huawei.com>
    Signed-off-by: Kefeng Wang <wangkefeng.wang@huawei.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 185563d3bdecf2677cc78f4985b5db5409662c16
Author: Anirudh Gupta <anirudhrudr@gmail.com>
Date:   Tue May 21 20:59:47 2019 +0530

    xfrm: Fix xfrm sel prefix length validation
```

```
    [ Upstream commit b38ff4075a80b4da5cb2202d7965332ca0efb213 ]

    Family of src/dst can be different from family of selector src/dst.
    Use xfrm selector family to validate address prefix length,
    while verifying new sa from userspace.

    Validated patch with this command:
    ip xfrm state add src 1.1.6.1 dst 1.1.6.2 proto esp spi 4260196 \
    reqid 20004 mode tunnel aead "rfc4106(gcm(aes))" \
    0x111101640000000000000000000000044440001 128 \
    sel src 1011:1:4::2/128 sel dst 1021:1:4::2/128 dev Port5

    Fixes: 07bf7908950a ("xfrm: Validate address prefix lengths in the xfrm selector.")
    Signed-off-by: Anirudh Gupta <anirudh.gupta@sophos.com>
    Acked-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Steffen Klassert <steffen.klassert@secunet.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9d047322bb5ff79b7d6b4359670844d15f912b80
Author: Jeremy Sowden <jeremy@azazel.net>
Date:   Sat May 25 19:09:35 2019 +0100

    af_key: fix leaks in key_pol_get_resp and dump_sp.

    [ Upstream commit 7c80eb1c7e2b8420477fbc998971d62a648035d9 ]

    In both functions, if pfkey_xfrm_policy2msg failed we leaked the newly
    allocated sk_buff.  Free it on error.

    Fixes: 55569ce256ce ("Fix conversion between IPSEC_MODE_xxx and XFRM_MODE_xxx.")
    Reported-by: syzbot+4f0529365f7f2208d9f0@syzkaller.appspotmail.com
    Signed-off-by: Jeremy Sowden <jeremy@azazel.net>
    Signed-off-by: Steffen Klassert <steffen.klassert@secunet.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4adddecd4630a132d9800700869327d958efb7a1
Author: Eric W. Biederman <ebiederm@xmission.com>
Date:   Wed May 15 12:29:52 2019 -0500

    signal/pid_namespace: Fix reboot_pid_ns to use send_sig not force_sig

    [ Upstream commit f9070dc94542093fd516ae4ccea17ef46a4362c5 ]

    The locking in force_sig_info is not prepared to deal with a task that
    exits or execs (as sighand may change).  The is not a locking problem
    in force_sig as force_sig is only built to handle synchronous
    exceptions.

    Further the function force_sig_info changes the signal state if the
    signal is ignored, or blocked or if SIGNAL_UNKILLABLE will prevent the
    delivery of the signal.  The signal SIGKILL can not be ignored and can
    not be blocked and SIGNAL_UNKILLABLE won't prevent it from being
    delivered.

    So using force_sig rather than send_sig for SIGKILL is confusing
    and pointless.

    Because it won't impact the sending of the signal and and because
    using force_sig is wrong, replace force_sig with send_sig.

    Cc: Daniel Lezcano <daniel.lezcano@free.fr>
    Cc: Serge Hallyn <serge@hallyn.com>
    Cc: Oleg Nesterov <oleg@redhat.com>
    Fixes: cf3f89214ef6 ("pidns: add reboot_pid_ns() to handle the reboot syscall")
```

```
    Signed-off-by: "Eric W. Biederman" <ebiederm@xmission.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2343820105375bc24ce6ec88827bc3cfe23a92b1
Author: Michal Kalderon <michal.kalderon@marvell.com>
Date:   Sun May 26 15:22:25 2019 +0300

    qed: Set the doorbell address correctly

    [ Upstream commit 8366d520019f366fabd6c7a13032bdcd837e18d4 ]

    In 100g mode the doorbell bar is united for both engines. Set
    the correct offset in the hwfn so that the doorbell returned
    for RoCE is in the affined hwfn.

    Signed-off-by: Ariel Elior <ariel.elior@marvell.com>
    Signed-off-by: Denis Bolotin <denis.bolotin@marvell.com>
    Signed-off-by: Michal Kalderon <michal.kalderon@marvell.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0fe85abd22af3f1675e70fe92f8365c495960315
Author: Jian Shen <shenjian15@huawei.com>
Date:   Fri May 24 19:19:48 2019 +0800

    net: hns3: fix for FEC configuration

    [ Upstream commit f438bfe9d4fe2e491505abfbf04d7c506e00d146 ]

    The FEC capbility may be changed with port speed changes. Driver
    needs to read the active FEC mode, and update FEC capability
    when port speed changes.

    Fixes: 7e6ec9148a1d ("net: hns3: add support for FEC encoding control")
    Signed-off-by: Jian Shen <shenjian15@huawei.com>
    Signed-off-by: Huazhong Tan <tanhuazhong@huawei.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ff081970a4340d6e3348111abbfef8903ec56ae8
Author: Jian Shen <shenjian15@huawei.com>
Date:   Fri May 24 19:19:45 2019 +0800

    net: hns3: initialize CPU reverse mapping

    [ Upstream commit ffab9691bcb2fe2594f4c38bfceb4d9685b93b87 ]

    Allocate CPU rmap and add entry for each irq. CPU rmap is
    used in aRFS to get the queue number of the rx completion
    interrupts.

    In additional, remove the calling of
    irq_set_affinity_notifier() in hns3_nic_init_irq(), because
    we have registered notifier in irq_cpu_rmap_add() for each
    vector, otherwise it may cause use-after-free issue.

    Signed-off-by: Jian Shen <shenjian15@huawei.com>
    Signed-off-by: Huazhong Tan <tanhuazhong@huawei.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 34fb74a62f78cea54941da333c51db4fd10c34af
Author: Maxime Chevallier <maxime.chevallier@bootlin.com>
Date:   Fri May 24 12:05:53 2019 +0200
```

```
    net: mvpp2: cls: Extract the RSS context when parsing the ethtool rule

    [ Upstream commit c561da68038a738f30eca21456534c2d1872d13d ]

    ethtool_rx_flow_rule_create takes into parameter the ethtool flow spec,
    which doesn't contain the rss context id. We therefore need to extract
    it ourself before parsing the ethtool rule.

    The FLOW_RSS flag is only set in info->fs.flow_type, and not
    info->flow_type.

    Signed-off-by: Maxime Chevallier <maxime.chevallier@bootlin.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 43306a82a89dba9ad044087e74d8fc0ac2873522
Author: Brett Creeley <brett.creeley@intel.com>
Date:   Tue Apr 16 10:24:37 2019 -0700

    ice: Fix couple of issues in ice_vsi_release

    [ Upstream commit aa6ccf3f2d7042f94c4e91538956ce7051e7856e ]

    Currently the driver is calling ice_napi_del() and then
    unregister_netdev(). The call to unregister_netdev() will result in a
    call to ice_stop() and then ice_vsi_close(). This is where we call
    napi_disable() for all the MSI-X vectors. This flow is reversed so make
    the changes to ensure napi_disable() happens prior to napi_del().

    Before calling napi_del() and free_netdev() make sure
    unregister_netdev() was called. This is done by making sure the
    __ICE_DOWN bit is set in the vsi->state for the interested VSI.

    Signed-off-by: Brett Creeley <brett.creeley@intel.com>
    Signed-off-by: Anirudh Venkataramanan <anirudh.venkataramanan@intel.com>
    Tested-by: Andrew Bowers <andrewx.bowers@intel.com>
    Signed-off-by: Jeff Kirsher <jeffrey.t.kirsher@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 55635ebe4f6aec64d7359ddf2901f8d138a8993d
Author: Jose Abreu <Jose.Abreu@synopsys.com>
Date:   Fri May 24 10:20:26 2019 +0200

    net: stmmac: Prevent missing interrupts when running NAPI

    [ Upstream commit a976ca79e23f13bff79c14e7266cea4a0ea51e67 ]

    When we trigger NAPI we are disabling interrupts but in case we receive
    or send a packet in the meantime, as interrupts are disabled, we will
    miss this event.

    Trigger both NAPI instances (RX and TX) when at least one event happens
    so that we don't miss any interrupts.

    Signed-off-by: Jose Abreu <joabreu@synopsys.com>
    Cc: Joao Pinto <jpinto@synopsys.com>
    Cc: David S. Miller <davem@davemloft.net>
    Cc: Giuseppe Cavallaro <peppe.cavallaro@st.com>
    Cc: Alexandre Torgue <alexandre.torgue@st.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 70923b730962725a63e673515b2d090d16d6daaa
Author: Jose Abreu <Jose.Abreu@synopsys.com>
Date:   Fri May 24 10:20:25 2019 +0200
```

```
    net: stmmac: dwmac4/5: Clear unused address entries

    [ Upstream commit 0620ec6c62a5a07625b65f699adc5d1b90394ee6 ]

    In case we don't use a given address entry we need to clear it because
    it could contain previous values that are no longer valid.

    Found out while running stmmac selftests.

    Signed-off-by: Jose Abreu <joabreu@synopsys.com>
    Cc: Joao Pinto <jpinto@synopsys.com>
    Cc: David S. Miller <davem@davemloft.net>
    Cc: Giuseppe Cavallaro <peppe.cavallaro@st.com>
    Cc: Alexandre Torgue <alexandre.torgue@st.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e0695dfda6c2a08f35443548b6f6dd1e4a54f312
Author: Jose Abreu <Jose.Abreu@synopsys.com>
Date:   Fri May 24 10:20:21 2019 +0200

    net: stmmac: dwmac1000: Clear unused address entries

    [ Upstream commit 9463c445590091202659cdfdd44b236acadfbd84 ]

    In case we don't use a given address entry we need to clear it because
    it could contain previous values that are no longer valid.

    Found out while running stmmac selftests.

    Signed-off-by: Jose Abreu <joabreu@synopsys.com>
    Cc: Joao Pinto <jpinto@synopsys.com>
    Cc: David S. Miller <davem@davemloft.net>
    Cc: Giuseppe Cavallaro <peppe.cavallaro@st.com>
    Cc: Alexandre Torgue <alexandre.torgue@st.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cb7da2ba5384c411438f715c872c8b9f76fe28bf
Author: Horia Geantă <horia.geanta@nxp.com>
Date:   Fri May 3 17:17:37 2019 +0300

    crypto: caam - avoid S/G table fetching for AEAD zero-length output

    [ Upstream commit dcd9c76e5a183af4f793beb5141efcd260b8d09f ]

    When enabling IOMMU support, the following issue becomes visible
    in the AEAD zero-length case.

    Even though the output sequence length is set to zero, the crypto engine
    tries to prefetch 4 S/G table entries (since SGF bit is set
    in SEQ OUT PTR command - which is either generated in SW in case of
    caam/jr or in HW in case of caam/qi, caam/qi2).
    The DMA read operation will trigger an IOMMU fault since the address in
    the SEQ OUT PTR is "dummy" (set to zero / not obtained via DMA API
    mapping).

    1. In case of caam/jr, avoid the IOMMU fault by clearing the SGF bit
    in SEQ OUT PTR command.

    2. In case of caam/qi - setting address, bpid, length to zero for output
    entry in the compound frame has a special meaning (cf. CAAM RM):
    "Output frame = Unspecified, Input address = Y. A unspecified frame is
    indicated by an unused SGT entry (an entry in which the Address, Length,
```

```
    and BPID fields are all zero). SEC obtains output buffers from BMan as
    prescribed by the preheader."

    Since no output buffers are needed, modify the preheader by setting
    (ABS = 1, ADDBUF = 0):
    -"ABS = 1 means obtain the number of buffers in ADDBUF (0 or 1) from
    the pool POOL ID"
    -ADDBUF: "If ABS is set, ADD BUF specifies whether to allocate
    a buffer or not"

    3. In case of caam/qi2, since engine:
    -does not support FLE[FMT]=2'b11 ("unused" entry) mentioned in DPAA2 RM
    -requires output entry to be present, even if not used
    the solution chosen is to leave output frame list entry zeroized.

    Fixes: 763069ba49d3 ("crypto: caam - handle zero-length AEAD output")
    Signed-off-by: Horia Geantă <horia.geanta@nxp.com>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d58142f19bf28f2be5b02842c780d584847efb64
Author: Wen Yang <wen.yang99@zte.com.cn>
Date:   Mon May 6 03:05:15 2019 -0400

    media: venus: firmware: fix leaked of_node references

    [ Upstream commit 2c41cc0be07b5ee2f1167f41cd8a86fc5b53d82c ]

    The call to of_parse_phandle 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/qcom/venus/firmware.c:90:2-8: ERROR: missing of_node_put;
acquired a node pointer with refcount incremented on line 82, but without a corresponding
object release within this function.
    drivers/media/platform/qcom/venus/firmware.c:94:2-8: ERROR: missing of_node_put;
acquired a node pointer with refcount incremented on line 82, but without a corresponding
object release within this function.
    drivers/media/platform/qcom/venus/firmware.c:128:1-7: ERROR: missing of_node_put;
acquired a node pointer with refcount incremented on line 82, but without a corresponding
object release within this function.

    Signed-off-by: Wen Yang <wen.yang99@zte.com.cn>
    Acked-by: Stanimir Varbanov <stanimir.varbanov@linaro.org>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 42db12d5cd081964e1844dad1f5f4088921fd303
Author: Brett Creeley <brett.creeley@intel.com>
Date:   Tue Apr 16 10:24:32 2019 -0700

    ice: Gracefully handle reset failure in ice_alloc_vfs()

    [ Upstream commit 72f9c2039859e6303550f202d6cc6b8d8af0178c ]

    Currently if ice_reset_all_vfs() fails in ice_alloc_vfs() we fail to
    free some resources, reset variables, and return an error value.
    Fix this by adding another unroll case to free the pf->vf array, set
    the pf->num_alloc_vfs to 0, and return an error code.

    Without this, if ice_reset_all_vfs() fails in ice_alloc_vfs() we will
    not be able to do SRIOV without hard rebooting the system because
    rmmod'ing the driver does not work.
```

```
    Signed-off-by: Brett Creeley <brett.creeley@intel.com>
    Signed-off-by: Anirudh Venkataramanan <anirudh.venkataramanan@intel.com>
    Tested-by: Andrew Bowers <andrewx.bowers@intel.com>
    Signed-off-by: Jeff Kirsher <jeffrey.t.kirsher@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 30b81aca75317a4ab846bccee2645d0602840cd0
Author: Jungo Lin <jungo.lin@mediatek.com>
Date:   Tue Apr 2 21:44:27 2019 -0400

    media: media_device_enum_links32: clean a reserved field

    [ Upstream commit f49308878d7202e07d8761238e01bd0e5fce2750 ]

    In v4l2-compliance utility, test MEDIA_IOC_ENUM_ENTITIES
    will check whether reserved field of media_links_enum filled
    with zero.

    However, for 32 bit program, the reserved field is missing
    copy from kernel space to user space in media_device_enum_links32
    function.

    This patch adds the cleaning a reserved field logic in
    media_device_enum_links32 function.

    Signed-off-by: Jungo Lin <jungo.lin@mediatek.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8f975b16168ed8d7fc8541411f4d6be1d2f512f0
Author: Kangjie Lu <kjlu@umn.edu>
Date:   Fri Mar 22 22:51:06 2019 -0400

    media: vpss: fix a potential NULL pointer dereference

    [ Upstream commit e08f0761234def47961d3252eac09ccedfe4c6a0 ]

    In case ioremap fails, the fix returns -ENOMEM to avoid NULL
    pointer dereference.

    Signed-off-by: Kangjie Lu <kjlu@umn.edu>
    Acked-by: Lad, Prabhakar <prabhakar.csengg@gmail.com>
    Reviewed-by: Mukesh Ojha <mojha@codeaurora.org>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d3e0b8c350e49f46aa36db269e5432a5e7366c8f
Author: Alexei Starovoitov <ast@kernel.org>
Date:   Tue May 21 20:14:20 2019 -0700

    selftests/bpf: adjust verifier scale test

    [ Upstream commit 7c0c6095d48dcd0e67c917aa73cdbb2715aafc36 ]

    Adjust scale tests to check for new jmp sequence limit.

    BPF_JGT had to be changed to BPF_JEQ because the verifier was
    too smart. It tracked the known safe range of R0 values
    and pruned the search earlier before hitting exact 8192 limit.
    bpf_semi_rand_get() was too (un)?lucky.

    k = 0; was missing in bpf_fill_scale2.
    It was testing a bit shorter sequence of jumps than intended.

    Signed-off-by: Alexei Starovoitov <ast@kernel.org>
```

```
    Acked-by: Andrii Nakryiko <andriin@fb.com>
    Signed-off-by: Daniel Borkmann <daniel@iogearbox.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1610f0fbd6ff10a29bdd8e8b95596d0c9d20bc76
Author: Lubomir Rintel <lkundrak@v3.sk>
Date:   Sun May 5 10:00:23 2019 -0400

    media: marvell-ccic: fix DMA s/g desc number calculation

    [ Upstream commit 0c7aa32966dab0b8a7424e1b34c7f206817953ec ]

    The commit d790b7eda953 ("[media] vb2-dma-sg: move dma_(un)map_sg here")
    left dma_desc_nent unset. It previously contained the number of DMA
    descriptors as returned from dma_map_sg().

    We can now (since the commit referred to above) obtain the same value from
    the sg_table and drop dma_desc_nent altogether.

    Tested on OLPC XO-1.75 machine. Doesn't affect the OLPC XO-1's Cafe
    driver, since that one doesn't do DMA.

    [mchehab+samsung@kernel.org: fix a checkpatch warning]

    Fixes: d790b7eda953 ("[media] vb2-dma-sg: move dma_(un)map_sg here")
    Signed-off-by: Lubomir Rintel <lkundrak@v3.sk>
    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 3c975a0373478c7d626a0799262b206ac1d0843b
Author: Akinobu Mita <akinobu.mita@gmail.com>
Date:   Wed Apr 17 10:06:39 2019 -0400

    media: ov7740: avoid invalid framesize setting

    [ Upstream commit 6e4ab830ac6d6a0d7cd7f87dc5d6536369bf24a8 ]

    If the requested framesize by VIDIOC_SUBDEV_S_FMT is larger than supported
    framesizes, it causes an out of bounds array access and the resulting
    framesize is unexpected.

    Avoid out of bounds array access and select the default framesize.

    Cc: Wenyou Yang <wenyou.yang@microchip.com>
    Cc: Eugen Hristev <eugen.hristev@microchip.com>
    Signed-off-by: Akinobu Mita <akinobu.mita@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 589af2ad708b2f3461d00c8a4b93e5e2fd2a5dc9
Author: Christophe Leroy <christophe.leroy@c-s.fr>
Date:   Wed May 15 12:29:03 2019 +0000

    crypto: talitos - fix skcipher failure due to wrong output IV

    [ Upstream commit 3e03e792865ae48b8cfc69a0b4d65f02f467389f ]

    Selftests report the following:

    [    2.984845] alg: skcipher: cbc-aes-talitos encryption test failed (wrong output IV)
on test vector 0, cfg="in-place"
    [    2.995377] 00000000: 3d af ba 42 9d 9e b4 30 b4 22 da 80 2c 9f ac 41
    [    3.032673] alg: skcipher: cbc-des-talitos encryption test failed (wrong output IV)
```

```
 on test vector 0, cfg="in-place"
     [    3.043185] 00000000: fe dc ba 98 76 54 32 10
     [    3.063238] alg: skcipher: cbc-3des-talitos encryption test failed (wrong output IV)
 on test vector 0, cfg="in-place"
     [    3.073818] 00000000: 7d 33 88 93 0f 93 b2 42

     This above dumps show that the actual output IV is indeed the input IV.
     This is due to the IV not being copied back into the request.

     This patch fixes that.

     Signed-off-by: Christophe Leroy <christophe.leroy@c-s.fr>
     Reviewed-by: Horia Geantă <horia.geanta@nxp.com>
     Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit 28eb23321a658e248f0143e5b4d27f1c51e40394
 Author: Daniel Gomez <dagmcr@gmail.com>
 Date:   Mon Apr 22 15:10:20 2019 -0400

     media: spi: IR LED: add missing of table registration

     [ Upstream commit 24e4cf770371df6ad49ed873f21618d9878f64c8 ]

     MODULE_DEVICE_TABLE(of, <of_match_table> should be called to complete DT
     OF mathing mechanism and register it.

     Before this patch:
     modinfo drivers/media/rc/ir-spi.ko  | grep alias

     After this patch:
     modinfo drivers/media/rc/ir-spi.ko  | grep alias
     alias:          of:N*T*Cir-spi-ledC*
     alias:          of:N*T*Cir-spi-led

     Reported-by: Javier Martinez Canillas <javier@dowhile0.org>
     Signed-off-by: Daniel Gomez <dagmcr@gmail.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 3b6503a21e62feb848f88df8cf183ea5be28afa8
 Author: Oliver Neukum <oneukum@suse.com>
 Date:   Tue Apr 30 09:07:36 2019 -0400

     media: dvb: usb: fix use after free in dvb_usb_device_exit

     [ Upstream commit 6cf97230cd5f36b7665099083272595c55d72be7 ]

     dvb_usb_device_exit() frees and uses the device name in that order.
     Fix by storing the name in a buffer before freeing it.

     Signed-off-by: Oliver Neukum <oneukum@suse.com>
     Reported-by: syzbot+26ec41e9f788b3eba396@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 0a9f543d24cc42fb3c5eb5082fa6a53563c8adf6
 Author: Jeremy Sowden <jeremy@azazel.net>
 Date:   Tue May 21 20:58:57 2019 +0100

     batman-adv: fix for leaked TVLV handler.

     [ Upstream commit 17f78dd1bd624a4dd78ed5db3284a63ee807fcc3 ]
```

```
    A handler for BATADV_TVLV_ROAM was being registered when the
    translation-table was initialized, but not unregistered when the
    translation-table was freed.  Unregister it.

    Fixes: 122edaa05940 ("batman-adv: tvlv - convert roaming adv packet to use tvlv unicast
packets")
    Reported-by: syzbot+d454a826e670502484b8@syzkaller.appspotmail.com
    Signed-off-by: Jeremy Sowden <jeremy@azazel.net>
    Signed-off-by: Sven Eckelmann <sven@narfation.org>
    Signed-off-by: Simon Wunderlich <sw@simonwunderlich.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dff8ea22632562d3bb1347679a713c7a4afe69e0
Author: Daniel Baluta <daniel.baluta@nxp.com>
Date:   Fri May 17 13:23:49 2019 +0000

    regmap: debugfs: Fix memory leak in regmap_debugfs_init

    [ Upstream commit 2899872b627e99b7586fe3b6c9f861da1b4d5072 ]

    As detected by kmemleak running on i.MX6ULL board:

    nreferenced object 0xd8366600 (size 64):
      comm "swapper/0", pid 1, jiffies 4294937370 (age 933.220s)
      hex dump (first 32 bytes):
        64 75 6d 6d 79 2d 69 6f 6d 75 78 63 2d 67 70 72  dummy-iomuxc-gpr
        40 32 30 65 34 30 30 30 00 e3 f3 ab fe d1 1b dd  @20e4000........
      backtrace:
        [<b0402aec>] kasprintf+0x2c/0x54
        [<a6fbad2c>] regmap_debugfs_init+0x7c/0x31c
        [<9c8d91fa>] __regmap_init+0xb5c/0xcf4
        [<5b1c3d2a>] of_syscon_register+0x164/0x2c4
        [<596a5d80>] syscon_node_to_regmap+0x64/0x90
        [<49bd597b>] imx6ul_init_machine+0x34/0xa0
        [<250a4dac>] customize_machine+0x1c/0x30
        [<2d19fdaf>] do_one_initcall+0x7c/0x398
        [<e6084469>] kernel_init_freeable+0x328/0x448
        [<168c9101>] kernel_init+0x8/0x114
        [<913268aa>] ret_from_fork+0x14/0x20
        [<ce7b131a>] 0x0

    Root cause is that map->debugfs_name is allocated using kasprintf
    and then the pointer is lost by assigning it other memory address.

    Reported-by: Stefan Wahren <stefan.wahren@i2se.com>
    Signed-off-by: Daniel Baluta <daniel.baluta@nxp.com>
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1708b49f4ef0a647f3cd8cb623942f67f3788b49
Author: Rakesh Pillai <pillair@codeaurora.org>
Date:   Tue Apr 2 18:12:50 2019 +0530

    ath10k: Fix encoding for protected management frames

    [ Upstream commit 42f1bc43e6a97b9ddbe976eba9bd05306c990c75 ]

    Currently the protected management frames are
    not appended with the MIC_LEN which results in
    the protected management frames being encoded
    incorrectly.

    Add the extra space at the end of the protected
    management frames to fix this encoding error for
```

```
    the protected management frames.

    Tested HW: WCN3990
    Tested FW: WLAN.HL.3.1-00784-QCAHLSWMTPLZ-1

    Fixes: 1807da49733e ("ath10k: wmi: add management tx by reference support over wmi")
    Signed-off-by: Rakesh Pillai <pillair@codeaurora.org>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b5190905a4ef823cea7d2f943c8040598f45df34
Author: Anilkumar Kolli <akolli@codeaurora.org>
Date:   Wed Mar 6 23:06:11 2019 +0530

    ath: DFS JP domain W56 fixed pulse type 3 RADAR detection

    [ Upstream commit d8792393a783158cbb2c39939cb897dc5e5299b6 ]

    Increase pulse width range from 1-2usec to 0-4usec.
    During data traffic HW occasionally fails detecting radar pulses,
    so that SW cannot get enough radar reports to achieve the success rate.

    Tested ath10k hw and fw:
            * QCA9888(10.4-3.5.1-00052)
            * QCA4019(10.4-3.2.1.1-00017)
            * QCA9984(10.4-3.6-00104)
            * QCA988X(10.2.4-1.0-00041)

    Tested ath9k hw: AR9300

    Tested-by: Tamizh chelvam <tamizhr@codeaurora.org>
    Signed-off-by: Tamizh chelvam <tamizhr@codeaurora.org>
    Signed-off-by: Anilkumar Kolli <akolli@codeaurora.org>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ba3e76c3cb404d1f7e560c6072f12f23bceb2533
Author: Maya Erez <merez@codeaurora.org>
Date:   Fri Apr 26 18:43:29 2019 +0300

    wil6210: fix spurious interrupts in 3-msi

    [ Upstream commit e10b0eddd5235aa5aef4e40b970e34e735611a80 ]

    Interrupt is set in ICM (ICR & ~IMV) rising trigger.
    As the driver masks the IRQ after clearing it, there can
    be a race where an additional spurious interrupt is triggered
    when the driver unmask the IRQ.
    This can happen in case HW triggers an interrupt after the clear
    and before the mask.

    To prevent the second spurious interrupt the driver needs to mask the
    IRQ before reading and clearing it.

    Signed-off-by: Maya Erez <merez@codeaurora.org>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 01fbe2f6478cd6eb38ca2cc88ef849a92e0c3ff8
Author: Wen Gong <wgong@codeaurora.org>
Date:   Mon Apr 29 19:17:12 2019 +0800

    ath10k: add peer id check in ath10k_peer_find_by_id

    [ Upstream commit 49ed34b835e231aa941257394716bc689bc98d9f ]
```

```
    For some SDIO chip, the peer id is 65535 for MPDU with error status,
    then test_bit will trigger buffer overflow for peer's memory, if kasan
    enabled, it will report error.

    Reason is when station is in disconnecting status, firmware do not delete
    the peer info since it not disconnected completely, meanwhile some AP will
    still send data packet to station, then hardware will receive the packet
    and send to firmware, firmware's logic will report peer id of 65535 for
    MPDU with error status.

    Add check for overflow the size of peer's peer_ids will avoid the buffer
    overflow access.

    Call trace of kasan:
    dump_backtrace+0x0/0x2ec
    show_stack+0x20/0x2c
    __dump_stack+0x20/0x28
    dump_stack+0xc8/0xec
    print_address_description+0x74/0x240
    kasan_report+0x250/0x26c
    __asan_report_load8_noabort+0x20/0x2c
    ath10k_peer_find_by_id+0x180/0x1e4 [ath10k_core]
    ath10k_htt_t2h_msg_handler+0x100c/0x2fd4 [ath10k_core]
    ath10k_htt_htc_t2h_msg_handler+0x20/0x34 [ath10k_core]
    ath10k_sdio_irq_handler+0xcc8/0x1678 [ath10k_sdio]
    process_sdio_pending_irqs+0xec/0x370
    sdio_run_irqs+0x68/0xe4
    sdio_irq_work+0x1c/0x28
    process_one_work+0x3d8/0x8b0
    worker_thread+0x508/0x7cc
    kthread+0x24c/0x264
    ret_from_fork+0x10/0x18

    Tested with QCA6174 SDIO with firmware
    WLAN.RMH.4.4.1-00007-QCARMSWP-1.

    Signed-off-by: Wen Gong <wgong@codeaurora.org>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e5e1db764e521b4aff809100f74a92c121d0b79f
Author: Dan Carpenter <dan.carpenter@oracle.com>
Date:   Thu Apr 4 11:56:51 2019 +0300

    ath6kl: add some bounds checking

    [ Upstream commit 5d6751eaff672ea77642e74e92e6c0ac7f9709ab ]

    The "ev->traffic_class" and "reply->ac" variables come from the network
    and they're used as an offset into the wmi->stream_exist_for_ac[] array.
    Those variables are u8 so they can be 0-255 but the stream_exist_for_ac[]
    array only has WMM_NUM_AC (4) elements.  We need to add a couple bounds
    checks to prevent array overflows.

    I also modified one existing check from "if (traffic_class > 3) {" to
    "if (traffic_class >= WMM_NUM_AC) {" just to make them all consistent.

    Fixes: bdcd81707973 (" Add ath6kl cleaned up driver")
    Signed-off-by: Dan Carpenter <dan.carpenter@oracle.com>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9eeb3cbd2b2daebc3554928b264790c16edee030
Author: Maya Erez <merez@codeaurora.org>
```

```
Date:   Fri Apr 26 18:43:35 2019 +0300

    wil6210: fix missed MISC mbox interrupt

    [ Upstream commit 7441be71ba7e07791fd4fa2b07c932dff14ff4d9 ]

    When MISC interrupt is triggered due to HALP bit, in parallel
    to mbox events handling by the MISC threaded IRQ, new mbox
    interrupt can be missed in the following scenario:
    1. MISC ICR is read in the IRQ handler
    2. Threaded IRQ is completed and all MISC interrupts are unmasked
    3. mbox interrupt is set by FW
    4. HALP is masked
    The mbox interrupt in step 3 can be missed due to constant high level
    of ICM.
    Masking all MISC IRQs instead of masking only HALP bit in step 4
    will guarantee that ICM will drop to 0 and interrupt will be triggered
    once MISC interrupts will be unmasked.

    Signed-off-by: Maya Erez <merez@codeaurora.org>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4b1c22402ce3df200b30b8437a1dcf4524b4bf3a
Author: Surabhi Vishnoi <svishnoi@codeaurora.org>
Date:   Tue Apr 9 12:13:13 2019 +0530

    ath10k: Fix the wrong value of enums for wmi tlv stats id

    [ Upstream commit 9280f4fc06f44d0b4dc9e831f72d97b3d7cd35d3 ]

    The enum value for WMI_TLV_STAT_PDEV, WMI_TLV_STAT_VDEV
    and WMI_TLV_STAT_PEER is wrong, due to which the vdev stats
    are not received from firmware in wmi_update_stats event.

    Fix the enum values for above stats to receive all stats
    from firmware in WMI_TLV_UPDATE_STATS_EVENTID.

    Tested HW: WCN3990
    Tested FW: WLAN.HL.3.1-00784-QCAHLSWMTPLZ-1

    Fixes: f40a307eb92c ("ath10k: Fill rx duration for each peer in fw_stats for WCN3990)
    Signed-off-by: Surabhi Vishnoi <svishnoi@codeaurora.org>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 835ccc154b77d7e42e47e44c24108c2022aa6c4a
Author: Tim Schumacher <timschumi@gmx.de>
Date:   Mon Mar 18 20:05:57 2019 +0100

    ath9k: Check for errors when reading SREV register

    [ Upstream commit 2f90c7e5d09437a4d8d5546feaae9f1cf48cfbe1 ]

    Right now, if an error is encountered during the SREV register
    read (i.e. an EIO in ath9k_regread()), that error code gets
    passed all the way to __ath9k_hw_init(), where it is visible
    during the "Chip rev not supported" message.

        ath9k_htc 1-1.4:1.0: ath9k_htc: HTC initialized with 33 credits
        ath: phy2: Mac Chip Rev 0x0f.3 is not supported by this driver
        ath: phy2: Unable to initialize hardware; initialization status: -95
        ath: phy2: Unable to initialize hardware; initialization status: -95
        ath9k_htc: Failed to initialize the device
```

```
    Check for -EIO explicitly in ath9k_hw_read_revisions() and return
    a boolean based on the success of the operation. Check for that in
    __ath9k_hw_init() and abort with a more debugging-friendly message
    if reading the revisions wasn't successful.

        ath9k_htc 1-1.4:1.0: ath9k_htc: HTC initialized with 33 credits
        ath: phy2: Failed to read SREV register
        ath: phy2: Could not read hardware revision
        ath: phy2: Unable to initialize hardware; initialization status: -95
        ath: phy2: Unable to initialize hardware; initialization status: -95
        ath9k_htc: Failed to initialize the device

    This helps when debugging by directly showing the first point of
    failure and it could prevent possible errors if a 0x0f.3 revision
    is ever supported.

    Signed-off-by: Tim Schumacher <timschumi@gmx.de>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f2104ba4dec515d57ed77fdd4477c6b4f3b7fd30
Author: Emil Renner Berthing <kernel@esmil.dk>
Date:   Fri Apr 12 12:53:20 2019 +0200

    spi: rockchip: turn down tx dma bursts

    [ Upstream commit 47300728fb213486a830565d2af49da967c9d16a ]

    This fixes tx and bi-directional dma transfers on rk3399-gru-kevin.

    It seems the SPI fifo must have room for 2 bursts when the dma_tx_req
    signal is generated or it might skip some words. This in turn makes
    the rx dma channel never complete for bi-directional transfers.

    Fix it by setting tx burst length to fifo_len / 4 and the dma
    watermark to fifo_len / 2.

    However the rk3399 TRM says (sic):
    "DMAC support incrementing-address burst and fixed-address burst. But in
    the case of access SPI and UART at byte or halfword size, DMAC only
    support fixed-address burst and the address must be aligned to word."

    So this relies on fifo_len being a multiple of 16 such that the
    burst length (= fifo_len / 4) is a multiple of 4 and the addresses
    will be word-aligned.

    Fixes: dcfc861d24ec ("spi: rockchip: adjust dma watermark and burstlen")
    Signed-off-by: Emil Renner Berthing <kernel@esmil.dk>
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0e55b0a03bda5cdc2b373ba9320d3ae0146a19df
Author: Surabhi Vishnoi <svishnoi@codeaurora.org>
Date:   Wed Apr 17 14:01:46 2019 +0530

    ath10k: Do not send probe response template for mesh

    [ Upstream commit 97354f2c432788e3163134df6bb144f4b6289d87 ]

    Currently mac80211 do not support probe response template for
    mesh point. When WMI_SERVICE_BEACON_OFFLOAD is enabled, host
    driver tries to configure probe response template for mesh, but
    it fails because the interface type is not NL80211_IFTYPE_AP but
    NL80211_IFTYPE_MESH_POINT.
```

```
    To avoid this failure, skip sending probe response template to
    firmware for mesh point.

    Tested HW: WCN3990/QCA6174/QCA9984

    Signed-off-by: Surabhi Vishnoi <svishnoi@codeaurora.org>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 23476612f4df55934b97c883f70b6111e4719f1a
Author: Gustavo A. R. Silva <gustavo@embeddedor.com>
Date:   Mon Apr 15 09:56:46 2019 -0500

    wil6210: fix potential out-of-bounds read

    [ Upstream commit bfabdd6997323adbedccb13a3fed1967fb8cf8f5 ]

    Notice that *rc* can evaluate to up to 5, include/linux/netdevice.h:

    enum gro_result {
            GRO_MERGED,
            GRO_MERGED_FREE,
            GRO_HELD,
            GRO_NORMAL,
            GRO_DROP,
            GRO_CONSUMED,
    };
    typedef enum gro_result gro_result_t;

    In case *rc* evaluates to 5, we end up having an out-of-bounds read
    at drivers/net/wireless/ath/wil6210/txrx.c:821:

            wil_dbg_txrx(wil, "Rx complete %d bytes => %s\n",
                         len, gro_res_str[rc]);

    Fix this by adding element "GRO_CONSUMED" to array gro_res_str.

    Addresses-Coverity-ID: 1444666 ("Out-of-bounds read")
    Fixes: 194b482b5055 ("wil6210: Debug print GRO Rx result")
    Signed-off-by: Gustavo A. R. Silva <gustavo@embeddedor.com>
    Reviewed-by: Maya Erez <merez@codeaurora.org>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b38e4647d24ef5c9d459a59e5250ce518046aab9
Author: Toke Høiland-Jørgensen <toke@redhat.com>
Date:   Thu Mar 7 19:39:44 2019 +0100

    ath9k: Don't trust TX status TID number when reporting airtime

    [ Upstream commit 389b72e58259336c2d56d58b660b79cf4b9e0dcb ]

    As already noted a comment in ath_tx_complete_aggr(), the hardware will
    occasionally send a TX status with the wrong tid number. If we trust the
    value, airtime usage will be reported to the wrong AC, which can cause the
    deficit on that AC to become very low, blocking subsequent attempts to
    transmit.

    To fix this, account airtime usage to the TID number from the original skb,
    instead of the one in the hardware TX status report.

    Reported-by: Miguel Catalan Cid <miguel.catalan@i2cat.net>
    Signed-off-by: Toke Høiland-Jørgensen <toke@redhat.com>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 1fd51551c78cecd8affab11945118c7a8446cddc
Author: Pradeep kumar Chitrapu <pradeepc@codeaurora.org>
Date:   Tue Apr 23 16:43:28 2019 +0300

    ath10k: fix incorrect multicast/broadcast rate setting

    [ Upstream commit 93ee3d108fc77e19efeac3ec5aa7d5886711bfef ]

    Invalid rate code is sent to firmware when multicast rate value of 0 is
    sent to driver indicating disabled case, causing broken mesh path.
    so fix that.

    Tested on QCA9984 with firmware 10.4-3.6.1-00827

    Sven tested on IPQ4019 with 10.4-3.5.3-00057 and QCA9888 with 10.4-3.5.3-00053
    (ath10k-firmware) and 10.4-3.6-00140 (linux-firmware 2018-12-16-211de167).

    Fixes: cd93b83ad92 ("ath10k: support for multicast rate control")
    Co-developed-by: Zhi Chen <zhichen@codeaurora.org>
    Signed-off-by: Zhi Chen <zhichen@codeaurora.org>
    Signed-off-by: Pradeep Kumar Chitrapu <pradeepc@codeaurora.org>
    Tested-by: Sven Eckelmann <sven@narfation.org>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8dba6daad53ffc11eb8b4d442c0e6616cbd18941
Author: Alagu Sankar <alagusankar@silex-india.com>
Date:   Fri Apr 19 10:28:49 2019 +0300

    ath10k: htt: don't use txdone_fifo with SDIO

    [ Upstream commit e2a6b711282a371c5153239e0468a48254f17ca6 ]

    HTT High Latency (ATH10K_DEV_TYPE_HL) does not use txdone_fifo at all, we don't
    even initialise it by skipping ath10k_htt_tx_alloc_buf() in
    ath10k_htt_tx_start(). Because of this using QCA6174 SDIO
    ath10k_htt_rx_tx_compl_ind() will crash when it accesses unitialised
    txdone_fifo. So skip txdone_fifo when using High Latency mode.

    Tested with QCA6174 SDIO with firmware WLAN.RMH.4.4.1-00007-QCARMSWP-1.

    Co-developed-by: Wen Gong <wgong@codeaurora.org>
    Signed-off-by: Alagu Sankar <alagusankar@silex-india.com>
    Signed-off-by: Wen Gong <wgong@codeaurora.org>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5f0cf3dfc01a33977bc8bcc096a6dfd90d8573a8
Author: Yingying Tang <yintang@codeaurora.org>
Date:   Thu May 2 21:36:50 2019 +0800

    ath10k: Check tx_stats before use it

    [ Upstream commit 9e7251fa38978b85108c44743e1436d48e8d0d76 ]

    tx_stats will be freed and set to NULL before debugfs_sta node is
    removed in station disconnetion process. So if read the debugfs_sta
    node there may be NULL pointer error. Add check for tx_stats before
    use it to resove this issue.

    Signed-off-by: Yingying Tang <yintang@codeaurora.org>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```