```
commit 5729a900a07b7f166b94577c543d44c4c42e5ec3
Author: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
Date:   Thu May 11 23:17:39 2023 +0900

    Linux 6.3.2

    Link: https://lore.kernel.org/r/20230508094432.603705160@linuxfoundation.org
    Tested-by: Ronald Warsow <rwarsow@gmx.de>
    Tested-by: Shuah Khan <skhan@linuxfoundation.org>
    Tested-by: Ron Economos <re@w6rz.net>
    Tested-by: Justin M. Forbes <jforbes@fedoraproject.org>
    Tested-by: Florian Fainelli <f.fainelli@gmail.com>
    Tested-by: Bagas Sanjaya <bagasdotme@gmail.com>
    Tested-by: Markus Reichelt <lkt+2023@mareichelt.com>
    Link: https://lore.kernel.org/r/20230509030705.399628514@linuxfoundation.org
    Tested-by: Rudi Heitbaum <rudi@heitbaum.com>
    Tested-by: Ronald Warsow <rwarsow@gmx.de>
    Tested-by: Conor Dooley <conor.dooley@microchip.com>
    Tested-by: Bagas Sanjaya <bagasdotme@gmail.com>
    Tested-by: Jon Hunter <jonathanh@nvidia.com>
    Tested-by: Chris Paterson (CIP) <chris.paterson2@renesas.com>
    Tested-by: Florian Fainelli <f.fainelli@gmail.com>
    Tested-by: Markus Reichelt <lkt+2023@mareichelt.com>
    Tested-by: Linux Kernel Functional Testing <lkft@linaro.org>
    Tested-by: Guenter Roeck <linux@roeck-us.net>
    Tested-by: Sudip Mukherjee <sudip.mukherjee@codethink.co.uk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f8486683ffa30456e0be4290282a44c4459a3287
Author: Pablo Neira Ayuso <pablo@netfilter.org>
Date:   Tue May 2 10:25:24 2023 +0200

    netfilter: nf_tables: deactivate anonymous set from preparation phase

    commit c1592a89942e9678f7d9c8030efa777c0d57edab upstream.

    Toggle deleted anonymous sets as inactive in the next generation, so
    users cannot perform any update on it. Clear the generation bitmask
    in case the transaction is aborted.

    The following KASAN splat shows a set element deletion for a bound
    anonymous set that has been already removed in the same transaction.

    [   64.921510] ================================================================
    [   64.923123] BUG: KASAN: wild-memory-access in nf_tables_commit+0xa24/0x1490
[nf_tables]
    [   64.924745] Write of size 8 at addr dead000000000122 by task test/890
    [   64.927903] CPU: 3 PID: 890 Comm: test Not tainted 6.3.0+ #253
    [   64.931120] Call Trace:
    [   64.932699]  <TASK>
    [   64.934292]  dump_stack_lvl+0x33/0x50
    [   64.935908]  ? nf_tables_commit+0xa24/0x1490 [nf_tables]
    [   64.937551]  kasan_report+0xda/0x120
    [   64.939186]  ? nf_tables_commit+0xa24/0x1490 [nf_tables]
    [   64.940814]  nf_tables_commit+0xa24/0x1490 [nf_tables]
    [   64.942452]  ? __kasan_slab_alloc+0x2d/0x60
    [   64.944070]  ? nf_tables_setelem_notify+0x190/0x190 [nf_tables]
    [   64.945710]  ? kasan_set_track+0x21/0x30
    [   64.947323]  nfnetlink_rcv_batch+0x709/0xd90 [nfnetlink]
    [   64.948898]  ? nfnetlink_rcv_msg+0x480/0x480 [nfnetlink]

    Signed-off-by: Pablo Neira Ayuso <pablo@netfilter.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit a2a8706d30634d5687adb1c0e46688f0fc2ea2a5
Author: Pierre Gondois <pierre.gondois@arm.com>
Date:   Fri Apr 14 10:14:51 2023 +0200

    arch_topology: Remove early cacheinfo error message if -ENOENT

    commit 3522340199cc060b70f0094e3039bdb43c3f6ee1 upstream.

    fetch_cache_info() tries to get the number of cache leaves/levels
    for each CPU in order to pre-allocate memory for cacheinfo struct.
    Allocating this memory later triggers a:
      'BUG: sleeping function called from invalid context'
    in PREEMPT_RT kernels.

    If there is no cache related information available in DT or ACPI,
    fetch_cache_info() fails and an error message is printed:
      'Early cacheinfo failed, ret = ...'

    Not having cache information should be a valid configuration.
    Remove the error message if fetch_cache_info() fails with -ENOENT.

    Suggested-by: Conor Dooley <conor.dooley@microchip.com>
    Link: https://lore.kernel.org/all/20230404-hatred-swimmer-6fecdf33b57a@spud/
    Signed-off-by: Pierre Gondois <pierre.gondois@arm.com>
    Reviewed-by: Conor Dooley <conor.dooley@microchip.com>
    Link: https://lore.kernel.org/r/20230414081453.244787-4-pierre.gondois@arm.com
    Signed-off-by: Sudeep Holla <sudeep.holla@arm.com>
    Cc: Florian Fainelli <f.fainelli@gmail.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 8f8cc81abcff5a64489129c1d75839dc4d67fe3d
Author: Thomas Gleixner <tglx@linutronix.de>
Date:   Mon May 1 17:42:06 2023 +0200

    debugobject: Ensure pool refill (again)

    commit 0af462f19e635ad522f28981238334620881badc upstream.

    The recent fix to ensure atomicity of lookup and allocation inadvertently
    broke the pool refill mechanism.

    Prior to that change debug_objects_activate() and debug_objecs_assert_init()
    invoked debug_objecs_init() to set up the tracking object for statically
    initialized objects. That's not longer the case and debug_objecs_init() is
    now the only place which does pool refills.

    Depending on the number of statically initialized objects this can be
    enough to actually deplete the pool, which was observed by Ido via a
    debugobjects OOM warning.

    Restore the old behaviour by adding explicit refill opportunities to
    debug_objects_activate() and debug_objecs_assert_init().

    Fixes: 63a759694eed ("debugobject: Prevent init race with static objects")
    Reported-by: Ido Schimmel <idosch@nvidia.com>
    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Tested-by: Ido Schimmel <idosch@nvidia.com>
    Link: https://lore.kernel.org/r/871qk05a9d.ffs@tglx
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3139ea00ef1b9f0d964b068b58a48b989d71c0b4
Author: Paulo Alcantara <pc@manguebit.com>
Date:   Thu Apr 27 04:40:08 2023 -0300

    cifs: avoid potential races when handling multiple dfs tcons
```

```
    commit 6be2ea33a4093402252724a00c4af8033725184c upstream.

    Now that a DFS tcon manages its own list of DFS referrals and
    sessions, there is no point in having a single worker to refresh
    referrals of all DFS tcons.  Make it faster and less prone to race
    conditions when having several mounts by queueing a worker per DFS
    tcon that will take care of refreshing only the DFS referrals related
    to it.

    Cc: stable@vger.kernel.org # v6.2+
    Signed-off-by: Paulo Alcantara (SUSE) <pc@manguebit.com>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e2fa5a6841d424c85b304fcef04201200d0c33b1
Author: Paulo Alcantara <pc@manguebit.com>
Date:   Wed Apr 26 13:43:53 2023 -0300

    cifs: protect access of TCP_Server_Info::{origin,leaf}_fullpath

    commit 3dc9c433c9dde15477d02b609ccb4328e2adb6dc upstream.

    Protect access of TCP_Server_Info::{origin,leaf}_fullpath when
    matching DFS connections, and get rid of
    TCP_Server_Info::current_fullpath while we're at it.

    Cc: stable@vger.kernel.org # v6.2+
    Signed-off-by: Paulo Alcantara (SUSE) <pc@manguebit.com>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 553476df55a111e6a66ad9155256aec0ec1b7ad0
Author: Paulo Alcantara <pc@manguebit.com>
Date:   Tue Apr 25 02:42:56 2023 -0300

    cifs: fix potential race when tree connecting ipc

    commit ee20d7c6100752eaf2409d783f4f1449c29ea33d upstream.

    Protect access of TCP_Server_Info::hostname when building the ipc tree
    name as it might get freed in cifsd thread and thus causing an
    use-after-free bug in __tree_connect_dfs_target().  Also, while at it,
    update status of IPC tcon on success and then avoid any extra tree
    connects.

    Cc: stable@vger.kernel.org # v6.2+
    Signed-off-by: Paulo Alcantara (SUSE) <pc@manguebit.com>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c082c3be0f96e759ff2e361d929832fda0b93851
Author: Paulo Alcantara <pc@manguebit.com>
Date:   Sun Apr 23 23:26:51 2023 -0300

    cifs: fix sharing of DFS connections

    commit 8e3554150d6c80a84b3cb046615d1a0e943811dc upstream.

    When matching DFS connections, we can't rely on the values set in
    cifs_sb_info::prepath and cifs_tcon::tree_name as they might change
    during DFS failover.  The DFS referrals related to a specific DFS tcon
    are already matched earlier in match_server(), therefore we can safely
    skip those checks altogether as the connection is guaranteed to be
    unique for the DFS tcon.
```

```
    Besides, when creating or finding an SMB session, make sure to also
    refcount any DFS root session related to it (cifs_ses::dfs_root_ses),
    so if a new DFS mount ends up reusing the connection from the old
    mount while there was an umount(2) still in progress (e.g. umount(2)
    -> cifs_umount() -> reconnect -> cifs_put_tcon()), the connection
    could potentially be put right after the umount(2) finished.

    Patch has minor update to include fix for unused variable issue
    noted by the kernel test robot

    Reported-by: kernel test robot <lkp@intel.com>
    Link: https://lore.kernel.org/oe-kbuild-all/202305041040.j7W2xQSy-lkp@intel.com/
    Cc: stable@vger.kernel.org # v6.2+
    Signed-off-by: Paulo Alcantara (SUSE) <pc@manguebit.com>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 7afd4e1fceb4236fd8e89a1516cdded6987f852e
Author: Paulo Alcantara <pc@manguebit.com>
Date:   Thu Apr 27 16:07:38 2023 -0300

    cifs: protect session status check in smb2_reconnect()

    commit 5bff9f741af60b143a5ae73417a8ec47fd5ff2f4 upstream.

    Use @ses->ses_lock to protect access of @ses->ses_status.

    Cc: stable@vger.kernel.org
    Signed-off-by: Paulo Alcantara (SUSE) <pc@manguebit.com>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 0b08c4c499200be67d54c439d56e5ea866869945
Author: Paulo Alcantara <pc@manguebit.com>
Date:   Fri Apr 21 15:52:32 2023 -0300

    cifs: fix potential use-after-free bugs in TCP_Server_Info::hostname

    commit 90c49fce1c43e1cc152695e20363ff5087897c09 upstream.

    TCP_Server_Info::hostname may be updated once or many times during
    reconnect, so protect its access outside reconnect path as well and
    then prevent any potential use-after-free bugs.

    Cc: stable@vger.kernel.org
    Signed-off-by: Paulo Alcantara (SUSE) <pc@manguebit.com>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3a0a9211d7d0138d55aecd209b05e6d4a9eec383
Author: Tobias Holl <tobias@tholl.xyz>
Date:   Wed May 3 08:59:50 2023 -0600

    io_uring/rsrc: check for nonconsecutive pages

    commit 776617db78c6d208780e7c69d4d68d1fa82913de upstream.

    Pages that are from the same folio do not necessarily need to be
    consecutive. In that case, we cannot consolidate them into a single bvec
    entry. Before applying the huge page optimization from commit 57bebf807e2a
    ("io_uring/rsrc: optimise registered huge pages"), check that the memory
    is actually consecutive.

    Cc: stable@vger.kernel.org
```

```
    Fixes: 57bebf807e2a ("io_uring/rsrc: optimise registered huge pages")
    Signed-off-by: Tobias Holl <tobias@tholl.xyz>
    [axboe: formatting]
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 8bc13e33c152badb6459f3cdb12fa52d493d2a4e
Author: Adrian Hunter <adrian.hunter@intel.com>
Date:   Mon Apr 3 18:48:31 2023 +0300

    perf intel-pt: Fix CYC timestamps after standalone CBR

    commit 430635a0ef1ce958b7b4311f172694ece2c692b8 upstream.

    After a standalone CBR (not associated with TSC), update the cycles
    reference timestamp and reset the cycle count, so that CYC timestamps
    are calculated relative to that point with the new frequency.

    Fixes: cc33618619cefc6d ("perf tools: Add Intel PT support for decoding CYC packets")
    Signed-off-by: Adrian Hunter <adrian.hunter@intel.com>
    Cc: Adrian Hunter <adrian.hunter@intel.com>
    Cc: Ian Rogers <irogers@google.com>
    Cc: Jiri Olsa <jolsa@kernel.org>
    Cc: Namhyung Kim <namhyung@kernel.org>
    Cc: stable@vger.kernel.org
    Link: https://lore.kernel.org/r/20230403154831.8651-2-adrian.hunter@intel.com
    Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c055b61d6dc4ec302598625128966ed72696d72f
Author: Adrian Hunter <adrian.hunter@intel.com>
Date:   Mon Apr 3 18:48:30 2023 +0300

    perf auxtrace: Fix address filter entire kernel size

    commit 1f9f33ccf0320be21703d9195dd2b36a1c9a07cb upstream.

    kallsyms is not completely in address order.

    In find_entire_kern_cb(), calculate the kernel end from the maximum
    address not the last symbol.

    Example:

     Before:

        $ sudo cat /proc/kallsyms | grep ' [twTw] ' | tail -1
        ffffffffc00b8bd0 t bpf_prog_6deef7357e7b4530    [bpf]
        $ sudo cat /proc/kallsyms | grep ' [twTw] ' | sort | tail -1
        ffffffffc15e0cc0 t iwl_mvm_exit [iwlmvm]
        $ perf.d093603a05aa record -v --kcore -e intel_pt// --filter 'filter *' -- uname |&
grep filter
        Address filter: filter 0xffffffff93200000/0x2ceba000

     After:

        $ perf.8fb0f7a01f8e record -v --kcore -e intel_pt// --filter 'filter *' -- uname |&
grep filter
        Address filter: filter 0xffffffff93200000/0x2e3e2000

    Fixes: 1b36c03e356936d6 ("perf record: Add support for using symbols in address
filters")
    Signed-off-by: Adrian Hunter <adrian.hunter@intel.com>
    Cc: Adrian Hunter <adrian.hunter@intel.com>
    Cc: Ian Rogers <irogers@google.com>
```

```
     Cc: Jiri Olsa <jolsa@kernel.org>
     Cc: Namhyung Kim <namhyung@kernel.org>
     Cc: stable@vger.kernel.org
     Link: https://lore.kernel.org/r/20230403154831.8651-2-adrian.hunter@intel.com
     Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
     Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

 commit d958b0ddbb57e5d31ab36db8b78277bd0ec4bc9f
 Author: Li Lingfeng <lilingfeng3@huawei.com>
 Date:   Tue Apr 18 16:38:04 2023 +0800

     dm: don't lock fs when the map is NULL in process of resume

     commit 38d11da522aacaa05898c734a1cec86f1e611129 upstream.

     Commit fa247089de99 ("dm: requeue IO if mapping table not yet available")
     added a detection of whether the mapping table is available in the IO
     submission process. If the mapping table is unavailable, it returns
     BLK_STS_RESOURCE and requeues the IO.
     This can lead to the following deadlock problem:

     dm create                                          mount
     ioctl(DM_DEV_CREATE_CMD)
     ioctl(DM_TABLE_LOAD_CMD)
                                    do_mount
                                     vfs_get_tree
                                      ext4_get_tree
                                       get_tree_bdev
                                        sget_fc
                                         alloc_super
                                          // got &s->s_umount
                                          down_write_nested(&s->s_umount, ...);
                                         ext4_fill_super
                                          ext4_load_super
                                           ext4_read_bh
                                            submit_bio
                                            // submit and wait io end
     ioctl(DM_DEV_SUSPEND_CMD)
     dev_suspend
      do_resume
       dm_suspend
        __dm_suspend
         lock_fs
          freeze_bdev
           get_active_super
            grab_super
             // wait for &s->s_umount
             down_write(&s->s_umount);
       dm_swap_table
        __bind
         // set md->map(can't get here)

     IO will be continuously requeued while holding the lock since mapping
     table is NULL. At the same time, mapping table won't be set since the
     lock is not available.
     Like request-based DM, bio-based DM also has the same problem.

     It's not proper to just abort IO if the mapping table not available.
     So clear DM_SKIP_LOCKFS_FLAG when the mapping table is NULL, this
     allows the DM table to be loaded and the IO submitted upon resume.

     Fixes: fa247089de99 ("dm: requeue IO if mapping table not yet available")
     Cc: stable@vger.kernel.org
     Signed-off-by: Li Lingfeng <lilingfeng3@huawei.com>
     Signed-off-by: Mike Snitzer <snitzer@kernel.org>
```

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

commit a554e6ec3626d4c11f55d7eef8d6aa93fb211c24
Author: Mike Snitzer <snitzer@kernel.org>
Date:   Mon Apr 17 11:59:56 2023 -0400

    dm ioctl: fix nested locking in table_clear() to remove deadlock concern

    commit 3d32aaa7e66d5c1479a3c31d6c2c5d45dd0d3b89 upstream.

    syzkaller found the following problematic rwsem locking (with write
    lock already held):

     down_read+0x9d/0x450 kernel/locking/rwsem.c:1509
     dm_get_inactive_table+0x2b/0xc0 drivers/md/dm-ioctl.c:773
     __dev_status+0x4fd/0x7c0 drivers/md/dm-ioctl.c:844
     table_clear+0x197/0x280 drivers/md/dm-ioctl.c:1537

    In table_clear, it first acquires a write lock
    https://elixir.bootlin.com/linux/v6.2/source/drivers/md/dm-ioctl.c#L1520
    down_write(&_hash_lock);

    Then before the lock is released at L1539, there is a path shown above:
    table_clear -> __dev_status -> dm_get_inactive_table ->  down_read
    https://elixir.bootlin.com/linux/v6.2/source/drivers/md/dm-ioctl.c#L773
    down_read(&_hash_lock);

    It tries to acquire the same read lock again, resulting in the deadlock
    problem.

    Fix this by moving table_clear()'s __dev_status() call to after its
    up_write(&_hash_lock);

    Cc: stable@vger.kernel.org
    Reported-by: Zheng Zhang <zheng.zhang@email.ucr.edu>
    Signed-off-by: Mike Snitzer <snitzer@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 8258d84a7917aeece773716518deadb7ad776cb7
Author: Mikulas Patocka <mpatocka@redhat.com>
Date:   Tue Apr 18 15:57:47 2023 -0400

    dm flakey: fix a crash with invalid table line

    commit 98dba02d9a93eec11bffbb93c7c51624290702d2 upstream.

    This command will crash with NULL pointer dereference:
     dmsetup create flakey --table \
      "0 `blockdev --getsize /dev/ram0` flakey /dev/ram0 0 0 1 2 corrupt_bio_byte 512"

    Fix the crash by checking if arg_name is non-NULL before comparing it.

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

commit e09a592fdd6c716506774bdbebb5f6c537b47767
Author: Mike Snitzer <snitzer@kernel.org>
Date:   Tue Apr 4 13:34:28 2023 -0400

    dm integrity: call kmem_cache_destroy() in dm_integrity_init() error path

    commit 6b79a428c02769f2a11f8ae76bf866226d134887 upstream.
```

```
    Otherwise the journal_io_cache will leak if dm_register_target() fails.

    Cc: stable@vger.kernel.org
    Signed-off-by: Mike Snitzer <snitzer@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f3063e860f35ec6b7f7c197842f770f51546f625
Author: Mike Snitzer <snitzer@kernel.org>
Date:   Tue Apr 4 11:59:00 2023 -0400

    dm clone: call kmem_cache_destroy() in dm_clone_init() error path

    commit 6827af4a9a9f5bb664c42abf7c11af4978d72201 upstream.

    Otherwise the _hydration_cache will leak if dm_register_target() fails.

    Cc: stable@vger.kernel.org
    Signed-off-by: Mike Snitzer <snitzer@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit af8eccd6e18ee904adf79f320323d03cea3baec9
Author: Yeongjin Gil <youngjin.gil@samsung.com>
Date:   Mon Mar 20 15:59:32 2023 +0900

    dm verity: fix error handling for check_at_most_once on FEC

    commit e8c5d45f82ce0c238a4817739892fe8897a3dcc3 upstream.

    In verity_end_io(), if bi_status is not BLK_STS_OK, it can be return
    directly. But if FEC configured, it is desired to correct the data page
    through verity_verify_io. And the return value will be converted to
    blk_status and passed to verity_finish_io().

    BTW, when a bit is set in v->validated_blocks, verity_verify_io() skips
    verification regardless of I/O error for the corresponding bio. In this
    case, the I/O error could not be returned properly, and as a result,
    there is a problem that abnormal data could be read for the
    corresponding block.

    To fix this problem, when an I/O error occurs, do not skip verification
    even if the bit related is set in v->validated_blocks.

    Fixes: 843f38d382b1 ("dm verity: add 'check_at_most_once' option to only validate hashes
once")
    Cc: stable@vger.kernel.org
    Reviewed-by: Sungjong Seo <sj1557.seo@samsung.com>
    Signed-off-by: Yeongjin Gil <youngjin.gil@samsung.com>
    Signed-off-by: Mike Snitzer <snitzer@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit dc142108d862a51502084d30eb23a6c7aac9979f
Author: Cindy Lu <lulu@redhat.com>
Date:   Thu Apr 20 23:17:34 2023 +0800

    vhost_vdpa: fix unmap process in no-batch mode

    commit c82729e06644f4e087f5ff0f91b8fb15e03b8890 upstream.

    While using the vdpa device with vIOMMU enabled
    in the guest VM, when the vdpa device bind to vfio-pci and run testpmd
    then system will fail to unmap.
    The test process is
    Load guest VM --> attach to virtio driver--> bind to vfio-pci driver
    So the mapping process is
    1)batched mode map to normal MR
```

```
    2)batched mode unmapped the normal MR
    3)unmapped all the memory
    4)mapped to iommu MR

    This error happened in step 3). The iotlb was freed in step 2)
    and the function vhost_vdpa_process_iotlb_msg will return fail
    Which causes failure.

    To fix this, we will not remove the AS while the iotlb->nmaps is 0.
    This will free in the vhost_vdpa_clean

    Cc: stable@vger.kernel.org
    Fixes: aaca8373c4b1 ("vhost-vdpa: support ASID based IOTLB API")
    Signed-off-by: Cindy Lu <lulu@redhat.com>
    Message-Id: <20230420151734.860168-1-lulu@redhat.com>
    Signed-off-by: Michael S. Tsirkin <mst@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 679e778075f0a20ff2c2123aa6127dc6025e7fdf
Author: Peter Xu <peterx@redhat.com>
Date:   Mon Apr 17 15:53:12 2023 -0400

    mm/hugetlb: fix uffd-wp during fork()

    commit 5a2f8d22ace4c8ac8798fab836dca7350fa710b1 upstream.

    Patch series "mm/hugetlb: More fixes around uffd-wp vs fork() / RO pins",
    v2.


    This patch (of 6):

    There're a bunch of things that were wrong:

      - Reading uffd-wp bit from a swap entry should use pte_swp_uffd_wp()
        rather than huge_pte_uffd_wp().

      - When copying over a pte, we should drop uffd-wp bit when
        !EVENT_FORK (aka, when !userfaultfd_wp(dst_vma)).

      - When doing early CoW for private hugetlb (e.g. when the parent page was
        pinned), uffd-wp bit should be properly carried over if necessary.

    No bug reported probably because most people do not even care about these
    corner cases, but they are still bugs and can be exposed by the recent unit
    tests introduced, so fix all of them in one shot.

    Link: https://lkml.kernel.org/r/20230417195317.898696-1-peterx@redhat.com
    Link: https://lkml.kernel.org/r/20230417195317.898696-2-peterx@redhat.com
    Fixes: bc70fbf269fd ("mm/hugetlb: handle uffd-wp during fork()")
    Signed-off-by: Peter Xu <peterx@redhat.com>
    Reviewed-by: David Hildenbrand <david@redhat.com>
    Cc: Andrea Arcangeli <aarcange@redhat.com>
    Cc: Axel Rasmussen <axelrasmussen@google.com>
    Cc: Mika Penttilä <mpenttil@redhat.com>
    Cc: Mike Kravetz <mike.kravetz@oracle.com>
    Cc: Nadav Amit <nadav.amit@gmail.com>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 2b2d5472d6ce863c6da7e0c8e2be6cecca48d7ee
Author: Lorenzo Stoakes <lstoakes@gmail.com>
Date:   Sun Apr 30 16:07:07 2023 +0100
```

```
    mm/mempolicy: correctly update prev when policy is equal on mbind

    commit 00ca0f2e86bf40b016a646e6323a8941a09cf106 upstream.

    The refactoring in commit f4e9e0e69468 ("mm/mempolicy: fix use-after-free
    of VMA iterator") introduces a subtle bug which arises when attempting to
    apply a new NUMA policy across a range of VMAs in mbind_range().

    The refactoring passes a **prev pointer to keep track of the previous VMA
    in order to reduce duplication, and in all but one case it keeps this
    correctly updated.

    The bug arises when a VMA within the specified range has an equivalent
    policy as determined by mpol_equal() - which unlike other cases, does not
    update prev.

    This can result in a situation where, later in the iteration, a VMA is
    found whose policy does need to change.  At this point, vma_merge() is
    invoked with prev pointing to a VMA which is before the previous VMA.

    Since vma_merge() discovers the curr VMA by looking for the one
    immediately after prev, it will now be in a situation where this VMA is
    incorrect and the merge will not proceed correctly.

    This is checked in the VM_WARN_ON() invariant case with end >
    curr->vm_end, which, if a merge is possible, results in a warning (if
    CONFIG_DEBUG_VM is specified).

    I note that vma_merge() performs these invariant checks only after
    merge_prev/merge_next are checked, which is debatable as it hides this
    issue if no merge is possible even though a buggy situation has arisen.

    The solution is simply to update the prev pointer even when policies are
    equal.

    This caused a bug to arise in the 6.2.y stable tree, and this patch
    resolves this bug.

    Link:
https://lkml.kernel.org/r/83f1d612acb519d777bebf7f3359317c4e7f4265.1682866629.git.lstoakes@g
mail.com
    Fixes: f4e9e0e69468 ("mm/mempolicy: fix use-after-free of VMA iterator")
    Signed-off-by: Lorenzo Stoakes <lstoakes@gmail.com>
    Reported-by: kernel test robot <oliver.sang@intel.com>
      Link: https://lore.kernel.org/oe-lkp/202304292203.44ddeff6-oliver.sang@intel.com
    Cc: Liam R. Howlett <Liam.Howlett@oracle.com>
    Cc: Mel Gorman <mgorman@suse.de>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a5f3d7b311109fc597f6c045e7e77044398af40e
Author: Hugh Dickins <hughd@google.com>
Date:   Sun Apr 16 22:17:05 2023 -0700

    ia64: fix an addr to taddr in huge_pte_offset()

    commit 3647ebcfbfca384840231fe13fae665453238a61 upstream.

    I know nothing of ia64 htlbpage_to_page(), but guess that the p4d
    line should be using taddr rather than addr, like everywhere else.

    Link: https://lkml.kernel.org/r/732eae88-3beb-246-2c72-281de786740@google.com
    Fixes: c03ab9e32a2c ("ia64: add support for folded p4d page tables")
    Signed-off-by: Hugh Dickins <hughd@google.com
```

```
    Acked-by: Mike Kravetz <mike.kravetz@oracle.com>
    Acked-by: Mike Rapoport (IBM) <rppt@kernel.org>
    Cc: Ard Biesheuvel <ardb@kernel.org>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit d41ffa8e81f74874d59266a0c784bec396612db7
Author: Stefan Haberland <sth@linux.ibm.com>
Date:   Wed Apr 5 16:20:17 2023 +0200

    s390/dasd: fix hanging blockdevice after request requeue

    commit d8898ee50edecacdf0141f26fd90acf43d7e9cd7 upstream.

    The DASD driver does not kick the requeue list when requeuing IO requests
    to the blocklayer. This might lead to hanging blockdevice when there is
    no other trigger for this.

    Fix by automatically kick the requeue list when requeuing DASD requests
    to the blocklayer.

    Fixes: e443343e509a ("s390/dasd: blk-mq conversion")
    CC: stable@vger.kernel.org # 4.14+
    Signed-off-by: Stefan Haberland <sth@linux.ibm.com>
    Reviewed-by: Jan Hoeppner <hoeppner@linux.ibm.com>
    Reviewed-by: Halil Pasic <pasic@linux.ibm.com>
    Link: https://lore.kernel.org/r/20230405142017.2446986-8-sth@linux.ibm.com
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3f466c7bc5d7ee1bb57ff0ced2bf00a4e5d55c13
Author: Qu Wenruo <wqu@suse.com>
Date:   Thu Apr 6 13:00:34 2023 +0800

    btrfs: scrub: reject unsupported scrub flags

    commit 604e6681e114d05a2e384c4d1e8ef81918037ef5 upstream.

    Since the introduction of scrub interface, the only flag that we support
    is BTRFS_SCRUB_READONLY.  Thus there is no sanity checks, if there are
    some undefined flags passed in, we just ignore them.

    This is problematic if we want to introduce new scrub flags, as we have
    no way to determine if such flags are supported.

    Address the problem by introducing a check for the flags, and if
    unsupported flags are set, return -EOPNOTSUPP to inform the user space.

    This check should be backported for all supported kernels before any new
    scrub flags are introduced.

    CC: stable@vger.kernel.org # 4.14+
    Reviewed-by: Anand Jain <anand.jain@oracle.com>
    Signed-off-by: Qu Wenruo <wqu@suse.com>
    Reviewed-by: David Sterba <dsterba@suse.com>
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 74f89661542b35987dc568e94993f29e8f56a2ce
Author: Peng Liu <liupeng17@lenovo.com>
Date:   Tue Mar 21 14:19:29 2023 +0800

    scripts/gdb: fix lx-timerlist for Python3
```

```
    commit 7362042f3556528e9e9b1eb5ce8d7a3a6331476b upstream.

    Below incompatibilities between Python2 and Python3 made lx-timerlist fail
    to run under Python3.

    o xrange() is replaced by range() in Python3
    o bytes and str are different types in Python3
    o the return value of Inferior.read_memory() is memoryview object in
      Python3

    akpm: cc stable so that older kernels are properly debuggable under newer
    Python.

    Link:
https://lkml.kernel.org/r/TYCP286MB2146EE1180A4D5176CBA8AB2C6819@TYCP286MB2146.JPNP286.PROD.
OUTLOOK.COM
    Signed-off-by: Peng Liu <liupeng17@lenovo.com>
    Reviewed-by: Jan Kiszka <jan.kiszka@siemens.com>
    Cc: Florian Fainelli <f.fainelli@gmail.com>
    Cc: Kieran Bingham <kbingham@kernel.org>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit eea78ab6605eee68867517ffe9727e54b4ce594d
Author: Marc Dionne <marc.dionne@auristor.com>
Date:   Fri Dec 2 10:19:42 2022 -0400

    afs: Avoid endless loop if file is larger than expected

    [ Upstream commit 9ea4eff4b6f4f36546d537a74da44fd3f30903ab ]

    afs_read_dir fetches an amount of data that's based on what the inode
    size is thought to be.  If the file on the server is larger than what
    was fetched, the code rechecks i_size and retries.  If the local i_size
    was not properly updated, this can lead to an endless loop of fetching
    i_size from the server and noticing each time that the size is larger on
    the server.

    If it is known that the remote size is larger than i_size, bump up the
    fetch size to that size.

    Fixes: f3ddee8dc4e2 ("afs: Fix directory handling")
    Signed-off-by: Marc Dionne <marc.dionne@auristor.com>
    Signed-off-by: David Howells <dhowells@redhat.com>
    cc: linux-afs@lists.infradead.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 248b7d24124708ee48d1c6507f13476f26a133b1
Author: David Howells <dhowells@redhat.com>
Date:   Tue Dec 6 13:49:42 2022 +0000

    afs: Fix getattr to report server i_size on dirs, not local size

    [ Upstream commit 45f66fa03ba9943cca5af88d691399332b8bde08 ]

    Fix afs_getattr() to report the server's idea of the file size of a
    directory rather than the local size.  The local size may differ as we edit
    the local copy to avoid having to redownload it and we may end up with a
    differently structured blob of a different size.

    However, if the directory is discarded from the pagecache we then download
    it again and the user may see the directory file size apparently change.

    Fixes: 63a4681ff39c ("afs: Locally edit directory data for mkdir/create/unlink/...")
```

```
    Signed-off-by: David Howells <dhowells@redhat.com>
    cc: Marc Dionne <marc.dionne@auristor.com>
    cc: linux-afs@lists.infradead.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 411c97c124055eddd4bf764390220f597a9dca74
Author: Marc Dionne <marc.dionne@auristor.com>
Date:   Fri Dec 2 10:07:01 2022 -0400

    afs: Fix updating of i_size with dv jump from server

    [ Upstream commit d7f74e9a917503ee78f2b603a456d7227cf38919 ]

    If the data version returned from the server is larger than expected,
    the local data is invalidated, but we may still want to note the remote
    file size.

    Since we're setting change_size, we have to also set data_changed
    for the i_size to get updated.

    Fixes: 3f4aa9818163 ("afs: Fix EOF corruption")
    Signed-off-by: Marc Dionne <marc.dionne@auristor.com>
    Signed-off-by: David Howells <dhowells@redhat.com>
    cc: linux-afs@lists.infradead.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f17d1bc39352d339ffb83887a3fff0740436294e
Author: Chen Yu <yu.c.chen@intel.com>
Date:   Fri Apr 14 20:10:42 2023 +0800

    PM: hibernate: Do not get block device exclusively in test_resume mode

    [ Upstream commit 5904de0d735bbb3b4afe9375c5b4f9748f882945 ]

    The system refused to do a test_resume because it found that the
    swap device has already been taken by someone else. Specifically,
    the swsusp_check()->blkdev_get_by_dev(FMODE_EXCL) is supposed to
    do this check.

    Steps to reproduce:
     dd if=/dev/zero of=/swapfile bs=$(cat /proc/meminfo |
           awk '/MemTotal/ {print $2}') count=1024 conv=notrunc
     mkswap /swapfile
     swapon /swapfile
     swap-offset /swapfile
     echo 34816 > /sys/power/resume_offset
     echo test_resume > /sys/power/disk
     echo disk > /sys/power/state

     PM: Using 3 thread(s) for compression
     PM: Compressing and saving image data (293150 pages)...
     PM: Image saving progress:   0%
     PM: Image saving progress:  10%
     ata1: SATA link up 1.5 Gbps (SStatus 113 SControl 300)
     ata1.00: configured for UDMA/100
     ata2: SATA link down (SStatus 0 SControl 300)
     ata5: SATA link down (SStatus 0 SControl 300)
     ata6: SATA link down (SStatus 0 SControl 300)
     ata3: SATA link down (SStatus 0 SControl 300)
     ata4: SATA link down (SStatus 0 SControl 300)
     PM: Image saving progress:  20%
     PM: Image saving progress:  30%
     PM: Image saving progress:  40%
     PM: Image saving progress:  50%
     pcieport 0000:00:02.5: pciehp: Slot(0-5): No device found
```

```
      PM: Image saving progress:  60%
      PM: Image saving progress:  70%
      PM: Image saving progress:  80%
      PM: Image saving progress:  90%
      PM: Image saving done
      PM: hibernation: Wrote 1172600 kbytes in 2.70 seconds (434.29 MB/s)
      PM: S|
      PM: hibernation: Basic memory bitmaps freed
      PM: Image not found (code -16)

    This is because when using the swapfile as the hibernation storage,
    the block device where the swapfile is located has already been mounted
    by the OS distribution(usually mounted as the rootfs). This is not
    an issue for normal hibernation, because software_resume()->swsusp_check()
    happens before the block device(rootfs) mount. But it is a problem for the
    test_resume mode. Because when test_resume happens, the block device has
    been mounted already.

    Thus remove the FMODE_EXCL for test_resume mode. This would not be a
    problem because in test_resume stage, the processes have already been
    frozen, and the race condition described in
    Commit 39fbef4b0f77 ("PM: hibernate: Get block device exclusively in swsusp_check()")
    is unlikely to happen.

    Fixes: 39fbef4b0f77 ("PM: hibernate: Get block device exclusively in swsusp_check()")
    Reported-by: Yifan Li <yifan2.li@intel.com>
    Suggested-by: Pavankumar Kondeti <quic_pkondeti@quicinc.com>
    Tested-by: Pavankumar Kondeti <quic_pkondeti@quicinc.com>
    Tested-by: Wendy Wang <wendy.wang@intel.com>
    Signed-off-by: Chen Yu <yu.c.chen@intel.com>
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d331f2aaaa014936a49a57cf4f930014495698a9
Author: Chen Yu <yu.c.chen@intel.com>
Date:   Fri Apr 14 20:10:26 2023 +0800

    PM: hibernate: Turn snapshot_test into global variable

    [ Upstream commit 08169a162f97819d3e5b4a342bb9cf5137787154 ]

    There is need to check snapshot_test and open block device
    in different mode, so as to avoid the race condition.

    No functional changes intended.

    Suggested-by: Pavankumar Kondeti <quic_pkondeti@quicinc.com>
    Signed-off-by: Chen Yu <yu.c.chen@intel.com>
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Stable-dep-of: 5904de0d735b ("PM: hibernate: Do not get block device exclusively in
test_resume mode")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cdd3fa5b77d39be8916c3f920f8971e2edf87992
Author: Hans de Goede <hdegoede@redhat.com>
Date:   Thu Apr 20 13:02:20 2023 +0200

    ACPI: PM: Do not turn of unused power resources on the Toshiba Click Mini

    [ Upstream commit 9b04d99788cf475cbd277f30ec66230ccb7e99f4 ]

    The CPR3 power resource on the Toshiba Click Mini toggles a GPIO
    which is called SISP (for SIS touchscreen power?) on/off.

    This CPR3 power resource is not listed in any _PR? lists, let alone
```

```
    in a _PR0 list for the SIS0817 touchscreen ACPI device which needs it.

    Before commit a1224f34d72a ("ACPI: PM: Check states of power resources
    during initialization") this was not an issue because since nothing
    referenced the CPR3 power resource its state was always
    ACPI_POWER_RESOURCE_STATE_UNKNOWN and power resources with this state
    get ignored by acpi_turn_off_unused_power_resources().

    This clearly is a bug in the DSDT of this device. Add a DMI quirk
    to make acpi_turn_off_unused_power_resources() a no-op on this
    model to fix the touchscreen no longer working since kernel 5.16 .

    This quirk also causes 2 other power resources to not get turned
    off, but the _OFF method on these already was a no-op, so this makes
    no difference for the other 2 power resources.

    Fixes: a1224f34d72a ("ACPI: PM: Check states of power resources during initialization")
    Reported-by: Gé Koerkamp <ge.koerkamp@gmail.com>
    Link: https://bugzilla.kernel.org/show_bug.cgi?id=216946
    Link: https://lore.kernel.org/regressions/32a14a8a-9795-4c8c-7e00-
da9012f548f8@leemhuis.info/
    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2a488602e3f09ef9e50feb5448ae46515a6fa789
Author: Dan Carpenter <dan.carpenter@linaro.org>
Date:   Wed Apr 19 17:30:00 2023 +0300

    hte: tegra-194: Fix off by one in tegra_hte_map_to_line_id()

    [ Upstream commit e078180d66848a6a890daf0a3ce28dc43cc66790 ]

    The "map_sz" is the number of elements in the "m" array so the >
    comparison needs to be changed to >= to prevent an out of bounds
    read.

    Fixes: 09574cca6ad6 ("hte: Add Tegra194 HTE kernel provider")
    Signed-off-by: Dan Carpenter <dan.carpenter@linaro.org>
    Acked-by: Dipen Patel <dipenp@nvidia.com>
    Signed-off-by: Dipen Patel <dipenp@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d233581fbfbdf04ba0c3ca8428ce6a9a90ada866
Author: Arnd Bergmann <arnd@arndb.de>
Date:   Thu Dec 15 17:46:19 2022 +0100

    hte: tegra: fix 'struct of_device_id' build error

    [ Upstream commit 6680c835ada1b34e882d0a32612f7294c62e27e0 ]

    Without the extra #include, this driver produces a build failure
    in some configurations.

    drivers/hte/hte-tegra194-test.c:96:34: error: array type has incomplete element type
'struct of_device_id'
       96 | static const struct of_device_id tegra_hte_test_of_match[] = {

    Fixes: 9a75a7cd03c9 ("hte: Add Tegra HTE test driver")
    Signed-off-by: Arnd Bergmann <arnd@arndb.de>
    Acked-by: Dipen Patel <dipenp@nvidia.com>
    Signed-off-by: Dipen Patel <dipenp@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b439c6c0191f23339c5497708cacf354a2bcf56c
```

```
Author: Charles Keepax <ckeepax@opensource.cirrus.com>
Date:   Thu Mar 23 13:41:38 2023 +0000

    mfd: arizona-spi: Add missing MODULE_DEVICE_TABLE

    [ Upstream commit 972c91fd7beddc3f19c8c855f6e60e7dbd435cbd ]

    This patch adds missing MODULE_DEVICE_TABLE definition
    which generates correct modalias for automatic loading
    of this driver when it is built as a module.

    Fixes: 3f65555c417c ("mfd: arizona: Split of_match table into I2C and SPI versions")
    Signed-off-by: Charles Keepax <ckeepax@opensource.cirrus.com>
    Signed-off-by: Lee Jones <lee@kernel.org>
    Link: https://lore.kernel.org/r/20230323134138.834369-1-ckeepax@opensource.cirrus.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ee146dad5c9639de80bd284dccadee822d67866f
Author: Colin Foster <colin.foster@in-advantage.com>
Date:   Wed Mar 22 07:11:30 2023 -0700

    mfd: ocelot-spi: Fix unsupported bulk read

    [ Upstream commit f0484d2f80a72022b7fac72bcb406392900ef1eb ]

    Ocelot chips (VSC7511, VSC7512, VSC7513, VSC7514) don't support bulk read
    operations over SPI.

    Many SPI buses have hardware that can optimize consecutive reads.
    Essentially an address is written to the chip, and if the SPI controller
    continues to toggle the clock, subsequent register values are reported.
    This can lead to significant optimizations, because the time between
    "address is written to the chip" and "chip starts to report data" can often
    take a fixed amount of time.

    When support for Ocelot chips were added in commit f3e893626abe ("mfd:
    ocelot: Add support for the vsc7512 chip via spi") it was believed that
    this optimization was supported. However it is not.

    Most register transactions with the Ocelot chips are not done in bulk, so
    this bug could go unnoticed. The one scenario where bulk register
    operations _are_ performed is when polling port statistics counters, which
    was added in commit d87b1c08f38a ("net: mscc: ocelot: use bulk reads for
    stats").

    Things get slightly more complicated here...

    A bug was introduced in commit d4c367650704 ("net: mscc: ocelot: keep
    ocelot_stat_layout by reg address, not offset") that broke the optimization
    of bulk reads. This means that when Ethernet support for the VSC7512 chip
    was added in commit 3d7316ac81ac ("net: dsa: ocelot: add external ocelot
    switch control") things were actually working "as expected".

    The bulk read opmtimization was discovered, and fixed in commit
    6acc72a43eac ("net: mscc: ocelot: fix stats region batching") and the
    timing optimizations for SPI were noticed. A bulk read went from ~14ms to
    ~2ms. But this timing improvement came at the cost of every register
    reading zero due the fact that bulk reads don't work.

    The read timings increase back to 13-14ms, but that's a price worth paying
    in order to receive valid data. This is verified in a DSA setup (cpsw-new
    switch tied to port 0 on the VSC7512, after having been running overnight)

         Rx Octets: 16222055 # Counters from CPSW switch
         Tx Octets: 12034702
```

```
          Net Octets: 28256757
          p00_rx_octets: 12034702 # Counters from Ocelot switch
          p00_rx_frames_below_65_octets: 0
          p00_rx_frames_65_to_127_octets: 88188
          p00_rx_frames_128_to_255_octets: 13
          p00_rx_frames_256_to_511_octets: 0
          p00_rx_frames_512_to_1023_octets: 0
          p00_rx_frames_over_1526_octets: 3306
          p00_tx_octets: 16222055

    Fixes: f3e893626abe ("mfd: ocelot: Add support for the vsc7512 chip via spi")
    Signed-off-by: Colin Foster <colin.foster@in-advantage.com>
    Signed-off-by: Lee Jones <lee@kernel.org>
    Link: https://lore.kernel.org/r/20230322141130.2531256-1-colin.foster@in-advantage.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 50d254f199d007284fc533e84649316df5b7e45c
Author: Matthias Schiffer <matthias.schiffer@ew.tq-group.com>
Date:   Mon Feb 20 12:25:46 2023 +0100

    mfd: tqmx86: Correct board names for TQMxE39x

    [ Upstream commit f376c479668557bcc2fd9e9fbc0f53e7819a11cd ]

    It seems that this driver was developed based on preliminary documentation.
    Report the correct names for all TQMxE39x variants, as they are used by
    the released hardware revisions:

    - Fix names for TQMxE39C1/C2 board IDs
    - Distinguish TQMxE39M and TQMxE39S, which use the same board ID

    The TQMxE39M/S are distinguished using the SAUC (Sanctioned Alternate
    Uses Configuration) register of the GPIO controller. This also prepares
    for the correct handling of the differences between the GPIO controllers
    of our COMe and SMARC modules.

    Fixes: 2f17dd34ffed ("mfd: tqmx86: IO controller with I2C, Wachdog and GPIO")
    Signed-off-by: Matthias Schiffer <matthias.schiffer@ew.tq-group.com>
    Reviewed-by: Andrew Lunn <andrew@lunn.ch>
    Signed-off-by: Lee Jones <lee@kernel.org>
    Link:
https://lore.kernel.org/r/aca9a7cb42a85181bcb456c437554d2728e708ec.1676892223.git.matthias.s
chiffer@ew.tq-group.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 411bd7df9bb7ea7ae816fcb9c419f672269713fe
Author: Matthias Schiffer <matthias.schiffer@ew.tq-group.com>
Date:   Mon Feb 20 12:25:45 2023 +0100

    mfd: tqmx86: Specify IO port register range more precisely

    [ Upstream commit 051c69ff4f607aa114c7bbdd7c41ed881367aeee ]

    Registers 0x160..0x17f are unassigned. Use 0x180 as base register and
    update offets accordingly.

    Also change the size of the range to include 0x19f. While 0x19f is
    currently reserved for future extensions, so are several of the previous
    registers up to 0x19e, and it is weird to leave out just the last one.

    Fixes: 2f17dd34ffed ("mfd: tqmx86: IO controller with I2C, Wachdog and GPIO")
    Signed-off-by: Matthias Schiffer <matthias.schiffer@ew.tq-group.com>
    Reviewed-by: Andrew Lunn <andrew@lunn.ch>
    Signed-off-by: Lee Jones <lee@kernel.org>
    Link:
```

```
https://lore.kernel.org/r/db4677ac318b1283c8956f637f409995a30a31c3.1676892223.git.matthias.s
chiffer@ew.tq-group.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 08dcb9224a3ff106bf6bf8ff6eae402a1aad676e
Author: Matthias Schiffer <matthias.schiffer@ew.tq-group.com>
Date:   Mon Feb 20 12:25:44 2023 +0100

    mfd: tqmx86: Do not access I2C_DETECT register through io_base

    [ Upstream commit 1be1b23696b3d4b0231c694f5e0767b4471d33a9 ]

    The I2C_DETECT register is at IO port 0x1a7, which is outside the range
    passed to devm_ioport_map() for io_base, and was only working because
    there aren't actually any bounds checks for IO port accesses.

    Extending the range does not seem like a good solution here, as it would
    then conflict with the IO resource assigned to the I2C controller. As
    this is just a one-off access during probe, use a simple inb() instead.

    While we're at it, drop the unused define TQMX86_REG_I2C_INT_EN.

    Fixes: 2f17dd34ffed ("mfd: tqmx86: IO controller with I2C, Wachdog and GPIO")
    Signed-off-by: Matthias Schiffer <matthias.schiffer@ew.tq-group.com>
    Reviewed-by: Andrew Lunn <andrew@lunn.ch>
    Signed-off-by: Lee Jones <lee@kernel.org>
    Link:
https://lore.kernel.org/r/e8300a30f0791afb67d79db8089fb6004855f378.1676892223.git.matthias.s
chiffer@ew.tq-group.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e83b320b9e914ddf51fdf2374034113b872a49d7
Author: Kang Chen <void0red@hust.edu.cn>
Date:   Wed Apr 19 10:07:48 2023 +0800

    thermal/drivers/mediatek: Use devm_of_iomap to avoid resource leak in mtk_thermal_probe

    [ Upstream commit f05c7b7d9ea9477fcc388476c6f4ade8c66d2d26 ]

    Smatch reports:
    1. mtk_thermal_probe() warn: 'apmixed_base' from of_iomap() not released.
    2. mtk_thermal_probe() warn: 'auxadc_base' from of_iomap() not released.

    The original code forgets to release iomap resource when handling errors,
    fix it by switch to devm_of_iomap.

    Fixes: 89945047b166 ("thermal: mediatek: Add tsensor support for V2 thermal system")
    Signed-off-by: Kang Chen <void0red@hust.edu.cn>
    Reviewed-by: Dongliang Mu <dzm91@hust.edu.cn>
    Reviewed-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Signed-off-by: Daniel Lezcano <daniel.lezcano@linaro.org>
    Link: https://lore.kernel.org/r/20230419020749.621257-1-void0red@hust.edu.cn
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 032c5d8fff29d1205c0fd51f14d0528ff34b7684
Author: Hans Verkuil <hverkuil-cisco@xs4all.nl>
Date:   Thu Apr 20 14:47:05 2023 +0200

    pinctrl-bcm2835.c: fix race condition when setting gpio dir

    [ Upstream commit b7badd752de05312fdb1aeb388480f706d0c087f ]

    In the past setting the pin direction called pinctrl_gpio_direction()
    which uses a mutex to serialize this. That was changed to set the
    direction directly in the pin controller driver, but that lost the
```

```
    serialization mechanism. Since the direction of multiple pins are in
    the same register you can have a race condition, something that was
    in fact observed with the cec-gpio driver.

    Add a new spinlock to serialize writing to the FSEL registers.

    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Fixes: 1a4541b68e25 ("pinctrl-bcm2835: don't call pinctrl_gpio_direction()")
    Link: https://lore.kernel.org/r/4302b66b-ca20-0f19-d2aa-ee8661118863@xs4all.nl
    Signed-off-by: Linus Walleij <linus.walleij@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 54e90e750a2823abe59a1ff35f1b3d83d932c2b4
Author: Claudiu Beznea <claudiu.beznea@microchip.com>
Date:   Tue Feb 14 17:18:25 2023 +0200

    dmaengine: at_xdmac: do not enable all cyclic channels

    [ Upstream commit f8435befd81dd85b7b610598551fadf675849bc1 ]

    Do not global enable all the cyclic channels in at_xdmac_resume(). Instead
    save the global status in at_xdmac_suspend() and re-enable the cyclic
    channel only if it was active before suspend.

    Fixes: e1f7c9eee707 ("dmaengine: at_xdmac: creation of the atmel eXtended DMA Controller
driver")
    Signed-off-by: Claudiu Beznea <claudiu.beznea@microchip.com>
    Link: https://lore.kernel.org/r/20230214151827.1050280-6-claudiu.beznea@microchip.com
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cb2ab1985f27ec0c42b98e8f3efd36b881803058
Author: Claudiu Beznea <claudiu.beznea@microchip.com>
Date:   Tue Feb 14 17:18:24 2023 +0200

    dmaengine: at_xdmac: restore the content of grws register

    [ Upstream commit 7c5eb63d16b01c202aaa95f374ae15a807745a73 ]

    In case the system suspends to a deep sleep state where power to DMA
    controller is cut-off we need to restore the content of GRWS register.
    This is a write only register and writing bit X tells the controller
    to suspend read and write requests for channel X. Thus set GRWS before
    restoring the content of GE (Global Enable) regiter.

    Fixes: e1f7c9eee707 ("dmaengine: at_xdmac: creation of the atmel eXtended DMA Controller
driver")
    Signed-off-by: Claudiu Beznea <claudiu.beznea@microchip.com>
    Link: https://lore.kernel.org/r/20230214151827.1050280-5-claudiu.beznea@microchip.com
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b4e66eb1bb5cf577932693d585afaa672095acf9
Author: Claudiu Beznea <claudiu.beznea@microchip.com>
Date:   Tue Feb 14 17:18:23 2023 +0200

    dmaengine: at_xdmac: do not resume channels paused by consumers

    [ Upstream commit 44fe8440bda545b5d167329df88c47609a645168 ]

    In case there are DMA channels not paused by consumers in suspend
    process (valid on AT91 SoCs for serial driver when no_console_suspend) the
    driver pauses them (using at_xdmac_device_pause() which is also the same
    function called by dmaengine_pause()) and then in the resume process the
    driver resumes them calling at_xdmac_device_resume() which is the same
```

```
    function called by dmaengine_resume()). This is good for DMA channels
    not paused by consumers but for drivers that calls
    dmaengine_pause()/dmaegine_resume() on suspend/resume path this may lead to
    DMA channel being enabled before the IP is enabled. For IPs that needs
    strict ordering with regards to DMA channel enablement this will lead to
    wrong behavior. To fix this add a new set of functions
    at_xdmac_device_pause_internal()/at_xdmac_device_resume_internal() to be
    called only on suspend/resume.

    Fixes: e1f7c9eee707 ("dmaengine: at_xdmac: creation of the atmel eXtended DMA Controller
driver")
    Signed-off-by: Claudiu Beznea <claudiu.beznea@microchip.com>
    Link: https://lore.kernel.org/r/20230214151827.1050280-4-claudiu.beznea@microchip.com
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1ac6270cb66fed8be733b1500f7ea7cb3c1c25b5
Author: Claudiu Beznea <claudiu.beznea@microchip.com>
Date:   Tue Feb 14 17:18:22 2023 +0200

    dmaengine: at_xdmac: fix imbalanced runtime PM reference counter

    [ Upstream commit e53957e1ec5196671e49a48f90a5c9555153189a ]

    In case there are channels not paused during suspend (which on AT91 case
    is valid for serial driver when no_console_suspend boot argument is used)
    the at_xdmac_runtime_suspend_descriptors() was called more than
    one time due to at_xdmac_off(). To fix this add a new argument to
    at_xdmac_off() to specify if runtime PM reference counter needs to be
    decremented for queued active descriptors. Along with it moved the
    at_xdmac_runtime_suspend_descriptors() call under at_xdmac_chan_is_paused()
    check on suspend path as for the rest of channels the suspend is delayed
    by atmel_xdmac_prepare() in case channel is enabled. Same approach has
    been applied on resume path.

    Fixes: 650b0e990cbd ("dmaengine: at_xdmac: add runtime pm support")
    Signed-off-by: Claudiu Beznea <claudiu.beznea@microchip.com>
    Link: https://lore.kernel.org/r/20230214151827.1050280-3-claudiu.beznea@microchip.com
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cd6b8883027a8aae69de5ba7bc0bf6491c34b143
Author: Claudiu Beznea <claudiu.beznea@microchip.com>
Date:   Tue Feb 14 17:18:21 2023 +0200

    dmaengine: at_xdmac: disable/enable clock directly on suspend/resume

    [ Upstream commit 2de5ddb5e68c94b781b3789bca1ce52000d7d0e0 ]

    Runtime PM APIs for at_xdmac just plays with clk_enable()/clk_disable()
    letting aside the clk_prepare()/clk_unprepare() that needs to be
    executed as the clock is also prepared on probe. Thus instead of using
    runtime PM force suspend/resume APIs use
    clk_disable_unprepare() + pm_runtime_put_noidle() on suspend and
    clk_prepare_enable() + pm_runtime_get_noresume() on resume. This
    approach as been chosen instead of using runtime PM force suspend/resume
    with clk_unprepare()/clk_prepare() as it looks simpler and the final
    code is better.

    While at it added the missing pm_runtime_mark_last_busy() on suspend before
    decrementing the reference counter.

    Fixes: 650b0e990cbd ("dmaengine: at_xdmac: add runtime pm support")
    Signed-off-by: Claudiu Beznea <claudiu.beznea@microchip.com>
    Link: https://lore.kernel.org/r/20230214151827.1050280-2-claudiu.beznea@microchip.com
```

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

commit 37c0f448c5a26fde36d4bc485b967789b1f6fbc4
Author: Shunsuke Mie <mie@igel.co.jp>
Date:   Tue Apr 11 19:17:58 2023 +0900

    dmaengine: dw-edma: Fix to enable to issue dma request on DMA processing

    [ Upstream commit 970b17dfe264a9085ba4e593730ecfd496b950ab ]

    The issue_pending request is ignored while driver is processing a DMA
    request. Fix to issue the pending requests on any dma channel status.

    Fixes: e63d79d1ffcd ("dmaengine: Add Synopsys eDMA IP core driver")
    Signed-off-by: Shunsuke Mie <mie@igel.co.jp>
    Link: https://lore.kernel.org/r/20230411101758.438472-2-mie@igel.co.jp
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3e3a1e42b92deb2e592e2da37d6214887e063888
Author: Shunsuke Mie <mie@igel.co.jp>
Date:   Tue Apr 11 19:17:57 2023 +0900

    dmaengine: dw-edma: Fix to change for continuous transfer

    [ Upstream commit a251994a441ee0a69ba7062c8cd2d08ead3db379 ]

    The dw-edma driver stops after processing a DMA request even if a request
    remains in the issued queue, which is not the expected behavior. The DMA
    engine API requires continuous processing.

    Add a trigger to start after one processing finished if there are requests
    remain.

    Fixes: e63d79d1ffcd ("dmaengine: Add Synopsys eDMA IP core driver")
    Signed-off-by: Shunsuke Mie <mie@igel.co.jp>
    Link: https://lore.kernel.org/r/20230411101758.438472-1-mie@igel.co.jp
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit adfa452f166602b25e55888e1908eb74aac58755
Author: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
Date:   Mon Apr 10 02:33:55 2023 +0300

    dma: gpi: remove spurious unlock in gpi_ch_init

    [ Upstream commit 91d6a468e335571f1e67e046050dea9af5fa4ebe ]

    gpi_ch_init() doesn't lock the ctrl_lock mutex, so there is no need to
    unlock it too. Instead the mutex is handled by the function
    gpi_alloc_chan_resources(), which properly locks and unlocks the mutex.

    ====================================
    WARNING: bad unlock balance detected!
    6.3.0-rc5-00253-g99792582ded1-dirty #15 Not tainted
    ------------------------------------
    kworker/u16:0/9 is trying to release lock (&gpii->ctrl_lock) at:
    [<ffffb99d04e1284c>] gpi_alloc_chan_resources+0x108/0x5bc
    but there are no more locks to release!

    other info that might help us debug this:
    6 locks held by kworker/u16:0/9:
     #0: ffff575740010938 ((wq_completion)events_unbound){+.+.}-{0:0}, at:
process_one_work+0x220/0x594
```

```
       #1: ffff80000809bdd0 (deferred_probe_work){+.+.}-{0:0}, at:
  process_one_work+0x220/0x594
       #2: ffff575740f2a0f8 (&dev->mutex){....}-{3:3}, at: __device_attach+0x38/0x188
       #3: ffff57574b5570f8 (&dev->mutex){....}-{3:3}, at: __device_attach+0x38/0x188
       #4: ffffb99d06a2f180 (of_dma_lock){+.+.}-{3:3}, at:
  of_dma_request_slave_channel+0x138/0x280
       #5: ffffb99d06a2ee20 (dma_list_mutex){+.+.}-{3:3}, at: dma_get_slave_channel+0x28/0x10c

      stack backtrace:
      CPU: 7 PID: 9 Comm: kworker/u16:0 Not tainted 6.3.0-rc5-00253-g99792582ded1-dirty #15
      Hardware name: Google Pixel 3 (DT)
      Workqueue: events_unbound deferred_probe_work_func
      Call trace:
       dump_backtrace+0xa0/0xfc
       show_stack+0x18/0x24
       dump_stack_lvl+0x60/0xac
       dump_stack+0x18/0x24
       print_unlock_imbalance_bug+0x130/0x148
       lock_release+0x270/0x300
       __mutex_unlock_slowpath+0x48/0x2cc
       mutex_unlock+0x20/0x2c
       gpi_alloc_chan_resources+0x108/0x5bc
       dma_chan_get+0x84/0x188
       dma_get_slave_channel+0x5c/0x10c
       gpi_of_dma_xlate+0x110/0x1a0
       of_dma_request_slave_channel+0x174/0x280
       dma_request_chan+0x3c/0x2d4
       geni_i2c_probe+0x544/0x63c
       platform_probe+0x68/0xc4
       really_probe+0x148/0x2ac
       __driver_probe_device+0x78/0xe0
       driver_probe_device+0x3c/0x160
       __device_attach_driver+0xb8/0x138
       bus_for_each_drv+0x84/0xe0
       __device_attach+0x9c/0x188
       device_initial_probe+0x14/0x20
       bus_probe_device+0xac/0xb0
       device_add+0x60c/0x7d8
       of_device_add+0x44/0x60
       of_platform_device_create_pdata+0x90/0x124
       of_platform_bus_create+0x15c/0x3c8
       of_platform_populate+0x58/0xf8
       devm_of_platform_populate+0x58/0xbc
       geni_se_probe+0xf0/0x164
       platform_probe+0x68/0xc4
       really_probe+0x148/0x2ac
       __driver_probe_device+0x78/0xe0
       driver_probe_device+0x3c/0x160
       __device_attach_driver+0xb8/0x138
       bus_for_each_drv+0x84/0xe0
       __device_attach+0x9c/0x188
       device_initial_probe+0x14/0x20
       bus_probe_device+0xac/0xb0
       deferred_probe_work_func+0x8c/0xc8
       process_one_work+0x2bc/0x594
       worker_thread+0x228/0x438
       kthread+0x108/0x10c
       ret_from_fork+0x10/0x20

      Fixes: 5d0c3533a19f ("dmaengine: qcom: Add GPI dma driver")
      Signed-off-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
      Link: https://lore.kernel.org/r/20230409233355.453741-1-dmitry.baryshkov@linaro.org
      Signed-off-by: Vinod Koul <vkoul@kernel.org>
      Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit cfece3b5f4ea9ecbaba689b95b72052731abe813
Author: Siddharth Vadapalli <s-vadapalli@ti.com>
Date:   Mon Apr 3 15:15:52 2023 +0530

    phy: ti: j721e-wiz: Fix unreachable code in wiz_mode_select()

    [ Upstream commit 57c0e1362fdd57d0cea7ab1e583b58abf4bd8c2d ]

    In the wiz_mode_select() function, the configuration performed for
    PHY_TYPE_USXGMII is unreachable. Fix it.

    Fixes: b64a85fb8f53 ("phy: ti: phy-j721e-wiz.c: Add usxgmii support in wiz driver")
    Signed-off-by: Siddharth Vadapalli <s-vadapalli@ti.com>
    Reviewed-by: Roger Quadros <rogerq@kernel.org>
    Link: https://lore.kernel.org/r/20230403094552.929108-1-s-vadapalli@ti.com
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9b33e47799cc688248ea4213f30c3b11f02acda3
Author: Gaosheng Cui <cuigaosheng1@huawei.com>
Date:   Tue Nov 29 19:16:34 2022 +0800

    phy: tegra: xusb: Add missing tegra_xusb_port_unregister for usb2_port and ulpi_port

    [ Upstream commit e024854048e733391b31fe5a398704b31b9af803 ]

    The tegra_xusb_port_unregister should be called when usb2_port
    and ulpi_port map fails in tegra_xusb_add_usb2_port() or in
    tegra_xusb_add_ulpi_port(), fix it.

    Fixes: 53d2a715c240 ("phy: Add Tegra XUSB pad controller support")
    Signed-off-by: Gaosheng Cui <cuigaosheng1@huawei.com>
    Acked-by: Thierry Reding <treding@nvidia.com>
    Link: https://lore.kernel.org/r/20221129111634.1547747-1-cuigaosheng1@huawei.com
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4ddae1bef783a01dc099ec1dbeee0560bbe15f4a
Author: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com>
Date:   Tue Mar 21 10:26:42 2023 +0800

    soundwire: intel: don't save hw_params for use in prepare

    [ Upstream commit 0a0d1740bd8fd7dafb81fcb102fb5d0b83b1ce73 ]

    The existing code copies the hw_params pointer and reuses it later in
    .prepare, specifically to re-initialize the ALH DMA channel
    information that's lost in suspend-resume cycles.

    This is not needed, we can directly access the information from the
    substream/rtd - as done for the HDAudio DAIs in
    sound/soc/sof/intel/hda-dai.c

    In addition, using the saved pointer causes the suspend-resume test
    cases to fail on specific platforms, depending on which version of GCC
    is used. Péter Ujfalusi and I have spent long hours to root-cause this
    problem that was reported by the Intel CI first with 6.2-rc1 and again
    v6.3-rc1. In the latter case we were lucky that the problem was 100%
    reproducible on local test devices, and found out that adding a
    dev_dbg() or adding a call to usleep_range() just before accessing the
    saved pointer "fixed" the issue. With errors appearing just by
    changing the compiler version or minor changes in the code generated,
    clearly we have a memory management Heisenbug.

    The root-cause seems to be that the hw_params pointer is not
```

```
    persistent. The soc-pcm code allocates the hw_params structure on the
    stack, and passes it to the BE dailink hw_params and DAIs
    hw_params. Saving such a pointer and reusing it later during the
    .prepare stage cannot possibly work reliably, it's broken-by-design
    since v5.10. It's astonishing that the problem was not seen earlier.

    This simple fix will have to be back-ported to -stable, due to changes
    to avoid the use of the get/set_dmadata routines this patch will only
    apply on kernels older than v6.1.

    Fixes: a5a0239c27fe ("soundwire: intel: reinitialize IP+DSP in .prepare(), but only when
resuming")
    Signed-off-by: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com>
    Reviewed-by: Ranjani Sridharan <ranjani.sridharan@linux.intel.com>
    Reviewed-by: Péter Ujfalusi <peter.ujfalusi@linux.intel.com>
    Signed-off-by: Bard Liao <yung-chuan.liao@linux.intel.com>
    Link: https://lore.kernel.org/r/20230321022642.1426611-1-yung-chuan.liao@linux.intel.com
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9dbc30c861bd98971502d663f0e5d139468d5377
Author: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
Date:   Mon Apr 3 15:30:54 2023 +0200

    pwm: mtk-disp: Configure double buffering before reading in .get_state()

    [ Upstream commit b16c310115f2084b8826a35b77ef42bab6786d9f ]

    The DISP_PWM controller's default behavior is to always use register
    double buffering: all reads/writes are then performed on shadow
    registers instead of working registers and this becomes an issue
    in case our chosen configuration in Linux is different from the
    default (or from the one that was pre-applied by the bootloader).

    An example of broken behavior is when the controller is configured
    to use shadow registers, but this driver wants to configure it
    otherwise: what happens is that the .get_state() callback is called
    right after registering the pwmchip and checks whether the PWM is
    enabled by reading the DISP_PWM_EN register;
    At this point, if shadow registers are enabled but their content
    was not committed before booting Linux, we are *not* reading the
    current PWM enablement status, leading to the kernel knowing that
    the hardware is actually enabled when, in reality, it's not.

    The aforementioned issue emerged since this driver was fixed with
    commit 0b5ef3429d8f ("pwm: mtk-disp: Fix the parameters calculated
    by the enabled flag of disp_pwm") making it to read the enablement
    status from the right register.

    Configure the controller in the .get_state() callback to avoid
    this desync issue and get the backlight properly working again.

    Fixes: 3f2b16734914 ("pwm: mtk-disp: Implement atomic API .get_state()")
    Signed-off-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Reviewed-by: Nícolas F. R. A. Prado <nfraprado@collabora.com>
    Tested-by: Nícolas F. R. A. Prado <nfraprado@collabora.com>
    Reviewed-by: Alexandre Mergnat <amergnat@baylibre.com>
    Tested-by: Alexandre Mergnat <amergnat@baylibre.com>
    Signed-off-by: Thierry Reding <thierry.reding@gmail.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 09531bbc8e1b4cceb34eff394fb2f796510e571f
Author: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
Date:   Mon Apr 3 15:30:53 2023 +0200
```

```
    pwm: mtk-disp: Disable shadow registers before setting backlight values

    [ Upstream commit 36dd7f530ae7d9ce9e853ffb8aa337de65c6600b ]

    If shadow registers usage is not desired, disable that before performing
    any write to CON0/1 registers in the .apply() callback, otherwise we may
    lose clkdiv or period/width updates.

    Fixes: cd4b45ac449a ("pwm: Add MediaTek MT2701 display PWM driver support")
    Signed-off-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Reviewed-by: Nícolas F. R. A. Prado <nfraprado@collabora.com>
    Tested-by: Nícolas F. R. A. Prado <nfraprado@collabora.com>
    Reviewed-by: Alexandre Mergnat <amergnat@baylibre.com>
    Tested-by: Alexandre Mergnat <amergnat@baylibre.com>
    Signed-off-by: Thierry Reding <thierry.reding@gmail.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 671b52c1507b1846c8e4f253fd55745be198f97a
Author: H. Nikolaus Schaller <hns@goldelico.com>
Date:   Sun Apr 2 13:12:59 2023 +0200

    leds: tca6507: Fix error handling of using fwnode_property_read_string

    [ Upstream commit c1087c29e96a48e9080377e168d35dcb52fb068b ]

    Commit 96f524105b9c ("leds: tca6507: use fwnode API instead of OF")

    changed to fwnode API but did not take into account that a missing property
    "linux,default-trigger" now seems to return an error and as a side effect
    sets value to -1. This seems to be different from of_get_property() which
    always returned NULL in any case of error.

    Neglecting this side-effect leads to

    [   11.201965] Unable to handle kernel paging request at virtual address ffffffff when
read

    in the strcmp() of led_trigger_set_default() if there is no led-trigger
    defined in the DTS.

    I don't know if this was recently introduced somewhere in the fwnode lib
    or if the effect was missed in initial testing. Anyways it seems to be a
    bug to ignore the error return value of an optional value here in the
    driver.

    Fixes: 96f524105b9c ("leds: tca6507: use fwnode API instead of OF")
    Signed-off-by: H. Nikolaus Schaller <hns@goldelico.com>
    Acked-by: Pavel Machek <pavel@ucw.cz>
    Reviewed-by: Marek Behún <kabel@kernel.org>
    Signed-off-by: Lee Jones <lee@kernel.org>
    Link:
https://lore.kernel.org/r/cbae7617db83113de726fcc423a805ebaa1bfca6.1680433978.git.hns@goldel
ico.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit caa413a557da59740284829180d93dce57d4656c
Author: Christophe JAILLET <christophe.jaillet@wanadoo.fr>
Date:   Sun Mar 26 09:06:37 2023 +0200

    dmaengine: mv_xor_v2: Fix an error code.

    [ Upstream commit 827026ae2e56ec05ef1155661079badbbfc0b038 ]

    If the probe is deferred, -EPROBE_DEFER should be returned, not
    +EPROBE_DEFER.
```

```
    Fixes: 3cd2c313f1d6 ("dmaengine: mv_xor_v2: Fix clock resource by adding a register
clock")
    Signed-off-by: Christophe JAILLET <christophe.jaillet@wanadoo.fr>
    Link:
https://lore.kernel.org/r/201170dff832a3c496d125772e10070cd834ebf2.1679814350.git.christophe
.jaillet@wanadoo.fr
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 32a4875db7b91a868339933697afcb7cbe9b9556
Author: Arınç ÜNAL <arinc.unal@arinc9.com>
Date:   Sat Mar 18 00:29:51 2023 +0300

    pinctrl: ralink: reintroduce ralink,rt2880-pinmux compatible string

    [ Upstream commit 7c19147d9cfc0f9328049d2e278279150d7de9ca ]

    There have been stable releases with the ralink,rt2880-pinmux compatible
    string included. Having it removed breaks the ABI. Reintroduce it.

    Fixes: e5981cd46183 ("pinctrl: ralink: add new compatible strings for each pinctrl
subdriver")
    Signed-off-by: Arınç ÜNAL <arinc.unal@arinc9.com>
    Reviewed-by: Sergio Paracuellos <sergio.paracuellos@gmail.com>
    Link: https://lore.kernel.org/r/20230317213011.13656-2-arinc.unal@arinc9.com
    Signed-off-by: Linus Walleij <linus.walleij@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c8d1ebfd8ce28dee5bce99e8b3c838b64ad693bd
Author: Randy Dunlap <rdunlap@infradead.org>
Date:   Sat Feb 25 21:39:49 2023 -0800

    leds: TI_LMU_COMMON: select REGMAP instead of depending on it

    [ Upstream commit a61079efc87888587e463afaed82417b162fbd69 ]

    REGMAP is a hidden (not user visible) symbol. Users cannot set it
    directly thru "make *config", so drivers should select it instead of
    depending on it if they need it.

    Consistently using "select" or "depends on" can also help reduce
    Kconfig circular dependency issues.

    Therefore, change the use of "depends on REGMAP" to "select REGMAP".

    Fixes: 3fce8e1eb994 ("leds: TI LMU: Add common code for TI LMU devices")
    Signed-off-by: Randy Dunlap <rdunlap@infradead.org>
    Acked-by: Pavel Machek <pavel@ucw.cz>
    Signed-off-by: Lee Jones <lee@kernel.org>
    Link: https://lore.kernel.org/r/20230226053953.4681-5-rdunlap@infradead.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 38c40706f02c7ef391dd94ec45fdb5324827dc22
Author: Geert Uytterhoeven <geert+renesas@glider.be>
Date:   Mon Nov 21 14:18:39 2022 +0100

    pinctrl: renesas: r8a779g0: Fix ERROROUTC function names

    [ Upstream commit 9da805344d2a2d27a32cc9271230055d8818e887 ]

    According to R-Car V4H Series User’s Manual: Hardware Rev. 0.54, the
    ERROROUTC signal is active-low.  Hence add the missing "_N" suffix to
    the pin function's names.
```

```
      Resize column 2 of all IPxSR* definitions to accomodate the longer
      names.

      Fixes: b811062e5fd0343c ("pinctrl: renesas: r8a779g0: Add missing ERROROUTC_A")
      Fixes: ad9bb2fec66262b0 ("pinctrl: renesas: Initial R8A779G0 (R-Car V4H) PFC support")
      Signed-off-by: Geert Uytterhoeven <geert+renesas@glider.be>
      Link:
  https://lore.kernel.org/r/1774303989e7d61f08fa81f1c2fa1b394505645f.1669036423.git.geert+rene
  sas@glider.be
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 2edf34339e276f55a9279dbbe1bc78c5c3072126
  Author: Geert Uytterhoeven <geert+renesas@glider.be>
  Date:   Mon Nov 21 14:18:38 2022 +0100

      pinctrl: renesas: r8a779g0: Fix Group 6/7 pin functions

      [ Upstream commit 203734a0419cade9c76016f66e2c7ba354c249b4 ]

      According to R-Car V4H Series User's Manual: Hardware Rev. 0.54, pin
      groups 6 and 7 do not use Module Select Registers to configure pin
      functions.

      Hence:
        - Remove the non-existent Module Select Registers (MODSEL[67]),
        - Correct the affected PINMUX definitions.

      Fixes: 36611d28f5130d8b ("pinctrl: renesas: r8a779g0: Add missing MODSELx for AVBx")
      Fixes: ad9bb2fec66262b0 ("pinctrl: renesas: Initial R8A779G0 (R-Car V4H) PFC support")
      Signed-off-by: Geert Uytterhoeven <geert+renesas@glider.be>
      Link:
  https://lore.kernel.org/r/06972cafd0efa4cfb395cfa76000a1bdae5e9e73.1669036423.git.geert+rene
  sas@glider.be
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 24d351ec03c35c5f44d43a3b16cf627558d5cdf4
  Author: Geert Uytterhoeven <geert+renesas@glider.be>
  Date:   Mon Nov 21 14:18:37 2022 +0100

      pinctrl: renesas: r8a779g0: Fix Group 4/5 pin functions

      [ Upstream commit 0a7a5226e7b177c68800985a19a80c1df9bceff6 ]

      According to R-Car V4H Series User's Manual: Hardware Rev. 0.54, pin
      groups 4 and 5 do not use Module Select Registers to configure pin
      functions, but use Peripheral Function Select Registers instead.

      Hence:
        - Remove the non-existent Module Select Registers (MODSEL[45]),
        - Add the missing Peripheral Function Select Registers (IPxSR[45]),
        - Correct the GPIO / Peripheral Function Select Register definitions
          (GPSR]45_*),
        - Correct the affected PINMUX definitions.

      Fixes: 36611d28f5130d8b ("pinctrl: renesas: r8a779g0: Add missing MODSELx for AVBx")
      Fixes: 36fb7b8af55b83e0 ("pinctrl: renesas: r8a779g0: Add missing MODSELx for TSN0")
      Fixes: ad9bb2fec66262b0 ("pinctrl: renesas: Initial R8A779G0 (R-Car V4H) PFC support")
      Signed-off-by: Geert Uytterhoeven <geert+renesas@glider.be>
      Link:
  https://lore.kernel.org/r/3d3833d1738f5e8fcc4c1002aa93832464d129a0.1669036423.git.geert+rene
  sas@glider.be
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 97afc87c6437347a12872cc5d426b20d9d1bf6ad
  Author: Phong Hoang <phong.hoang.wz@renesas.com>
```

```
Date:   Fri Jan 27 14:11:59 2023 +0100

    pinctrl: renesas: r8a779f0: Fix tsn1_avtp_pps pin group

    [ Upstream commit 60003351e99167d8cfa7c161e95856efc016f381 ]

    Correct a typo mistake in the definition of the tsn1_avtp_pps pin group
    mux.

    Signed-off-by: Phong Hoang <phong.hoang.wz@renesas.com>
    Fixes: babe298e9caaa3d7 ("pinctrl: renesas: r8a779f0: Add Ethernet pins, groups, and
functions")
    Signed-off-by: Geert Uytterhoeven <geert+renesas@glider.be>
    Link:
https://lore.kernel.org/r/45ea6e87b91c36fd0b9706cf58ff50a4d1a99c44.1674825039.git.geert+rene
sas@glider.be
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 19783e201f691c4565fc45781d4c70bf7dce39ac
Author: Hai Pham <hai.pham.ud@renesas.com>
Date:   Fri Jan 27 14:10:31 2023 +0100

    pinctrl: renesas: r8a779a0: Remove incorrect AVB[01] pinmux configuration

    [ Upstream commit a145c9a8674ac8fbfa1595276e1b6cbfc5139038 ]

    AVB[01]_{MAGIC,MDC,MDIO,TXCREFCLK} are registered as both
    PINMUX_SINGLE(fn) and PINMUX_IPSR_GPSR(fn) in the pinmux_data array.

    The latter are correct, hence remove the former.
    Without this fix, the Ethernet PHY is not operational on the MDIO bus.

    Signed-off-by: Hai Pham <hai.pham.ud@renesas.com>
    Signed-off-by: LUU HOAI <hoai.luu.ub@renesas.com>
    Fixes: 741a7370fc3b8b54 ("pinctrl: renesas: Initial R8A779A0 (V3U) PFC support")
    Signed-off-by: Geert Uytterhoeven <geert+renesas@glider.be>
    Link:
https://lore.kernel.org/r/6fd217b71e83ba9a8157513ed671a1fa218b23b6.1674824958.git.geert+rene
sas@glider.be
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 14da044725a3ab10affa3566d29c15737c0e67a4
Author: Ye Bin <yebin10@huawei.com>
Date:   Thu Apr 6 11:16:27 2023 +0000

    ext4: fix use-after-free read in ext4_find_extent for bigalloc + inline

    [ Upstream commit 835659598c67907b98cd2aa57bb951dfaf675c69 ]

    Syzbot found the following issue:
    loop0: detected capacity change from 0 to 2048
    EXT4-fs (loop0): mounted filesystem 00000000-0000-0000-0000-000000000000 without
journal. Quota mode: none.
    ==================================================================
    BUG: KASAN: use-after-free in ext4_ext_binsearch_idx fs/ext4/extents.c:768 [inline]
    BUG: KASAN: use-after-free in ext4_find_extent+0x76e/0xd90 fs/ext4/extents.c:931
    Read of size 4 at addr ffff888073644750 by task syz-executor420/5067

    CPU: 0 PID: 5067 Comm: syz-executor420 Not tainted 6.2.0-rc1-syzkaller #0
    Hardware name: Google Google Compute Engine/Google Compute Engine, BIOS Google
10/26/2022
    Call Trace:
     <TASK>
     __dump_stack lib/dump_stack.c:88 [inline]
     dump_stack_lvl+0x1b1/0x290 lib/dump_stack.c:106
```

```
      print_address_description+0x74/0x340 mm/kasan/report.c:306
      print_report+0x107/0x1f0 mm/kasan/report.c:417
      kasan_report+0xcd/0x100 mm/kasan/report.c:517
      ext4_ext_binsearch_idx fs/ext4/extents.c:768 [inline]
      ext4_find_extent+0x76e/0xd90 fs/ext4/extents.c:931
      ext4_clu_mapped+0x117/0x970 fs/ext4/extents.c:5809
      ext4_insert_delayed_block fs/ext4/inode.c:1696 [inline]
      ext4_da_map_blocks fs/ext4/inode.c:1806 [inline]
      ext4_da_get_block_prep+0x9e8/0x13c0 fs/ext4/inode.c:1870
      ext4_block_write_begin+0x6a8/0x2290 fs/ext4/inode.c:1098
      ext4_da_write_begin+0x539/0x760 fs/ext4/inode.c:3082
      generic_perform_write+0x2e4/0x5e0 mm/filemap.c:3772
      ext4_buffered_write_iter+0x122/0x3a0 fs/ext4/file.c:285
      ext4_file_write_iter+0x1d0/0x18f0
      call_write_iter include/linux/fs.h:2186 [inline]
      new_sync_write fs/read_write.c:491 [inline]
      vfs_write+0x7dc/0xc50 fs/read_write.c:584
      ksys_write+0x177/0x2a0 fs/read_write.c:637
      do_syscall_x64 arch/x86/entry/common.c:50 [inline]
      do_syscall_64+0x3d/0xb0 arch/x86/entry/common.c:80
      entry_SYSCALL_64_after_hwframe+0x63/0xcd
    RIP: 0033:0x7f4b7a9737b9
    RSP: 002b:00007ffc5cac3668 EFLAGS: 00000246 ORIG_RAX: 0000000000000001
    RAX: ffffffffffffffda RBX: 0000000000000000 RCX: 00007f4b7a9737b9
    RDX: 00000000175d9003 RSI: 0000000020000200 RDI: 0000000000000004
    RBP: 00007f4b7a933050 R08: 0000000000000000 R09: 0000000000000000
    R10: 000000000000079f R11: 0000000000000246 R12: 00007f4b7a9330e0
    R13: 0000000000000000 R14: 0000000000000000 R15: 0000000000000000
     </TASK>

    Above issue is happens when enable bigalloc and inline data feature. As
    commit 131294c35ed6 fixed delayed allocation bug in ext4_clu_mapped for
    bigalloc + inline. But it only resolved issue when has inline data, if
    inline data has been converted to extent(ext4_da_convert_inline_data_to_extent)
    before writepages, there is no EXT4_STATE_MAY_INLINE_DATA flag. However
    i_data is still store inline data in this scene. Then will trigger UAF
    when find extent.
    To resolve above issue, there is need to add judge "ext4_has_inline_data(inode)"
    in ext4_clu_mapped().

    Fixes: 131294c35ed6 ("ext4: fix delayed allocation bug in ext4_clu_mapped for bigalloc +
inline")
    Reported-by: syzbot+bf4bb7731ef73b83a3b4@syzkaller.appspotmail.com
    Reviewed-by: Jan Kara <jack@suse.cz>
    Reviewed-by: Ye Bin <yebin10@huawei.com>
    Reviewed-by: Tudor Ambarus <tudor.ambarus@linaro.org>
    Tested-by: Tudor Ambarus <tudor.ambarus@linaro.org>
    Link: https://lore.kernel.org/r/20230406111627.1916759-1-tudor.ambarus@linaro.org
    Signed-off-by: Theodore Ts'o <tytso@mit.edu>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 53877ed201baa6b58f7ce9df92664a839113c30e
Author: Zhihao Cheng <chengzhihao1@huawei.com>
Date:   Tue Mar 21 09:37:21 2023 +0800

    ext4: fix i_disksize exceeding i_size problem in paritally written case

    [ Upstream commit 1dedde690303c05ef732b7c5c8356fdf60a4ade3 ]

    It is possible for i_disksize can exceed i_size, triggering a warning.

    generic_perform_write
     copied = iov_iter_copy_from_user_atomic(len) // copied < len
     ext4_da_write_end
     | ext4_update_i_disksize
```

```
       |  new_i_size = pos + copied;
       |  WRITE_ONCE(EXT4_I(inode)->i_disksize, newsize) // update i_disksize
       | generic_write_end
       |  copied = block_write_end(copied, len) // copied = 0
       |   if (unlikely(copied < len))
       |    if (!PageUptodate(page))
       |     copied = 0;
       |  if (pos + copied > inode->i_size) // return false
       if (unlikely(copied == 0))
        goto again;
       if (unlikely(iov_iter_fault_in_readable(i, bytes))) {
        status = -EFAULT;
        break;
       }

     We get i_disksize greater than i_size here, which could trigger WARNING
     check 'i_size_read(inode) < EXT4_I(inode)->i_disksize' while doing dio:

     ext4_dio_write_iter
      iomap_dio_rw
       __iomap_dio_rw // return err, length is not aligned to 512
      ext4_handle_inode_extension
       WARN_ON_ONCE(i_size_read(inode) < EXT4_I(inode)->i_disksize) // Oops

      WARNING: CPU: 2 PID: 2609 at fs/ext4/file.c:319
      CPU: 2 PID: 2609 Comm: aa Not tainted 6.3.0-rc2
      RIP: 0010:ext4_file_write_iter+0xbc7
      Call Trace:
       vfs_write+0x3b1
       ksys_write+0x77
       do_syscall_64+0x39

     Fix it by updating 'copied' value before updating i_disksize just like
     ext4_write_inline_data_end() does.

     A reproducer can be found in the buganizer link below.

     Link: https://bugzilla.kernel.org/show_bug.cgi?id=217209
     Fixes: 64769240bd07 ("ext4: Add delayed allocation support in data=writeback mode")
     Signed-off-by: Zhihao Cheng <chengzhihao1@huawei.com>
     Reviewed-by: Jan Kara <jack@suse.cz>
     Link: https://lore.kernel.org/r/20230321013721.89818-1-chengzhihao1@huawei.com
     Signed-off-by: Theodore Ts'o <tytso@mit.edu>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 08d11290455053f6873f156b68c9f2e2f3f3c22c
Author: Lars-Peter Clausen <lars@metafoo.de>
Date:   Sun Apr 9 09:25:44 2023 -0700

     rtc: jz4740: Make sure clock provider gets removed

     [ Upstream commit c7a639dac8e4d7e63450bef2f3a19fb331566fb1 ]

     The jz4740 RTC driver registers a clock provider, but never removes it.
     This leaves a stale clock provider behind that references freed clocks when
     the device is unbound.

     Use the managed `devm_of_clk_add_hw_provider()` instead of
     `of_clk_add_hw_provider()` to make sure the provider gets automatically
     removed on unbind.

     Fixes: 5ddfa148de8c ("rtc: jz4740: Register clock provider for the CLK32K pin")
     Signed-off-by: Lars-Peter Clausen <lars@metafoo.de>
     Link: https://lore.kernel.org/r/20230409162544.16155-1-lars@metafoo.de
     Signed-off-by: Alexandre Belloni <alexandre.belloni@bootlin.com>
```

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

commit aecb96159c29ae2f50fb6690e909c4eced116ce2
Author: Bharath SM <bharathsm@microsoft.com>
Date:   Wed Apr 26 14:05:16 2023 +0000

    SMB3: Close deferred file handles in case of handle lease break

    [ Upstream commit d906be3fa571f6fc9381911304a0eca99f1b6951 ]

    We should not cache deferred file handles if we dont have
    handle lease on a file. And we should immediately close all
    deferred handles in case of handle lease break.

    Fixes: 9e31678fb403 ("SMB3: fix lease break timeout when multiple deferred close handles
for the same file.")
    Signed-off-by: Bharath SM <bharathsm@microsoft.com>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 32a046ccaeea6c19965c04a4c521e703f6607924
Author: Bharath SM <bharathsm@microsoft.com>
Date:   Thu Apr 20 13:54:33 2023 +0000

    SMB3: Add missing locks to protect deferred close file list

    [ Upstream commit ab9ddc87a9055c4bebd6524d5d761d605d52e557 ]

    cifs_del_deferred_close function has a critical section which modifies
    the deferred close file list. We must acquire deferred_lock before
    calling cifs_del_deferred_close function.

    Fixes: ca08d0eac020 ("cifs: Fix memory leak on the deferred close")
    Signed-off-by: Bharath SM <bharathsm@microsoft.com>
    Acked-off-by: Paulo Alcantara (SUSE) <pc@manguebit.com>
    Acked-by: Ronnie Sahlberg <lsahlber@redhat.com>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f42dcc818af8043f763cb37a6fa14a256321750c
Author: Geert Uytterhoeven <geert+renesas@glider.be>
Date:   Wed Apr 26 15:43:34 2023 +0200

    timekeeping: Fix references to nonexistent ktime_get_fast_ns()

    [ Upstream commit 158009f1b4a33bc0f354b994eea361362bd83226 ]

    There was never a function named ktime_get_fast_ns().
    Presumably these should refer to ktime_get_mono_fast_ns() instead.

    Fixes: c1ce406e80fb15fa ("timekeeping: Fix up function documentation for the NMI safe
accessors")
    Signed-off-by: Geert Uytterhoeven <geert+renesas@glider.be>
    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Acked-by: John Stultz <jstultz@google.com>
    Link:
https://lore.kernel.org/r/06df7b3cbd94f016403bbf6cd2b38e4368e7468f.1682516546.git.geert+rene
sas@glider.be
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7a739918bc2071a2edbc4cf210535be956f2c91c
Author: Stafford Horne <shorne@gmail.com>
Date:   Sat Feb 11 19:14:06 2023 +0900

    openrisc: Properly store r31 to pt_regs on unhandled exceptions
```

```
    [ Upstream commit 812489ac4dd91144a74ce65ecf232252a2e406fb ]

    In commit 91993c8c2ed5 ("openrisc: use shadow registers to save regs on
    exception") the unhandled exception path was changed to do an early
    store of r30 instead of r31.  The entry code was not updated and r31 is
    not getting stored to pt_regs.

    This patch updates the entry handler to store r31 instead of r30.  We
    also remove some misleading commented out store r30 and r31
    instructrions.

    I noticed this while working on adding floating point exception
    handling,  This issue probably would never impact anything since we kill
    the process or Oops right away on unhandled exceptions.

    Fixes: 91993c8c2ed5 ("openrisc: use shadow registers to save regs on exception")
    Signed-off-by: Stafford Horne <shorne@gmail.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4bb98b6e5603cc10b8162360389acd6c70484e21
Author: Qinrun Dai <flno@hust.edu.cn>
Date:   Thu Apr 13 13:50:37 2023 +0000

    clocksource/drivers/davinci: Fix memory leak in davinci_timer_register when init fails

    [ Upstream commit fb73556386e074e9bee9fa2d253aeaefe4e063e0 ]

    Smatch reports:
    drivers/clocksource/timer-davinci.c:332 davinci_timer_register()
    warn: 'base' from ioremap() not released on lines: 274.

    Fix this and other potential memory leak problems
    by adding a set of corresponding exit lables.

    Fixes: 721154f972aa ("clocksource/drivers/davinci: Add support for clockevents")
    Signed-off-by: Qinrun Dai <flno@hust.edu.cn>
    Link: https://lore.kernel.org/r/20230413135037.1505799-1-flno@hust.edu.cn
    Signed-off-by: Daniel Lezcano <daniel.lezcano@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 98bf95141cc3e5e9f9133bae25b20ff43f89efa6
Author: Mark Zhang <markzhang@nvidia.com>
Date:   Thu Apr 20 04:39:06 2023 +0300

    RDMA/mlx5: Use correct device num_ports when modify DC

    [ Upstream commit 746aa3c8cb1a650ff2583497ac646e505831b9b9 ]

    Just like other QP types, when modify DC, the port_num should be compared
    with dev->num_ports, instead of HCA_CAP.num_ports.  Otherwise Multi-port
    vHCA on DC may not work.

    Fixes: 776a3906b692 ("IB/mlx5: Add support for DC target QP")
    Link: https://lore.kernel.org/r/20230420013906.1244185-1-markzhang@nvidia.com
    Signed-off-by: Mark Zhang <markzhang@nvidia.com>
    Reviewed-by: Maor Gottlieb <maorg@nvidia.com>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d449be3fb9bb268eec79df6b83b0aa9365c0a951
Author: Dai Ngo <dai.ngo@oracle.com>
Date:   Tue Apr 18 13:19:02 2023 -0700

    SUNRPC: remove the maximum number of retries in call_bind_status
```

```
    [ Upstream commit 691d0b782066a6eeeecbfceb7910a8f6184e6105 ]

    Currently call_bind_status places a hard limit of 3 to the number of
    retries on EACCES error. This limit was done to prevent NLM unlock
    requests from being hang forever when the server keeps returning garbage.
    However this change causes problem for cases when NLM service takes
    longer than 9 seconds to register with the port mapper after a restart.

    This patch removes this hard coded limit and let the RPC handles
    the retry based on the standard hard/soft task semantics.

    Fixes: 0b760113a3a1 ("NLM: Don't hang forever on NLM unlock requests")
    Reported-by: Helen Chao <helen.chao@oracle.com>
    Tested-by: Helen Chao <helen.chao@oracle.com>
    Signed-off-by: Dai Ngo <dai.ngo@oracle.com>
    Reviewed-by: Jeff Layton <jlayton@kernel.org>
    Signed-off-by: Anna Schumaker <Anna.Schumaker@Netapp.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit abe73a71ecb4cc5e7cf4bd521d2937ba6de09867
Author: Mark Bloch <mbloch@nvidia.com>
Date:   Thu Apr 13 12:23:09 2023 +0300

    RDMA/mlx5: Fix flow counter query via DEVX

    [ Upstream commit 3e358ea8614ddfbc59ca7a3f5dff5dde2b350b2c ]

    Commit cited in "fixes" tag added bulk support for flow counters but it
    didn't account that's also possible to query a counter using a non-base id
    if the counter was allocated as bulk.

    When a user performs a query, validate the flow counter id given in the
    mailbox is inside the valid range taking bulk value into account.

    Fixes: 208d70f562e5 ("IB/mlx5: Support flow counters offset for bulk counters")
    Signed-off-by: Mark Bloch <mbloch@nvidia.com>
    Reviewed-by: Maor Gottlieb <maorg@nvidia.com>
    Link:
https://lore.kernel.org/r/79d7fbe291690128e44672418934256254d93115.1681377114.git.leon@kerne
l.org
    Signed-off-by: Leon Romanovsky <leon@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6dccd770e75d1f67cb2a2064ab187e94a825f590
Author: Avihai Horon <avihaih@nvidia.com>
Date:   Mon Apr 10 16:07:51 2023 +0300

    RDMA/mlx5: Check pcie_relaxed_ordering_enabled() in UMR

    [ Upstream commit d43b020b0f82c088ef8ff3196ef00575a97d200e ]

    relaxed_ordering_read HCA capability is set if both the device supports
    relaxed ordering (RO) read and RO is set in PCI config space.

    RO in PCI config space can change during runtime. This will change the
    value of relaxed_ordering_read HCA capability in FW, but the driver will
    not see it since it queries the capabilities only once.

    This can lead to the following scenario:
    1. RO in PCI config space is enabled.
    2. User creates MKey without RO.
    3. RO in PCI config space is disabled.
       As a result, relaxed_ordering_read HCA capability is turned off in FW
       but remains on in driver copy of the capabilities.
```

```
    4. User requests to reconfig the MKey with RO via UMR.
    5. Driver will try to reconfig the MKey with RO read although it
       shouldn't (as relaxed_ordering_read HCA capability is really off).

    To fix this, check pcie_relaxed_ordering_enabled() before setting RO
    read in UMR.

    Fixes: 896ec9735336 ("RDMA/mlx5: Set mkey relaxed ordering by UMR with ConnectX-7")
    Signed-off-by: Avihai Horon <avihaih@nvidia.com>
    Reviewed-by: Shay Drory <shayd@nvidia.com>
    Link:
https://lore.kernel.org/r/8d39eb8317e7bed1a354311a20ae707788fd94ed.1681131553.git.leon@kerne
l.org
    Reviewed-by: Jacob Keller <jacob.e.keller@intel.com>
    Signed-off-by: Leon Romanovsky <leon@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0d938264fcfe4927e54f0e519da05af1d5d720b4
Author: Zhu Yanjun <yanjun.zhu@linux.dev>
Date:   Thu Apr 13 18:11:15 2023 +0800

    RDMA/rxe: Fix the error "trying to register non-static key in rxe_cleanup_task"

    [ Upstream commit b2b1ddc457458fecd1c6f385baa9fbda5f0c63ad ]

    In the function rxe_create_qp(), rxe_qp_from_init() is called to
    initialize qp, internally things like rxe_init_task are not setup until
    rxe_qp_init_req().

    If an error occurred before this point then the unwind will call
    rxe_cleanup() and eventually to rxe_qp_do_cleanup()/rxe_cleanup_task()
    which will oops when trying to access the uninitialized spinlock.

    If rxe_init_task is not executed, rxe_cleanup_task will not be called.

    Reported-by: syzbot+cfcc1a3c85be15a40cba@syzkaller.appspotmail.com
    Link: https://syzkaller.appspot.com/bug?id=fd85757b74b3eb59f904138486f755f71e090df8
    Fixes: 8700e3e7c485 ("Soft RoCE driver")
    Fixes: 2d4b21e0a291 ("IB/rxe: Prevent from completer to operate on non valid QP")
    Signed-off-by: Zhu Yanjun <yanjun.zhu@linux.dev>
    Link: https://lore.kernel.org/r/20230413101115.1366068-1-yanjun.zhu@intel.com
    Signed-off-by: Leon Romanovsky <leon@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9d2e108a5c1a9020ad4cf0f336eb726280cdfd96
Author: Bob Pearson <rpearsonhpe@gmail.com>
Date:   Sat Mar 4 11:45:32 2023 -0600

    RDMA/rxe: Remove __rxe_do_task()

    [ Upstream commit 960ebe97e5238565d15063c8f4d1b2108efe2e65 ]

    The subroutine __rxe_do_task is not thread safe and it has no way to
    guarantee that the tasks, which are designed with the assumption that they
    are non-reentrant, are not reentered. All of its uses are non-performance
    critical.

    This patch replaces calls to __rxe_do_task with calls to
    rxe_sched_task. It also removes irrelevant or unneeded if tests.

    Instead of calling the task machinery a single call to the tasklet
    function (rxe_requester, etc.) is sufficient to draing the queues if task
    execution has been disabled or stopped.

    Together these changes allow the removal of __rxe_do_task.
```

```
    Link: https://lore.kernel.org/r/20230304174533.11296-7-rpearsonhpe@gmail.com
    Signed-off-by: Ian Ziemba <ian.ziemba@hpe.com>
    Signed-off-by: Bob Pearson <rpearsonhpe@gmail.com>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Stable-dep-of: b2b1ddc45745 ("RDMA/rxe: Fix the error "trying to register non-static key
in rxe_cleanup_task"")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1661cc42bb9ceac591d318a024c5328fdc3c934d
Author: Bob Pearson <rpearsonhpe@gmail.com>
Date:   Sat Mar 4 11:45:27 2023 -0600

    RDMA/rxe: Convert tasklet args to queue pairs

    [ Upstream commit 3946fc2a42b18cf0b675121158a2625825ce27b5 ]

    Originally is was thought that the tasklet machinery in rxe_task.c would
    be used in other applications but that has not happened for years. This
    patch replaces the 'void *arg' by struct 'rxe_qp *qp' in the parameters to
    the tasklet calls. This change will have no affect on performance but may
    make the code a little clearer.

    Link: https://lore.kernel.org/r/20230304174533.11296-2-rpearsonhpe@gmail.com
    Signed-off-by: Bob Pearson <rpearsonhpe@gmail.com>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Stable-dep-of: b2b1ddc45745 ("RDMA/rxe: Fix the error "trying to register non-static key
in rxe_cleanup_task"")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a68ed2e0a8544c43873dd19f0f2d14a1b6971bed
Author: Michael Kelley <mikelley@microsoft.com>
Date:   Thu Apr 13 08:37:30 2023 -0700

    swiotlb: fix debugfs reporting of reserved memory pools

    [ Upstream commit 5499d01c029069044a3b3e50501c77b474c96178 ]

    For io_tlb_nslabs, the debugfs code reports the correct value for a
    specific reserved memory pool.  But for io_tlb_used, the value reported
    is always for the default pool, not the specific reserved pool. Fix this.

    Fixes: 5c850d31880e ("swiotlb: fix passing local variable to debugfs_create_ulong()")
    Signed-off-by: Michael Kelley <mikelley@microsoft.com>
    Signed-off-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4116bd222b1b67f9be6caf6f8b6bf0adbed9eae3
Author: Doug Berger <opendmb@gmail.com>
Date:   Fri Apr 14 14:29:25 2023 -0700

    swiotlb: relocate PageHighMem test away from rmem_swiotlb_setup

    [ Upstream commit a90922fa25370902322e9de6640e58737d459a50 ]

    The reservedmem_of_init_fn's are invoked very early at boot before the
    memory zones have even been defined. This makes it inappropriate to test
    whether the page corresponding to a PFN is in ZONE_HIGHMEM from within
    one.

    Removing the check allows an ARM 32-bit kernel with SPARSEMEM enabled to
    boot properly since otherwise we would be de-referencing an
    uninitialized sparsemem map to perform pfn_to_page() check.

    The arm64 architecture happens to work (and also has no high memory) but
```

```
    other 32-bit architectures could also be having similar issues.

    While it would be nice to provide early feedback about a reserved DMA
    pool residing in highmem, it is not possible to do that until the first
    time we try to use it, which is where the check is moved to.

    Fixes: 0b84e4f8b793 ("swiotlb: Add restricted DMA pool initialization")
    Signed-off-by: Doug Berger <opendmb@gmail.com>
    Signed-off-by: Florian Fainelli <f.fainelli@gmail.com>
    Signed-off-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9216aa5cfd86809a2681be3683cd9ac30432de0c
Author: Miaoqian Lin <linmq006@gmail.com>
Date:   Thu Apr 13 23:05:20 2023 -0700

    Input: raspberrypi-ts - fix refcount leak in rpi_ts_probe

    [ Upstream commit 5bca3688bdbc3b58a2894b8671a8e2378efe28bd ]

    rpi_firmware_get() take reference, we need to release it in error paths
    as well. Use devm_rpi_firmware_get() helper to handling the resources.
    Also remove the existing rpi_firmware_put().

    Fixes: 0b9f28fed3f7 ("Input: add official Raspberry Pi's touchscreen driver")
    Fixes: 3b8ddff780b7 ("input: raspberrypi-ts: Release firmware handle when not needed")
    Signed-off-by: Miaoqian Lin <linmq006@gmail.com>
    Reviewed-by: Mattijs Korpershoek <mkorpershoek@baylibre.com>
    Link: https://lore.kernel.org/r/20221223074657.810346-1-linmq006@gmail.com
    Signed-off-by: Dmitry Torokhov <dmitry.torokhov@gmail.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b9c798f1957c135eee519b41a4af2d18d53310d6
Author: Konrad Dybcio <konrad.dybcio@linaro.org>
Date:   Wed Apr 12 16:53:06 2023 +0200

    clk: qcom: dispcc-qcm2290: Remove inexistent DSI1PHY clk

    [ Upstream commit 68d1151f03067533827fc50b770954ef33149533 ]

    There's only one DSI PHY on this SoC. Remove the ghost entry for the
    clock produced by a secondary one.

    Fixes: cc517ea3333f ("clk: qcom: Add display clock controller driver for QCM2290")
    Signed-off-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230412-topic-qcm_dispcc-v1-2-bf2989a75ae4@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cd21c6bdaa11a34085daa94f811efeaeb478c6d2
Author: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
Date:   Wed Apr 12 16:48:29 2023 +0300

    clk: qcom: gcc-sm8350: fix PCIe PIPE clocks handling

    [ Upstream commit 1a500e0bc97b6cb3c0d9859e81973b8dd07d1b7b ]

    On SM8350 platform the PCIe PIPE clocks require additional handling to
    function correctly. They are to be switched to the tcxo source before
    turning PCIe GDSCs off and should be switched to PHY PIPE source once
    they are working. Switch PCIe PHY clocks to use clk_regmap_phy_mux_ops,
    which provide support for this dance.

    Fixes: 44c20c9ed37f ("clk: qcom: gcc: Add clock driver for SM8350")
    Signed-off-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
```

```
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230412134829.3686467-1-dmitry.baryshkov@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d97f25e940e597d56db072a692f72c4734b5fb9f
Author: Mohammad Rafi Shaik <quic_mohs@quicinc.com>
Date:   Fri Apr 7 14:52:55 2023 +0530

    clk: qcom: lpassaudiocc-sc7280: Add required gdsc power domain clks in
lpass_cc_sc7280_desc

    [ Upstream commit aad09fc7c4a522892eb64a79627b17a3869936cb ]

    Add GDSCs in lpass_cc_sc7280_desc struct.
    When qcom,adsp-pil-mode is enabled, GDSCs required to solve
    dependencies in lpass_audiocc probe().

    Fixes: 0cbcfbe50cbf ("clk: qcom: lpass: Handle the regmap overlap of lpasscc and
lpass_aon")
    Signed-off-by: Mohammad Rafi Shaik <quic_mohs@quicinc.com>
    Reviewed-by: Stephen Boyd <swboyd@chromium.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230407092255.119690-4-quic_mohs@quicinc.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9e86e89fc09232a813254f7c7499f706b7061960
Author: Srinivasa Rao Mandadapu <quic_srivasam@quicinc.com>
Date:   Fri Apr 7 14:52:54 2023 +0530

    clk: qcom: lpasscc-sc7280: Skip qdsp6ss clock registration

    [ Upstream commit 4fc1c2d9a2b7a394f3b873aae5e03bffd8b5cd31 ]

    The qdsp6ss memory region is being shared by ADSP remoteproc device and
    lpasscc clock device, hence causing memory conflict.
    To avoid this, when qdsp6ss clocks are being enabled in remoteproc driver,
    skip qdsp6ss clock registration if "qcom,adsp-pil-mode" is enabled and
    also assign max_register value.

    Fixes: 4ab43d171181 ("clk: qcom: Add lpass clock controller driver for SC7280")
    Signed-off-by: Srinivasa Rao Mandadapu <quic_srivasam@quicinc.com>
    Signed-off-by: Mohammad Rafi Shaik <quic_mohs@quicinc.com>
    Reviewed-by: Stephen Boyd <swboyd@chromium.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230407092255.119690-3-quic_mohs@quicinc.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit bd62631843ff6a75225072b9a38bd12b547d1c99
Author: Heiko Carstens <hca@linux.ibm.com>
Date:   Thu Mar 30 12:55:46 2023 +0200

    s390/checksum: always use cksm instruction

    [ Upstream commit e42ac7789df64120d7d3d57433dfc9f37ec0cb99 ]

    Commit dfe843dce775 ("s390/checksum: support GENERIC_CSUM, enable it for
    KASAN") switched s390 to use the generic checksum functions, so that KASAN
    instrumentation also works checksum functions by avoiding architecture
    specific inline assemblies.

    There is however the problem that the generic csum_partial() function
    returns a 32 bit value with a 16 bit folded checksum, while the original
    s390 variant does not fold to 16 bit. This in turn causes that the
    ipib_checksum in lowcore contains different values depending on kernel
    config options.
```

```
    The ipib_checksum is used by system dumpers to verify if pointers in
    lowcore point to valid data. Verification is done by comparing checksum
    values. The system dumpers still use 32 bit checksum values which are not
    folded, and therefore the checksum verification fails (incorrectly).

    Symptom is that reboot after dump does not work anymore when a KASAN
    instrumented kernel is dumped.

    Fix this by not using the generic checksum implementation. Instead add an
    explicit kasan_check_read() so that KASAN knows about the read access from
    within the inline assembly.

    Reported-by: Alexander Egorenkov <egorenar@linux.ibm.com>
    Fixes: dfe843dce775 ("s390/checksum: support GENERIC_CSUM, enable it for KASAN")
    Tested-by: Alexander Egorenkov <egorenar@linux.ibm.com>
    Signed-off-by: Heiko Carstens <hca@linux.ibm.com>
    Signed-off-by: Vasily Gorbik <gor@linux.ibm.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3ee6d3155df8555df3719c25cc77240e91ae8007
Author: Jerry Snitselaar <jsnitsel@redhat.com>
Date:   Tue Apr 4 00:27:42 2023 -0700

    iommu/amd: Set page size bitmap during V2 domain allocation

    [ Upstream commit 8f880d19e6ad645a4b8066d5ff091c980b3231e7 ]

    With the addition of the V2 page table support, the domain page size
    bitmap needs to be set prior to iommu core setting up direct mappings
    for reserved regions. When reserved regions are mapped, if this is not
    done, it will be looking at the V1 page size bitmap when determining
    the page size to use in iommu_pgsize(). When it gets into the actual
    amd mapping code, a check of see if the page size is supported can
    fail, because at that point it is checking it against the V2 page size
    bitmap which only supports 4K, 2M, and 1G.

    Add a check to __iommu_domain_alloc() to not override the
    bitmap if it was already set by the iommu ops domain_alloc() code path.

    Cc: Vasant Hegde <vasant.hegde@amd.com>
    Cc: Suravee Suthikulpanit <suravee.suthikulpanit@amd.com>
    Cc: Robin Murphy <robin.murphy@arm.com>
    Cc: Will Deacon <will@kernel.org>
    Cc: Joerg Roedel <joro@8bytes.org>
    Fixes: 4db6c41f0946 ("iommu/amd: Add support for using AMD IOMMU v2 page table for DMA-
API")
    Signed-off-by: Jerry Snitselaar <jsnitsel@redhat.com>
    Reviewed-by: Vasant Hegde <vasant.hegde@amd.com>
    Link: https://lore.kernel.org/r/20230404072742.1895252-1-jsnitsel@redhat.com
    Signed-off-by: Joerg Roedel <jroedel@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 67918a875545472991c7554b4b157e5a7e57c4d5
Author: Trond Myklebust <trond.myklebust@hammerspace.com>
Date:   Mon Mar 13 18:45:53 2023 -0400

    NFSv4.1: Always send a RECLAIM_COMPLETE after establishing lease

    [ Upstream commit 40882deb83c29d8df4470d4e5e7f137b6acf7ad1 ]

    The spec requires that we always at least send a RECLAIM_COMPLETE when
    we're done establishing the lease and recovering any state.

    Fixes: fce5c838e133 ("nfs41: RECLAIM_COMPLETE functionality")
```

```
    Signed-off-by: Trond Myklebust <trond.myklebust@hammerspace.com>
    Signed-off-by: Anna Schumaker <Anna.Schumaker@Netapp.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8219d09f5435f4fa9b4d5e1757a973c79e949238
Author: Peng Fan <peng.fan@nxp.com>
Date:   Fri Mar 31 14:38:10 2023 +0800

    clk: imx: imx8ulp: Fix XBAR_DIVBUS and AD_SLOW clock parents

    [ Upstream commit d608c18018c897b88d66f1340fe274b7181817fa ]

    XBAR_DIVBUS and AD_SLOW should set parent to XBAR_AD_DIVPLAT and
    XBAR_DIVBUS respectively, not the NIC_AD. otherwise we will get
    wrong clock rate.

    Fixes: c43a801a5789 ("clk: imx: Add clock driver for imx8ulp")
    Reviewed-by: Jacky Bai <ping.bai@nxp.com>
    Signed-off-by: Ye Li <ye.li@nxp.com>
    Signed-off-by: Peng Fan <peng.fan@nxp.com>
    Reviewed-by: Abel Vesa <abel.vesa@linaro.org>
    Link: https://lore.kernel.org/r/20230331063814.2462059-2-peng.fan@oss.nxp.com
    Signed-off-by: Abel Vesa <abel.vesa@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 758c8206a19ad86c72a468935c51c98f5aac65cc
Author: Peng Fan <peng.fan@nxp.com>
Date:   Mon Apr 3 17:52:55 2023 +0800

    clk: imx: fracn-gppll: disable hardware select control

    [ Upstream commit 4435467b15b069e5a6f50ca9a9260e86b74dbc13 ]

    When programming PLL, should disable Hardware control select to make PLL
    controlled by register, not hardware inputs through OSCPLL.

    Fixes: 1b26cb8a77a4 ("clk: imx: support fracn gppll")
    Signed-off-by: Peng Fan <peng.fan@nxp.com>
    Reviewed-by: Abel Vesa <abel.vesa@linaro.org>
    Link: https://lore.kernel.org/r/20230403095300.3386988-3-peng.fan@oss.nxp.com
    Signed-off-by: Abel Vesa <abel.vesa@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a6d621f420e63bd30e7d42bbd30577fe9d4ababb
Author: Peng Fan <peng.fan@nxp.com>
Date:   Mon Apr 3 17:52:54 2023 +0800

    clk: imx: fracn-gppll: fix the rate table

    [ Upstream commit cf8dccfedce848f67eaa42e8839305d028319161 ]

    The Fvco should be range 2.4GHz to 5GHz, the original table voilate the
    spec, so update the table to fix it.

    Fixes: c196175acdd3 ("clk: imx: clk-fracn-gppll: Add more freq config for video pll")
    Fixes: 044034efbeea ("clk: imx: clk-fracn-gppll: fix mfd value")
    Fixes: 1b26cb8a77a4 ("clk: imx: support fracn gppll")
    Signed-off-by: Jacky Bai <ping.bai@nxp.com>
    Signed-off-by: Peng Fan <peng.fan@nxp.com>
    Reviewed-by: Abel Vesa <abel.vesa@linaro.org>
    Link: https://lore.kernel.org/r/20230403095300.3386988-2-peng.fan@oss.nxp.com
    Signed-off-by: Abel Vesa <abel.vesa@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7e6010f79b58f45b204cf18aa58f4b73c3f30adc
```

```
Author: Patrick Kelsey <pat.kelsey@cornelisnetworks.com>
Date:   Fri Apr 7 12:52:44 2023 -0400

    IB/hfi1: Fix bugs with non-PAGE_SIZE-end multi-iovec user SDMA requests

    [ Upstream commit 00cbce5cbf88459cd1aa1d60d0f1df15477df127 ]

    hfi1 user SDMA request processing has two bugs that can cause data
    corruption for user SDMA requests that have multiple payload iovecs
    where an iovec other than the tail iovec does not run up to the page
    boundary for the buffer pointed to by that iovec.a

    Here are the specific bugs:
    1. user_sdma_txadd() does not use struct user_sdma_iovec->iov.iov_len.
       Rather, user_sdma_txadd() will add up to PAGE_SIZE bytes from iovec
       to the packet, even if some of those bytes are past
       iovec->iov.iov_len and are thus not intended to be in the packet.
    2. user_sdma_txadd() and user_sdma_send_pkts() fail to advance to the
       next iovec in user_sdma_request->iovs when the current iovec
       is not PAGE_SIZE and does not contain enough data to complete the
       packet. The transmitted packet will contain the wrong data from the
       iovec pages.

    This has not been an issue with SDMA packets from hfi1 Verbs or PSM2
    because they only produce iovecs that end short of PAGE_SIZE as the tail
    iovec of an SDMA request.

    Fixing these bugs exposes other bugs with the SDMA pin cache
    (struct mmu_rb_handler) that get in way of supporting user SDMA requests
    with multiple payload iovecs whose buffers do not end at PAGE_SIZE. So
    this commit fixes those issues as well.

    Here are the mmu_rb_handler bugs that non-PAGE_SIZE-end multi-iovec
    payload user SDMA requests can hit:
    1. Overlapping memory ranges in mmu_rb_handler will result in duplicate
       pinnings.
    2. When extending an existing mmu_rb_handler entry (struct mmu_rb_node),
       the mmu_rb code (1) removes the existing entry under a lock, (2)
       releases that lock, pins the new pages, (3) then reacquires the lock
       to insert the extended mmu_rb_node.

       If someone else comes in and inserts an overlapping entry between (2)
       and (3), insert in (3) will fail.

       The failure path code in this case unpins _all_ pages in either the
       original mmu_rb_node or the new mmu_rb_node that was inserted between
       (2) and (3).
    3. In hfi1_mmu_rb_remove_unless_exact(), mmu_rb_node->refcount is
       incremented outside of mmu_rb_handler->lock. As a result, mmu_rb_node
       could be evicted by another thread that gets mmu_rb_handler->lock and
       checks mmu_rb_node->refcount before mmu_rb_node->refcount is
       incremented.
    4. Related to #2 above, SDMA request submission failure path does not
       check mmu_rb_node->refcount before freeing mmu_rb_node object.

       If there are other SDMA requests in progress whose iovecs have
       pointers to the now-freed mmu_rb_node(s), those pointers to the
       now-freed mmu_rb nodes will be dereferenced when those SDMA requests
       complete.

    Fixes: 7be85676f1d1 ("IB/hfi1: Don't remove RB entry when not needed.")
    Fixes: 7724105686e7 ("IB/hfi1: add driver files")
    Signed-off-by: Brendan Cunningham <bcunningham@cornelisnetworks.com>
    Signed-off-by: Patrick Kelsey <pat.kelsey@cornelisnetworks.com>
    Signed-off-by: Dennis Dalessandro <dennis.dalessandro@cornelisnetworks.com>
```

```
    Link:
https://lore.kernel.org/r/168088636445.3027109.10054635277810177889.stgit@252.162.96.66.stat
ic.eigbox.net
    Signed-off-by: Leon Romanovsky <leon@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cfda73276a367236430f508c1593218523f0444a
Author: Patrick Kelsey <pat.kelsey@cornelisnetworks.com>
Date:   Fri Apr 7 12:52:39 2023 -0400

    IB/hfi1: Fix SDMA mmu_rb_node not being evicted in LRU order

    [ Upstream commit 9fe8fec5e43d5a80f43cbf61aaada1b047a1eb61 ]

    hfi1_mmu_rb_remove_unless_exact() did not move mmu_rb_node objects in
    mmu_rb_handler->lru_list after getting a cache hit on an mmu_rb_node.

    As a result, hfi1_mmu_rb_evict() was not guaranteed to evict truly
    least-recently used nodes.

    This could be a performance issue for an application when that
    application:
    - Uses some long-lived buffers frequently.
    - Uses a large number of buffers once.
    - Hits the mmu_rb_handler cache size or pinned-page limits, forcing
      mmu_rb_handler cache entries to be evicted.

    In this case, the one-time use buffers cause the long-lived buffer
    entries to eventually filter to the end of the LRU list where
    hfi1_mmu_rb_evict() will consider evicting a frequently-used long-lived
    entry instead of evicting one of the one-time use entries.

    Fix this by inserting new mmu_rb_node at the tail of
    mmu_rb_handler->lru_list and move mmu_rb_ndoe to the tail of
    mmu_rb_handler->lru_list when the mmu_rb_node is a hit in
    hfi1_mmu_rb_remove_unless_exact(). Change hfi1_mmu_rb_evict() to evict
    from the head of mmu_rb_handler->lru_list instead of the tail.

    Fixes: 0636e9ab8355 ("IB/hfi1: Add cache evict LRU list")
    Signed-off-by: Brendan Cunningham <bcunningham@cornelisnetworks.com>
    Signed-off-by: Patrick Kelsey <pat.kelsey@cornelisnetworks.com>
    Signed-off-by: Dennis Dalessandro <dennis.dalessandro@cornelisnetworks.com>
    Link:
https://lore.kernel.org/r/168088635931.3027109.10423156330761536044.stgit@252.162.96.66.stat
ic.eigbox.net
    Signed-off-by: Leon Romanovsky <leon@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b713623bfef8cb1df9c769a3887fa10db63d1c54
Author: Saravanan Vajravel <saravanan.vajravel@broadcom.com>
Date:   Wed Apr 5 21:25:49 2023 -0700

    RDMA/srpt: Add a check for valid 'mad_agent' pointer

    [ Upstream commit eca5cd9474cd26d62f9756f536e2e656d3f62f3a ]

    When unregistering MAD agent, srpt module has a non-null check
    for 'mad_agent' pointer before invoking ib_unregister_mad_agent().
    This check can pass if 'mad_agent' variable holds an error value.
    The 'mad_agent' can have an error value for a short window when
    srpt_add_one() and srpt_remove_one() is executed simultaneously.

    In srpt module, added a valid pointer check for 'sport->mad_agent'
    before unregistering MAD agent.
```

```
    This issue can hit when RoCE driver unregisters ib_device

    Stack Trace:
    ------------
    BUG: kernel NULL pointer dereference, address: 000000000000004d
    PGD 145003067 P4D 145003067 PUD 2324fe067 PMD 0
    Oops: 0002 [#1] PREEMPT SMP NOPTI
    CPU: 10 PID: 4459 Comm: kworker/u80:0 Kdump: loaded Tainted: P
    Hardware name: Dell Inc. PowerEdge R640/06NR82, BIOS 2.5.4 01/13/2020
    Workqueue: bnxt_re bnxt_re_task [bnxt_re]
    RIP: 0010:_raw_spin_lock_irqsave+0x19/0x40
    Call Trace:
      ib_unregister_mad_agent+0x46/0x2f0 [ib_core]
      IPv6: ADDRCONF(NETDEV_CHANGE): bond0: link becomes ready
      ? __schedule+0x20b/0x560
      srpt_unregister_mad_agent+0x93/0xd0 [ib_srpt]
      srpt_remove_one+0x20/0x150 [ib_srpt]
      remove_client_context+0x88/0xd0 [ib_core]
      bond0: (slave p2p1): link status definitely up, 100000 Mbps full duplex
      disable_device+0x8a/0x160 [ib_core]
      bond0: active interface up!
      ? kernfs_name_hash+0x12/0x80
     (NULL device *): Bonding Info Received: rdev: 000000006c0b8247
      __ib_unregister_device+0x42/0xb0 [ib_core]
     (NULL device *):         Master: mode: 4 num_slaves:2
      ib_unregister_device+0x22/0x30 [ib_core]
     (NULL device *):         Slave: id: 105069936 name:p2p1 link:0 state:0
      bnxt_re_stopqps_and_ib_uninit+0x83/0x90 [bnxt_re]
      bnxt_re_alloc_lag+0x12e/0x4e0 [bnxt_re]

    Fixes: a42d985bd5b2 ("ib_srpt: Initial SRP Target merge for v3.3-rc1")
    Reviewed-by: Selvin Xavier <selvin.xavier@broadcom.com>
    Reviewed-by: Kashyap Desai <kashyap.desai@broadcom.com>
    Signed-off-by: Saravanan Vajravel <saravanan.vajravel@broadcom.com>
    Link: https://lore.kernel.org/r/20230406042549.507328-1-saravanan.vajravel@broadcom.com
    Reviewed-by: Bart Van Assche <bvanassche@acm.org>
    Signed-off-by: Leon Romanovsky <leon@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 28d6d4373b18cbd199dace3305c1177953c2cc74
Author: Mark Zhang <markzhang@nvidia.com>
Date:   Thu Mar 30 10:23:51 2023 +0300

    RDMA/cm: Trace icm_send_rej event before the cm state is reset

    [ Upstream commit bd9de1badac7e4ff6780365d4aa38983f5e2a436 ]

    Trace icm_send_rej event before the cm state is reset to idle, so that
    correct cm state will be logged. For example when an incoming request is
    rejected, the old trace log was:
        icm_send_rej: local_id=961102742 remote_id=3829151631 state=IDLE
reason=REJ_CONSUMER_DEFINED
    With this patch:
        icm_send_rej: local_id=312971016 remote_id=3778819983 state=MRA_REQ_SENT
reason=REJ_CONSUMER_DEFINED

    Fixes: 8dc105befe16 ("RDMA/cm: Add tracepoints to track MAD send operations")
    Signed-off-by: Mark Zhang <markzhang@nvidia.com>
    Link: https://lore.kernel.org/r/20230330072351.481200-1-markzhang@nvidia.com
    Signed-off-by: Leon Romanovsky <leon@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8b97ab3181ed6498f5269337df60c3c4c847f8b9
Author: Chris Morgan <macromorgan@hotmail.com>
Date:   Fri Apr 7 11:18:27 2023 -0500
```

```
    power: supply: rk817: Fix low SOC bugs

    [ Upstream commit baba1315a74d12772d4940a05d58dc03e6ec0635 ]

    When the SOC approaches zero, an integer overflows in the columb
    counter causing the driver to react poorly. This makes the driver
    think it's at (above) the fully charged capacity when in fact it's
    zero. It would then write this full capacity to NVRAM which would be
    used on boot if the device remained off for less than 5 hours and
    not plugged in.

    This can be fixed and guarded against by doing the following:
     - Changing the type of tmp in rk817_read_or_set_full_charge_on_boot()
       to be an int instead of a u32. That way we can account for negative
       numbers.
     - Guard against negative values for the full charge on boot by setting
       the charge to 0 if the system charge reports less than 0.
     - Catch scenarios where the battery voltage is below the design
       minimum voltage and set the system SOC to 0 at that time and update
       the columb counter with a charge level of 0.
     - Change the off time value from 5 hours to 30 minutes before we
       recalculate the current capacity based on the OCV tables.

    These changes allow the driver to operate better at low voltage/low
    capacity conditions.

    Fixes: 3268a4d9b0b8 ("power: supply: rk817: Fix unsigned comparison with less than
zero")
    Fixes: 11cb8da0189b ("power: supply: Add charger driver for Rockchip RK817")
    Signed-off-by: Chris Morgan <macromorgan@hotmail.com>
    Signed-off-by: Sebastian Reichel <sebastian.reichel@collabora.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7b61fae34114645761673f753ce439c72ff04c57
Author: Konrad Dybcio <konrad.dybcio@linaro.org>
Date:   Wed Apr 5 00:47:19 2023 +0200

    clk: qcom: gcc-sm6115: Mark RCGs shared where applicable

    [ Upstream commit 996c32b745a15a637e8244a25f06b74acce98976 ]

    The vast majority of shared RCGs were not marked as such. Fix it.

    Fixes: cbe63bfdc54f ("clk: qcom: Add Global Clock controller (GCC) driver for SM6115")
    Signed-off-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230404224719.909746-1-konrad.dybcio@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 63bb90611c0a88f8af256760dc17af68e69e167a
Author: Tetsuo Handa <penguin-kernel@I-love.SAKURA.ne.jp>
Date:   Sun Apr 2 14:10:13 2023 +0900

    RDMA/siw: Remove namespace check from siw_netdev_event()

    [ Upstream commit 266e9b3475ba82212062771fdbc40be0e3c06ec8 ]

    syzbot is reporting that siw_netdev_event(NETDEV_UNREGISTER) cannot destroy
    siw_device created after unshare(CLONE_NEWNET) due to net namespace check.
    It seems that this check was by error there and should be removed.

    Reported-by: syzbot <syzbot+5e70d01ee8985ae62a3b@syzkaller.appspotmail.com>
    Link: https://syzkaller.appspot.com/bug?extid=5e70d01ee8985ae62a3b
    Suggested-by: Jason Gunthorpe <jgg@ziepe.ca>
```

```
    Suggested-by: Leon Romanovsky <leon@kernel.org>
    Fixes: bdcf26bf9b3a ("rdma/siw: network and RDMA core interface")
    Signed-off-by: Tetsuo Handa <penguin-kernel@I-love.SAKURA.ne.jp>
    Link: https://lore.kernel.org/r/a44e9ac5-44e2-d575-9e30-02483cc7ffd1@I-love.SAKURA.ne.jp
    Reviewed-by: Bernard Metzler <bmt@zurich.ibm.com>
    Signed-off-by: Leon Romanovsky <leon@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5e3168bd9c05eac7b4cae07dcdac07b52b6c392f
Author: Clément Léger <clement.leger@bootlin.com>
Date:   Tue Jan 31 09:32:27 2023 +0100

    clk: add missing of_node_put() in "assigned-clocks" property parsing

    [ Upstream commit 27a6e1b09a782517fddac91259970ac466a3f7b6 ]

    When returning from of_parse_phandle_with_args(), the np member of the
    of_phandle_args structure should be put after usage. Add missing
    of_node_put() calls in both __set_clk_parents() and __set_clk_rates().

    Fixes: 86be408bfbd8 ("clk: Support for clock parents and rates assigned from device
tree")
    Signed-off-by: Clément Léger <clement.leger@bootlin.com>
    Link: https://lore.kernel.org/r/20230131083227.10990-1-clement.leger@bootlin.com
    Signed-off-by: Stephen Boyd <sboyd@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5205aed1cf91b48f4d01de9020466ba0e5d2d36e
Author: Sebastian Reichel <sre@kernel.org>
Date:   Fri Mar 17 23:56:57 2023 +0100

    power: supply: generic-adc-battery: fix unit scaling

    [ Upstream commit 44263f50065969f2344808388bd589740f026167 ]

    power-supply properties are reported in µV, µA and µW.
    The IIO API provides mV, mA, mW, so the values need to
    be multiplied by 1000.

    Fixes: e60fea794e6e ("power: battery: Generic battery driver using IIO")
    Reviewed-by: Linus Walleij <linus.walleij@linaro.org>
    Reviewed-by: Matti Vaittinen <mazziesaccount@gmail.com>
    Signed-off-by: Sebastian Reichel <sre@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7af38ed57046726b96d0b77eb711aeab171def6c
Author: Bob Pearson <rpearsonhpe@gmail.com>
Date:   Mon Mar 27 16:56:44 2023 -0500

    RDMA/rxe: Remove tasklet call from rxe_cq.c

    [ Upstream commit 78b26a335310a097d6b22581b706050db42f196c ]

    Remove the tasklet call in rxe_cq.c and also the is_dying in the
    cq struct. There is no reason for the rxe driver to defer the call
    to the cq completion handler by scheduling a tasklet. rxe_cq_post()
    is not called in a hard irq context.

    The rxe driver currently is incorrect because the tasklet call is
    made without protecting the cq pointer with a reference from having
    the underlying memory freed before the deferred routine is called.
    Executing the comp_handler inline fixes this problem.

    Fixes: 8700e3e7c485 ("Soft RoCE driver")
    Signed-off-by: Bob Pearson <rpearsonhpe@gmail.com>
```

```
    Link: https://lore.kernel.org/r/20230327215643.10410-1-rpearsonhpe@gmail.com
    Acked-by: Zhu Yanjun <zyjzyj2000@gmail.com>
    Signed-off-by: Leon Romanovsky <leon@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6b8d0cd4f118193ccce0fb0336b715b74cdc7212
Author: Yong Wu <yong.wu@mediatek.com>
Date:   Thu Mar 16 18:14:45 2023 +0800

    iommu/mediatek: Set dma_mask for PGTABLE_PA_35_EN

    [ Upstream commit f045e9df6537175d02565f21616ac1a9dd59b61c ]

    When we enable PGTABLE_PA_35_EN, the PA for pgtable may be 35bits.
    Thus add dma_mask for it.

    Fixes: 301c3ca12576 ("iommu/mediatek: Allow page table PA up to 35bit")
    Signed-off-by: Chengci.Xu <chengci.xu@mediatek.com>
    Signed-off-by: Yong Wu <yong.wu@mediatek.com>
    Reviewed-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Link: https://lore.kernel.org/r/20230316101445.12443-1-yong.wu@mediatek.com
    Signed-off-by: Joerg Roedel <jroedel@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4a034ece7e2877673d9085d6e7ed45e6ee40b761
Author: Zeng Heng <zengheng4@huawei.com>
Date:   Mon Dec 12 09:31:34 2022 +0800

    fs/ntfs3: Fix slab-out-of-bounds read in hdr_delete_de()

    [ Upstream commit ab84eee4c7ab929996602eda7832854c35a6dda2 ]

    Here is a BUG report from syzbot:

    BUG: KASAN: slab-out-of-bounds in hdr_delete_de+0xe0/0x150 fs/ntfs3/index.c:806
    Read of size 16842960 at addr ffff888079cc0600 by task syz-executor934/3631

    Call Trace:
     memmove+0x25/0x60 mm/kasan/shadow.c:54
     hdr_delete_de+0xe0/0x150 fs/ntfs3/index.c:806
     indx_delete_entry+0x74f/0x3670 fs/ntfs3/index.c:2193
     ni_remove_name+0x27a/0x980 fs/ntfs3/frecord.c:2910
     ntfs_unlink_inode+0x3d4/0x720 fs/ntfs3/inode.c:1712
     ntfs_rename+0x41a/0xcb0 fs/ntfs3/namei.c:276

    Before using the meta-data in struct INDEX_HDR, we need to
    check index header valid or not. Otherwise, the corruptedi
    (or malicious) fs image can cause out-of-bounds access which
    could make kernel panic.

    Fixes: 82cae269cfa9 ("fs/ntfs3: Add initialization of super block")
    Reported-by: syzbot+9c2811fd56591639ff5f@syzkaller.appspotmail.com
    Signed-off-by: Zeng Heng <zengheng4@huawei.com>
    Signed-off-by: Konstantin Komarov <almaz.alexandrovich@paragon-software.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4bf3b564e27a518f158a83d5e1a50064ed6136a0
Author: ZhangPeng <zhangpeng362@huawei.com>
Date:   Wed Dec 7 09:46:10 2022 +0000

    fs/ntfs3: Fix OOB read in indx_insert_into_buffer

    [ Upstream commit b8c44949044e5f7f864525fdffe8e95135ce9ce5 ]

    Syzbot reported a OOB read bug:
```

```
    BUG: KASAN: slab-out-of-bounds in indx_insert_into_buffer+0xaa3/0x13b0
    fs/ntfs3/index.c:1755
    Read of size 17168 at addr ffff8880255e06c0 by task syz-executor308/3630

    Call Trace:
     <TASK>
     memmove+0x25/0x60 mm/kasan/shadow.c:54
     indx_insert_into_buffer+0xaa3/0x13b0 fs/ntfs3/index.c:1755
     indx_insert_entry+0x446/0x6b0 fs/ntfs3/index.c:1863
     ntfs_create_inode+0x1d3f/0x35c0 fs/ntfs3/inode.c:1548
     ntfs_create+0x3e/0x60 fs/ntfs3/namei.c:100
     lookup_open fs/namei.c:3413 [inline]

    If the member struct INDEX_BUFFER *index of struct indx_node is
    incorrect, that is, the value of __le32 used is greater than the value
    of __le32 total in struct INDEX_HDR. Therefore, OOB read occurs when
    memmove is called in indx_insert_into_buffer().
    Fix this by adding a check in hdr_find_e().

    Fixes: 82cae269cfa9 ("fs/ntfs3: Add initialization of super block")
    Reported-by: syzbot+d882d57193079e379309@syzkaller.appspotmail.com
    Signed-off-by: ZhangPeng <zhangpeng362@huawei.com>
    Signed-off-by: Konstantin Komarov <almaz.alexandrovich@paragon-software.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cdcdfd57f4c701f832787da1309cc6687917d783
Author: Jiasheng Jiang <jiasheng@iscas.ac.cn>
Date:   Wed Nov 23 16:48:46 2022 +0800

    fs/ntfs3: Add check for kmemdup

    [ Upstream commit e6c3cef24cb0d045f99d5cb039b344874e3cfd74 ]

    Since the kmemdup may return NULL pointer,
    it should be better to add check for the return value
    in order to avoid NULL pointer dereference.

    Fixes: b46acd6a6a62 ("fs/ntfs3: Add NTFS journal")
    Signed-off-by: Jiasheng Jiang <jiasheng@iscas.ac.cn>
    Signed-off-by: Konstantin Komarov <almaz.alexandrovich@paragon-software.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3bb0d3eb475f01744ce6d6e998dfbd80220852a1
Author: Chen Zhongjin <chenzhongjin@huawei.com>
Date:   Tue Nov 22 17:24:14 2022 +0800

    fs/ntfs3: Fix memory leak if ntfs_read_mft failed

    [ Upstream commit bfa434c60157c9793e9b12c9b68ade02aff9f803 ]

    Label ATTR_ROOT in ntfs_read_mft() sets is_root = true and
    ni->ni_flags |= NI_FLAG_DIR, then next attr will goto label ATTR_ALLOC
    and alloc ni->dir.alloc_run. However two states are not always
    consistent and can make memory leak.

     1) attr_name in ATTR_ROOT does not fit the condition it will set
     is_root = true but NI_FLAG_DIR is not set.
     2) next attr_name in ATTR_ALLOC fits the condition and alloc
     ni->dir.alloc_run
     3) in cleanup function ni_clear(), when NI_FLAG_DIR is set, it frees
     ni->dir.alloc_run, otherwise it frees ni->file.run
     4) because NI_FLAG_DIR is not set in this case, ni->dir.alloc_run is
     leaked as kmemleak reported:
```

```
      unreferenced object 0xffff888003bc5480 (size 64):
        backtrace:
          [<000000003d42e6b0>] __kmalloc_node+0x4e/0x1c0
          [<00000000d8e19b8a>] kvmalloc_node+0x39/0x1f0
          [<00000000fc3eb5b8>] run_add_entry+0x18a/0xa40 [ntfs3]
          [<0000000011c9f978>] run_unpack+0x75d/0x8e0 [ntfs3]
          [<00000000e7cf1819>] run_unpack_ex+0xbc/0x500 [ntfs3]
          [<00000000bbf0a43d>] ntfs_iget5+0xb25/0x2dd0 [ntfs3]
          [<00000000a6e50693>] ntfs_fill_super+0x218d/0x3580 [ntfs3]
          [<00000000b9170608>] get_tree_bdev+0x3fb/0x710
          [<000000004833798a>] vfs_get_tree+0x8e/0x280
          [<000000006e20b8e6>] path_mount+0xf3c/0x1930
          [<000000007bf15a5f>] do_mount+0xf3/0x110
          ...

      Fix this by always setting is_root and NI_FLAG_DIR together.

      Fixes: 82cae269cfa9 ("fs/ntfs3: Add initialization of super block")
      Signed-off-by: Chen Zhongjin <chenzhongjin@huawei.com>
      Signed-off-by: Konstantin Komarov <almaz.alexandrovich@paragon-software.com>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 6ff3259bea93565353bec53afa10761483104a2f
  Author: Cheng Xu <chengyou@linux.alibaba.com>
  Date:   Tue Mar 7 18:29:23 2023 +0800

      RDMA/erdma: Use fixed hardware page size

      [ Upstream commit d649c638dc26f3501da510cf7fceb5c15ca54258 ]

      Hardware's page size is 4096, but the kernel's page size may vary. Driver
      should use hardware's page size when communicating with hardware.

      Fixes: 155055771704 ("RDMA/erdma: Add verbs implementation")
      Link: https://lore.kernel.org/r/20230307102924.70577-2-chengyou@linux.alibaba.com
      Signed-off-by: Cheng Xu <chengyou@linux.alibaba.com>
      Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 911e4c93c0c03464d27810a868e7cdbdef813e5f
  Author: Bob Pearson <rpearsonhpe@gmail.com>
  Date:   Fri Mar 3 16:16:21 2023 -0600

      RDMA/rxe: Replace exists by rxe in rxe.c

      [ Upstream commit 9168d125ea032ad199275193493c13cb077da5cc ]

      'exists' looks like a boolean. This patch replaces it by the
      normal name used for the rxe device, 'rxe', which should be a
      little less confusing. The second rxe_dbg() message is
      incorrect since rxe is known to be NULL and this will cause a
      seg fault if this message were ever sent. Replace it by pr_debug
      for the moment.

      Fixes: c6aba5ea0055 ("RDMA/rxe: Replace pr_xxx by rxe_dbg_xxx in rxe.c")
      Link: https://lore.kernel.org/r/20230303221623.8053-2-rpearsonhpe@gmail.com
      Signed-off-by: Bob Pearson <rpearsonhpe@gmail.com>
      Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 4d2d91bbcbc9c6b9f7a3249f896cfa8de0e53cd7
  Author: Dhruva Gole <d-gole@ti.com>
  Date:   Thu Mar 23 14:29:04 2023 +0530

      rtc: k3: handle errors while enabling wake irq
```

```
    [ Upstream commit d31d7300ebc0c43021ec48c0e6a3a427386f4617 ]

    Due to the potential failure of enable_irq_wake(), it would be better to
    return error if it fails.

    Fixes: b09d633575e5 ("rtc: Introduce ti-k3-rtc")
    Cc: Nishanth Menon <nm@ti.com>
    Signed-off-by: Dhruva Gole <d-gole@ti.com>
    Link: https://lore.kernel.org/r/20230323085904.957999-1-d-gole@ti.com
    Signed-off-by: Alexandre Belloni <alexandre.belloni@bootlin.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 22c4297ec2b469bdc5235ec4e92f65caa4bc06f9
Author: Martin Blumenstingl <martin.blumenstingl@googlemail.com>
Date:   Mon Mar 20 22:21:42 2023 +0100

    rtc: meson-vrtc: Use ktime_get_real_ts64() to get the current time

    [ Upstream commit 0e6255fa3f649170da6bd1a544680589cfae1131 ]

    The VRTC alarm register can be programmed with an amount of seconds
    after which the SoC will be woken up by the VRTC timer again. We are
    already converting the alarm time from meson_vrtc_set_alarm() to
    "seconds since 1970". This means we also need to use "seconds since
    1970" for the current time.

    This fixes a problem where setting the alarm to one minute in the future
    results in the firmware (which handles wakeup) to output (on the serial
    console) that the system will be woken up in billions of seconds.
    ktime_get_raw_ts64() returns the time since boot, not since 1970. Switch
    to ktime_get_real_ts64() to fix the calculation of the alarm time and to
    make the SoC wake up at the specified date/time. Also the firmware
    (which manages suspend) now prints either 59 or 60 seconds until wakeup
    (depending on how long it takes for the system to enter suspend).

    Fixes: 6ef35398e827 ("rtc: Add Amlogic Virtual Wake RTC")
    Signed-off-by: Martin Blumenstingl <martin.blumenstingl@googlemail.com>
    Reviewed-by: Neil Armstrong <neil.armstrong@linaro.org>
    Reviewed-by: Kevin Hilman <khilman@baylibre.com>
    Link: https://lore.kernel.org/r/20230320212142.2355062-1-
martin.blumenstingl@googlemail.com
    Signed-off-by: Alexandre Belloni <alexandre.belloni@bootlin.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a183905869e692b6b7805b7472235585eff8e429
Author: Dan Carpenter <error27@gmail.com>
Date:   Tue Mar 7 12:51:27 2023 +0300

    RDMA/mlx4: Prevent shift wrapping in set_user_sq_size()

    [ Upstream commit d50b3c73f1ac20dabc53dc6e9d64ce9c79a331eb ]

    The ucmd->log_sq_bb_count variable is controlled by the user so this
    shift can wrap.  Fix it by using check_shl_overflow() in the same way
    that it was done in commit 515f60004ed9 ("RDMA/hns: Prevent undefined
    behavior in hns_roce_set_user_sq_size()").

    Fixes: 839041329fd3 ("IB/mlx4: Sanity check userspace send queue sizes")
    Signed-off-by: Dan Carpenter <error27@gmail.com>
    Link: https://lore.kernel.org/r/a8dfbd1d-c019-4556-930b-bab1ded73b10@kili.mountain
    Signed-off-by: Leon Romanovsky <leon@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit afc86b68982eaf03a11de870b3401e822497c65b
```

```
Author: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
Date:   Sat Mar 11 10:40:21 2023 +0100

    rtc: omap: include header for omap_rtc_power_off_program prototype

    [ Upstream commit f69c2b5420497b7a54181ce170d682cbeb1f119f ]

    Non-static functions should have a prototype:

      drivers/rtc/rtc-omap.c:410:5: error: no previous prototype for
‘omap_rtc_power_off_program’ [-Werror=missing-prototypes]

    Fixes: 6256f7f7f217 ("rtc: OMAP: Add support for rtc-only mode")
    Signed-off-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Link: https://lore.kernel.org/r/20230311094021.79730-1-krzysztof.kozlowski@linaro.org
    Signed-off-by: Alexandre Belloni <alexandre.belloni@bootlin.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a9fce0f7bfbe2c31680935e20c7173db815574d0
Author: Petr Mladek <pmladek@suse.com>
Date:   Tue Mar 7 13:53:31 2023 +0100

    workqueue: Fix hung time report of worker pools

    [ Upstream commit 335a42ebb0ca8ee9997a1731aaaae6dcd704c113 ]

    The workqueue watchdog prints a warning when there is no progress in
    a worker pool. Where the progress means that the pool started processing
    a pending work item.

    Note that it is perfectly fine to process work items much longer.
    The progress should be guaranteed by waking up or creating idle
    workers.

    show_one_worker_pool() prints state of non-idle worker pool. It shows
    a delay since the last pool->watchdog_ts.

    The timestamp is updated when a first pending work is queued in
    __queue_work(). Also it is updated when a work is dequeued for
    processing in worker_thread() and rescuer_thread().

    The delay is misleading when there is no pending work item. In this
    case it shows how long the last work item is being proceed. Show
    zero instead. There is no stall if there is no pending work.

    Fixes: 82607adcf9cdf40fb7b ("workqueue: implement lockup detector")
    Signed-off-by: Petr Mladek <pmladek@suse.com>
    Signed-off-by: Tejun Heo <tj@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 610abed72a33ba1b4e347832d2a2356523f4ca6b
Author: Konrad Dybcio <konrad.dybcio@linaro.org>
Date:   Wed Mar 15 18:30:48 2023 +0100

    clk: qcom: gcc-qcm2290: Fix up gcc_sdcc2_apps_clk_src

    [ Upstream commit 1bf088a9f0e50acd175ba8deef0db11c099fa26e ]

    Add the PARENT_ENABLE flag to prevent  the clock from getting stuck
    at boot and use floor_ops to avoid SDHCI overclocking.

    Fixes: 496d1a13d405 ("clk: qcom: Add Global Clock Controller driver for QCM2290")
    Signed-off-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230315173048.3497655-1-konrad.dybcio@linaro.org
```

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

commit 934124ef5cab8fe03d8651444900ea10f7e813c4
Author: Yang Yingliang <yangyingliang@huawei.com>
Date:   Thu Dec 29 17:29:46 2022 +0800

    clk: mediatek: clk-pllfh: fix missing of_node_put() in fhctl_parse_dt()

    [ Upstream commit f1d97a37f975ac615e4d6875c27516150642d499 ]

    The device_node pointer returned by of_find_compatible_node() with
    refcount incremented, when finish using it, the refcount need be
    decreased.

    Fixes: d7964de8a8ea ("clk: mediatek: Add new clock driver to handle FHCTL hardware")
    Signed-off-by: Yang Yingliang <yangyingliang@huawei.com>
    Link: https://lore.kernel.org/r/20221229092946.4162345-1-yangyingliang@huawei.com
    [sboyd@kernel.org: Also unmap on error]
    Signed-off-by: Stephen Boyd <sboyd@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f1a0bad9ea9e77c3d0991b4c79fc6087fb58dda3
Author: Natalia Petrova <n.petrova@fintech.ru>
Date:   Fri Mar 3 15:44:08 2023 +0300

    RDMA/rdmavt: Delete unnecessary NULL check

    [ Upstream commit b73a0b80c69de77d8d4942abb37066531c0169b2 ]

    There is no need to check 'rdi->qp_dev' for NULL. The field 'qp_dev'
    is created in rvt_register_device() which will fail if the 'qp_dev'
    allocation fails in rvt_driver_qp_init(). Overwise this pointer
    doesn't changed and passed to rvt_qp_exit() by the next step.

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

    Fixes: 0acb0cc7ecc1 ("IB/rdmavt: Initialize and teardown of qpn table")
    Signed-off-by: Natalia Petrova <n.petrova@fintech.ru>
    Link: https://lore.kernel.org/r/20230303124408.16685-1-n.petrova@fintech.ru
    Signed-off-by: Leon Romanovsky <leon@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f210067c5f2f49038fe20453bca9b526ac055787
Author: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
Date:   Mon Mar 6 15:05:40 2023 +0100

    clk: mediatek: mt8135: Properly use CLK_IS_CRITICAL flag

    [ Upstream commit f4f9a9c003b52ea3cffda186753bfb3e37b970f8 ]

    Instead of calling clk_prepare_enable() for clocks that shall stay
    enabled, use the CLK_IS_CRITICAL flag, which purpose is exactly that.

    Fixes: a8aede794843 ("clk: mediatek: Add basic clocks for Mediatek MT8135.")
    Signed-off-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Reviewed-by: Chen-Yu Tsai <wenst@chromium.org>
    Link: https://lore.kernel.org/r/20230306140543.1813621-52-
angelogioacchino.delregno@collabora.com
    Signed-off-by: Stephen Boyd <sboyd@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6820921bf7aeacdb0be476c92b8ed84abc35265c
Author: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
Date:   Mon Mar 6 15:05:12 2023 +0100
```

```
      clk: mediatek: mt7622: Properly use CLK_IS_CRITICAL flag

      [ Upstream commit fa8c0d01df62130ff596d560380a6f844f62639e ]

      Instead of calling clk_prepare_enable() for clocks that shall stay
      enabled, use the CLK_IS_CRITICAL flag, which purpose is exactly that.

      Fixes: 2fc0a509e4ee ("clk: mediatek: add clock support for MT7622 SoC")
      Signed-off-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
      Reviewed-by: Chen-Yu Tsai <wenst@chromium.org>
      Link: https://lore.kernel.org/r/20230306140543.1813621-24-
  angelogioacchino.delregno@collabora.com
      Signed-off-by: Stephen Boyd <sboyd@kernel.org>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 62cbd03ba43cdc8276d4b5ed21f9a56e6eaf0da2
  Author: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
  Date:   Mon Mar 6 15:05:11 2023 +0100

      clk: mediatek: Consistently use GATE_MTK() macro

      [ Upstream commit 4c85e20b656607897e3bb06ff565822fa4b4de95 ]

      All the various MediaTek clock drivers are, in a way or another,
      redefining the GATE_MTK() macro with different names: while some
      are doing that by actually using GATE_MTK(), others are copying
      it entirely (hence, entirely redefining it).

      Change all clock drivers to always and consistently use this macro.

      Signed-off-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
      Reviewed-by: Chen-Yu Tsai <wenst@chromium.org>
      Tested-by: Chen-Yu Tsai <wenst@chromium.org> # MT8183, MT8192, MT8195 Chromebooks
      Link: https://lore.kernel.org/r/20230306140543.1813621-23-
  angelogioacchino.delregno@collabora.com
      Signed-off-by: Stephen Boyd <sboyd@kernel.org>
      Stable-dep-of: fa8c0d01df62 ("clk: mediatek: mt7622: Properly use CLK_IS_CRITICAL flag")
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 2a281d2ea74d49648b75f254eeb0f8341ae8a436
  Author: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
  Date:   Mon Mar 6 15:04:56 2023 +0100

      clk: mediatek: mt2712: Add error handling to clk_mt2712_apmixed_probe()

      [ Upstream commit 20cace1b9d7e33f68f0ee17196bf0df618dbacbe ]

      This function was completely missing error handling: add it.

      Fixes: e2f744a82d72 ("clk: mediatek: Add MT2712 clock support")
      Signed-off-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
      Reviewed-by: Chen-Yu Tsai <wenst@chromium.org>
      Link: https://lore.kernel.org/r/20230306140543.1813621-8-
  angelogioacchino.delregno@collabora.com
      Signed-off-by: Stephen Boyd <sboyd@kernel.org>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 642a7c9d1d98a7b0c18fe046561257556604c091
  Author: Daniil Dulov <d.dulov@aladdin.ru>
  Date:   Mon Feb 27 01:17:51 2023 -0800

      RDMA/siw: Fix potential page_array out of range access

      [ Upstream commit 271bfcfb83a9f77cbae3d6e1a16e3c14132922f0 ]
```

```
    When seg is equal to MAX_ARRAY, the loop should break, otherwise
    it will result in out of range access.

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

    Fixes: b9be6f18cf9e ("rdma/siw: transmit path")
    Signed-off-by: Daniil Dulov <d.dulov@aladdin.ru>
    Link: https://lore.kernel.org/r/20230227091751.589612-1-d.dulov@aladdin.ru
    Signed-off-by: Leon Romanovsky <leon@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 33f175a312a3dea35dd33402357f1b0fdbac5942
Author: Kang Chen <void0red@gmail.com>
Date:   Mon Feb 27 18:02:12 2023 +0800

    IB/hifi1: add a null check of kzalloc_node in hfi1_ipoib_txreq_init

    [ Upstream commit c874ad879c2f29ebe040a34b974389875c0d81eb ]

    kzalloc_node may fails, check it and do the cleanup.

    Fixes: b1151b74ff68 ("IB/hfi1: Fix alloc failure with larger txqueuelen")
    Signed-off-by: Kang Chen <void0red@gmail.com>
    Link: https://lore.kernel.org/r/20230227100212.910820-1-void0red@gmail.com
    Signed-off-by: Leon Romanovsky <leon@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9b4aa8a78242c02f503214275fe9d3eb27bc3810
Author: Claudiu Beznea <claudiu.beznea@microchip.com>
Date:   Mon Feb 27 12:59:31 2023 +0200

    clk: at91: clk-sam9x60-pll: fix return value check

    [ Upstream commit 1bd8e27fd0db0fe7f489213836dcbab92934f8fa ]

    sam9x60_frac_pll_compute_mul_frac() can't return zero. Remove the check
    against zero to reflect this.

    Fixes: 43b1bb4a9b3e ("clk: at91: clk-sam9x60-pll: re-factor to support plls with
multiple outputs")
    Reported-by: Dan Carpenter <error27@gmail.com>
    Signed-off-by: Claudiu Beznea <claudiu.beznea@microchip.com>
    Link: https://lore.kernel.org/r/20230227105931.2812412-1-claudiu.beznea@microchip.com
    Signed-off-by: Stephen Boyd <sboyd@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit fa7f2f5d1739452280c22727c4384a52b72ab5de
Author: Beau Belgrave <beaub@linux.microsoft.com>
Date:   Tue Apr 25 15:51:04 2023 -0700

    tracing/user_events: Ensure write index cannot be negative

    [ Upstream commit cd98c93286a30cc4588dfd02453bec63c2f4acf4 ]

    The write index indicates which event the data is for and accesses a
    per-file array. The index is passed by user processes during write()
    calls as the first 4 bytes. Ensure that it cannot be negative by
    returning -EINVAL to prevent out of bounds accesses.

    Update ftrace self-test to ensure this occurs properly.

    Link: https://lkml.kernel.org/r/20230425225107.8525-2-beaub@linux.microsoft.com

    Fixes: 7f5a08c79df3 ("user_events: Add minimal support for trace_event into ftrace")
    Reported-by: Doug Cook <dcook@linux.microsoft.com>
```

```
    Signed-off-by: Beau Belgrave <beaub@linux.microsoft.com>
    Signed-off-by: Steven Rostedt (Google) <rostedt@goodmis.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a6a8b111f31dc53991882d60f8f5497e07220c89
Author: Daniel Bristot de Oliveira <bristot@kernel.org>
Date:   Fri Mar 24 20:26:42 2023 +0100

    rtla/timerlat: Fix "Previous IRQ" auto analysis' line

    [ Upstream commit 82253a271aae9271fcf0aaa5e0ecc6dd38fb872b ]

    The "Previous IRQ interference" line is misaligned and without
    a \n, breaking the tool's output:

     ## CPU 12 hit stop tracing, analyzing it ##
      Previous IRQ interference:                    up to      2.22 us  IRQ handler delay:
18.06 us (0.00 %)
      IRQ latency:                                             18.52 us
      Timerlat IRQ duration:                                    4.41 us (0.00 %)
      Blocking thread:                                        216.93 us (0.03 %)

    Fix the output:

     ## CPU 7 hit stop tracing, analyzing it ##
      Previous IRQ interference:                     up to       8.93 us
      IRQ handler delay:                                         0.98 us (0.00 %)
      IRQ latency:                                               2.95 us
      Timerlat IRQ duration:                                    11.26 us (0.03 %)

    Link: https://lore.kernel.org/linux-trace-
devel/8b5819077f15ccf24745c9bf3205451e16ee32d9.1679685525.git.bristot@kernel.org

    Fixes: 27e348b221f6 ("rtla/timerlat: Add auto-analysis core")
    Cc: Masami Hiramatsu <mhiramat@kernel.org>
    Signed-off-by: Daniel Bristot de Oliveira <bristot@kernel.org>
    Signed-off-by: Steven Rostedt (Google) <rostedt@goodmis.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f35cf01925905409b34cf5d52cc17a7da3c4641b
Author: Colin Ian King <colin.i.king@gmail.com>
Date:   Mon Apr 24 10:47:30 2023 +0100

    rv: Fix addition on an uninitialized variable 'run'

    [ Upstream commit 54a0dffa62de0c91b406ff32082a121ccfa0d7f1 ]

    The variable run is not initialized however it is being accumulated
    by the return value from the call to ikm_run_monitor.  Fix this by
    initializing run to zero at the start of the function.

    Link: https://lkml.kernel.org/r/20230424094730.105313-1-colin.i.king@gmail.com

    Fixes: 4bc4b131d44c ("rv: Add rv tool")

    Signed-off-by: Colin Ian King <colin.i.king@gmail.com>
    Acked-by: Daniel Bristot de Oliveira <bristot@kernel.org>
    Signed-off-by: Steven Rostedt (Google) <rostedt@goodmis.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b35286f6842dce61ca656cc7ae9570ccf1770ffc
Author: Aaron Thompson <dev@aaront.org>
Date:   Thu Apr 13 17:50:12 2023 +0000

    sched/clock: Fix local_clock() before sched_clock_init()
```

```
    [ Upstream commit f31dcb152a3d0816e2f1deab4e64572336da197d ]

    Have local_clock() return sched_clock() if sched_clock_init() has not
    yet run. sched_clock_cpu() has this check but it was not included in the
    new noinstr implementation of local_clock().

    The effect can be seen on x86 with CONFIG_PRINTK_TIME enabled, for
    instance. scd->clock quickly reaches the value of TICK_NSEC and that
    value is returned until sched_clock_init() runs.

    dmesg without this patch:

        [    0.000000] kvm-clock: ...
        [    0.000002] kvm-clock: ...
        [    0.000672] clocksource: ...
        [    0.001000] tsc: ...
        [    0.001000] e820: ...
        [    0.001000] e820: ...
         ...
        [    0.001000] ..TIMER: ...
        [    0.001000] clocksource: ...
        [    0.378956] Calibrating delay loop ...
        [    0.379955] pid_max: ...

    dmesg with this patch:

        [    0.000000] kvm-clock: ...
        [    0.000001] kvm-clock: ...
        [    0.000675] clocksource: ...
        [    0.002685] tsc: ...
        [    0.003331] e820: ...
        [    0.004190] e820: ...
         ...
        [    0.421939] ..TIMER: ...
        [    0.422842] clocksource: ...
        [    0.424582] Calibrating delay loop ...
        [    0.425580] pid_max: ...

    Fixes: 776f22913b8e ("sched/clock: Make local_clock() noinstr")
    Signed-off-by: Aaron Thompson <dev@aaront.org>
    Signed-off-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Link: https://lkml.kernel.org/r/20230413175012.2201-1-dev@aaront.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 91ba88989a633e50366f9497a1b2aa72932fa2bc
Author: Schspa Shi <schspa@gmail.com>
Date:   Mon Aug 29 01:03:02 2022 +0800

    sched/rt: Fix bad task migration for rt tasks

    [ Upstream commit feffe5bb274dd3442080ef0e4053746091878799 ]

    Commit 95158a89dd50 ("sched,rt: Use the full cpumask for balancing")
    allows find_lock_lowest_rq() to pick a task with migration disabled.
    The purpose of the commit is to push the current running task on the
    CPU that has the migrate_disable() task away.

    However, there is a race which allows a migrate_disable() task to be
    migrated. Consider:

      CPU0                                    CPU1
      push_rt_task
        check is_migration_disabled(next_task)
```

```
                                          task not running and
                                          migration_disabled == 0

        find_lock_lowest_rq(next_task, rq);
          _double_lock_balance(this_rq, busiest);
            raw_spin_rq_unlock(this_rq);
            double_rq_lock(this_rq, busiest);
              <<wait for busiest rq>>
                                              <wakeup>
                                          task become running
                                          migrate_disable();
                                            <context out>
        deactivate_task(rq, next_task, 0);
        set_task_cpu(next_task, lowest_rq->cpu);
          WARN_ON_ONCE(is_migration_disabled(p));
```

    Fixes: 95158a89dd50 ("sched,rt: Use the full cpumask for balancing")
    Signed-off-by: Schspa Shi <schspa@gmail.com>
    Signed-off-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Reviewed-by: Steven Rostedt (Google) <rostedt@goodmis.org>
    Reviewed-by: Dietmar Eggemann <dietmar.eggemann@arm.com>
    Reviewed-by: Valentin Schneider <vschneid@redhat.com>
    Tested-by: Dwaine Gonyier <dgonyier@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a65bb02daf53d7747004ffe64150164054163899
Author: Alexandre Ghiti <alexghiti@rivosinc.com>
Date: Fri Feb 3 08:52:31 2023 +0100

    riscv: Fix ptdump when KASAN is enabled

    [Upstream commit ecd7ebaf0b5a094a6180b299a5635c0eea42be4b]

    The KASAN shadow region was moved next to the kernel mapping but the
    ptdump code was not updated and it appears to break the dump of the kernel
    page table, so fix this by moving the KASAN shadow region in ptdump.

    Fixes: f7ae02333d13 ("riscv: Move KASAN mapping next to the kernel mapping")
    Signed-off-by: Alexandre Ghiti <alexghiti@rivosinc.com>
    Tested-by: Björn Töpel <bjorn@rivosinc.com>
    Reviewed-by: Björn Töpel <bjorn@rivosinc.com>
    Link: https://lore.kernel.org/r/20230203075232.274282-6-alexghiti@rivosinc.com
    Signed-off-by: Palmer Dabbelt <palmer@rivosinc.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 78ac06c7d3d840bed00f668e85af411b20292755
Author: Josh Poimboeuf <jpoimboe@kernel.org>
Date: Wed Apr 12 10:29:01 2023 -0700

    Revert "objtool: Support addition to set CFA base"

    [Upstream commit e18398e80c73e3cc7d9c3d2e0bc06a4af8f4f1cb]

    Commit 468af56a7bba ("objtool: Support addition to set CFA base") was
    added as a preparatory patch for arm64 support, but that support never
    came. It triggers a false positive warning on x86, so just revert it
    for now.

    Fixes the following warning:

      vmlinux.o: warning: objtool: cdce925_regmap_i2c_write+0xdb: stack state mismatch:
cfa1=4+120 cfa2=5+40

    Fixes: 468af56a7bba ("objtool: Support addition to set CFA base")
    Reported-by: kernel test robot <lkp@intel.com>

```
    Signed-off-by: Josh Poimboeuf <jpoimboe@kernel.org>
    Signed-off-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Link: https://lore.kernel.org/oe-kbuild-all/202304080538.j5G6h1AB-lkp@intel.com/
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 623c4ae2e227c8dfca4710665953ea7580b9c127
Author: Yang Jihong <yangjihong1@huawei.com>
Date:   Mon Feb 27 10:35:08 2023 +0800

    perf/core: Fix hardlockup failure caused by perf throttle

    [ Upstream commit 15def34e2635ab7e0e96f1bc32e1b69609f14942 ]

    commit e050e3f0a71bf ("perf: Fix broken interrupt rate throttling")
    introduces a change in throttling threshold judgment. Before this,
    compare hwc->interrupts and max_samples_per_tick, then increase
    hwc->interrupts by 1, but this commit reverses order of these two
    behaviors, causing the semantics of max_samples_per_tick to change.
    In literal sense of "max_samples_per_tick", if hwc->interrupts ==
    max_samples_per_tick, it should not be throttled, therefore, the judgment
    condition should be changed to "hwc->interrupts > max_samples_per_tick".

    In fact, this may cause the hardlockup to fail, The minimum value of
    max_samples_per_tick may be 1, in this case, the return value of
    __perf_event_account_interrupt function is 1.
    As a result, nmi_watchdog gets throttled, which would stop PMU (Use x86
    architecture as an example, see x86_pmu_handle_irq).

    Fixes: e050e3f0a71b ("perf: Fix broken interrupt rate throttling")
    Signed-off-by: Yang Jihong <yangjihong1@huawei.com>
    Signed-off-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Link: https://lkml.kernel.org/r/20230227023508.102230-1-yangjihong1@huawei.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6ac7b3946ec7bb9251ee722e03ad0ec4a526dca8
Author: Libo Chen <libo.chen@oracle.com>
Date:   Wed Aug 10 15:33:13 2022 -0700

    sched/fair: Fix inaccurate tally of ttwu_move_affine

    [ Upstream commit 39afe5d6fc59237ff7738bf3ede5a8856822d59d ]

    There are scenarios where non-affine wakeups are incorrectly counted as
    affine wakeups by schedstats.

    When wake_affine_idle() returns prev_cpu which doesn't equal to
    nr_cpumask_bits, it will slip through the check: target == nr_cpumask_bits
    in wake_affine() and be counted as if target == this_cpu in schedstats.

    Replace target == nr_cpumask_bits with target != this_cpu to make sure
    affine wakeups are accurately tallied.

    Fixes: 806486c377e33 (sched/fair: Do not migrate if the prev_cpu is idle)
    Suggested-by: Daniel Jordan <daniel.m.jordan@oracle.com>
    Signed-off-by: Libo Chen <libo.chen@oracle.com>
    Signed-off-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Reviewed-by: Gautham R. Shenoy <gautham.shenoy@amd.com>
    Link: https://lore.kernel.org/r/20220810223313.386614-1-libo.chen@oracle.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 10862dd52866abfb8423822ef00dd186d5a4a9ac
Author: Nathan Lynch <nathanl@linux.ibm.com>
Date:   Mon Mar 6 15:33:41 2023 -0600

    powerpc/rtas: use memmove for potentially overlapping buffer copy
```

```
    [ Upstream commit 271208ee5e335cb1ad280d22784940daf7ddf820 ]

    Using memcpy() isn't safe when buf is identical to rtas_err_buf, which
    can happen during boot before slab is up. Full context which may not
    be obvious from the diff:

            if (altbuf) {
                    buf = altbuf;
            } else {
                    buf = rtas_err_buf;
                    if (slab_is_available())
                            buf = kmalloc(RTAS_ERROR_LOG_MAX, GFP_ATOMIC);
            }
            if (buf)
                    memcpy(buf, rtas_err_buf, RTAS_ERROR_LOG_MAX);

    This was found by inspection and I'm not aware of it causing problems
    in practice. It appears to have been introduced by commit
    033ef338b6e0 ("powerpc: Merge rtas.c into arch/powerpc/kernel"); the
    old ppc64 version of this code did not have this problem.

    Use memmove() instead.

    Fixes: 033ef338b6e0 ("powerpc: Merge rtas.c into arch/powerpc/kernel")
    Signed-off-by: Nathan Lynch <nathanl@linux.ibm.com>
    Reviewed-by: Andrew Donnellan <ajd@linux.ibm.com>
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Link: https://msgid.link/20230220-rtas-queue-for-6-4-v1-2-010e4416f13f@linux.ibm.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cc0fc424fc44ef6f4adafc49986881e93110658a
Author: Randy Dunlap <rdunlap@infradead.org>
Date:   Wed Feb 22 17:42:41 2023 -0800

    macintosh: via-pmu-led: requires ATA to be set

    [ Upstream commit 05dce4ba125336875cd3eed3c1503fa81cd2f691 ]

    LEDS_TRIGGER_DISK depends on ATA, so selecting LEDS_TRIGGER_DISK
    when ATA is not set/enabled causes a Kconfig warning:

    WARNING: unmet direct dependencies detected for LEDS_TRIGGER_DISK
      Depends on [n]: NEW_LEDS [=y] && LEDS_TRIGGERS [=y] && ATA [=n]
      Selected by [y]:
      - ADB_PMU_LED_DISK [=y] && MACINTOSH_DRIVERS [=y] && ADB_PMU_LED [=y] && LEDS_CLASS
[=y]

    Fix this by making ADB_PMU_LED_DISK depend on ATA.

    Seen on both PPC32 and PPC64.

    Fixes: 0e865a80c135 ("macintosh: Remove dependency on IDE_GD_ATA if ADB_PMU_LED_DISK is
selected")
    Signed-off-by: Randy Dunlap <rdunlap@infradead.org>
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Link: https://msgid.link/20230223014241.20878-1-rdunlap@infradead.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3832f0bb594baff8038a7cd2329d7d7e39ef5a4f
Author: Randy Dunlap <rdunlap@infradead.org>
Date:   Wed Feb 22 23:01:16 2023 -0800

    powerpc/sysdev/tsi108: fix resource printk format warnings
```

```
    [ Upstream commit 55d8bd02cc1b9f1063993b5c42c9cabf4af67dea ]

    Use "%pa" format specifier for resource_size_t to avoid a compiler
    printk format warning.

      arch/powerpc/sysdev/tsi108_pci.c: In function 'tsi108_setup_pci':
      include/linux/kern_levels.h:5:25: error: format '%x' expects argument of type
'unsigned int', but argument 2 has type 'resource_size_t'

    Fixes: c4342ff92bed ("[POWERPC] Update mpc7448hpc2 board irq support using device tree")
    Fixes: 2b9d7467a6db ("[POWERPC] Add tsi108 pci and platform device data register
function")
    Signed-off-by: Randy Dunlap <rdunlap@infradead.org>
    [mpe: Use pr_info() and unsplit string]
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Link: https://msgid.link/20230223070116.660-5-rdunlap@infradead.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2effb1339c6686cc15f1f68edb3b524f85e1c1e0
Author: Randy Dunlap <rdunlap@infradead.org>
Date:   Wed Feb 22 23:01:14 2023 -0800

    powerpc/wii: fix resource printk format warnings

    [ Upstream commit 7b69600d4da0049244e9be2f5ef5a2f8e04fcd9a ]

    Use "%pa" format specifier for resource_size_t to avoid compiler
    printk format warnings.

    ../arch/powerpc/platforms/embedded6xx/flipper-pic.c: In function 'flipper_pic_init':
    ../include/linux/kern_levels.h:5:25: error: format '%x' expects argument of type
'unsigned int', but argument 2 has type 'resource_size_t' {aka 'long long unsigned int'} [-
Werror=format=]
    ../arch/powerpc/platforms/embedded6xx/flipper-pic.c:148:9: note: in expansion of macro
'pr_info'
      148 |         pr_info("controller at 0x%08x mapped to 0x%p\n", res.start, io_base);
          |         ^~~~~~

    ../arch/powerpc/platforms/embedded6xx/hlwd-pic.c: In function 'hlwd_pic_init':
    ../include/linux/kern_levels.h:5:25: error: format '%x' expects argument of type
'unsigned int', but argument 2 has type 'resource_size_t' {aka 'long long unsigned int'} [-
Werror=format=]
    ../arch/powerpc/platforms/embedded6xx/hlwd-pic.c:174:9: note: in expansion of macro
'pr_info'
      174 |         pr_info("controller at 0x%08x mapped to 0x%p\n", res.start, io_base);
          |         ^~~~~~

    ../arch/powerpc/platforms/embedded6xx/wii.c: In function 'wii_ioremap_hw_regs':
    ../include/linux/kern_levels.h:5:25: error: format '%x' expects argument of type
'unsigned int', but argument 3 has type 'resource_size_t' {aka 'long long unsigned int'} [-
Werror=format=]
    ../arch/powerpc/platforms/embedded6xx/wii.c:77:17: note: in expansion of macro 'pr_info'
       77 |                 pr_info("%s at 0x%08x mapped to 0x%p\n", name,
          |                 ^~~~~~

    Fixes: 028ee972f032 ("powerpc: gamecube/wii: flipper interrupt controller support")
    Fixes: 9c21025c7845 ("powerpc: wii: hollywood interrupt controller support")
    Fixes: 5a7ee3198dfa ("powerpc: wii: platform support")
    Signed-off-by: Randy Dunlap <rdunlap@infradead.org>
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Link: https://msgid.link/20230223070116.660-3-rdunlap@infradead.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 826ffa63e78cabce3cf6d5cd00e3a02fd12eb4cf
Author: Randy Dunlap <rdunlap@infradead.org>
```

```
Date:   Wed Feb 22 23:01:13 2023 -0800

    powerpc/mpc512x: fix resource printk format warning

    [ Upstream commit 7538c97e2b80ff6b7a8ea2ecf16a04355461b439 ]

    Use "%pa" format specifier for resource_size_t to avoid a compiler
    printk format warning.

    ../arch/powerpc/platforms/512x/clock-commonclk.c: In function
'mpc5121_clk_provide_backwards_compat':
    ../arch/powerpc/platforms/512x/clock-commonclk.c:989:44: error: format '%x' expects
argument of type 'unsigned int', but argument 4 has type 'resource_size_t' {aka 'long long
unsigned int'} [-Werror=format=]
      989 |         snprintf(devname, sizeof(devname), "%08x.%s", res.start, np->name); \
          |                                            ^~~~~~~~  ~~~~~~~~~
          |                                                     |
          |                                                     resource_size_t {aka
long long unsigned int}

    Prevents 24 such warnings.

    Fixes: 01f25c371658 ("clk: mpc512x: add backwards compat to the CCF code")
    Signed-off-by: Randy Dunlap <rdunlap@infradead.org>
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Link: https://msgid.link/20230223070116.660-2-rdunlap@infradead.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b3e2cc7af58fce4655864b1a24310bc3488803f8
Author: Christophe Leroy <christophe.leroy@csgroup.eu>
Date:   Mon Feb 27 16:54:45 2023 +0100

    powerpc/perf: Properly detect mpc7450 family

    [ Upstream commit e7299f961fe5e4496db0bfaa9e819f5e97f3846b ]

    Unlike PVR_POWER8, etc ...., PVR_7450 represents a full PVR
    value and not a family value.

    To avoid confusion, do like E500 family and define the relevant
    PVR_VER_xxxx values for the 7450 family:
      0x8000 ==> 7450
      0x8001 ==> 7455
      0x8002 ==> 7447
      0x8003 ==> 7447A
      0x8004 ==> 7448

    And use them to detect 7450 family for perf events.

    Reported-by: kernel test robot <lkp@intel.com>
    Reported-by: Dan Carpenter <error27@gmail.com>
    Link: https://lore.kernel.org/r/202302260657.7dM9Uwev-lkp@intel.com/
    Fixes: ec3eb9d941a9 ("powerpc/perf: Use PVR rather than oprofile field to determine CPU
version")
    Signed-off-by: Christophe Leroy <christophe.leroy@csgroup.eu>
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Link:
https://msgid.link/99ca1da2e5a6cf82a8abf4bc034918e500e31781.1677513277.git.christophe.leroy@
csgroup.eu
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f7d2f1e771952077ce63c02daaf2a9acee41fb98
Author: Liang He <windhl@126.com>
Date:   Thu Mar 30 11:35:58 2023 +0800
```

```
    macintosh/windfarm_smu_sat: Add missing of_node_put()

    [ Upstream commit 631cf002826007ab7415258ee647dcaf8845ad5a ]

    We call of_node_get() in wf_sat_probe() after sat is created,
    so we need the of_node_put() before *kfree(sat)*.

    Fixes: ac171c46667c ("[PATCH] powerpc: Thermal control for dual core G5s")
    Signed-off-by: Liang He <windhl@126.com>
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Link: https://msgid.link/20230330033558.2562778-1-windhl@126.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a27ee0e6b8729b33054a5eb07eb943f803a63d14
Author: Kajol Jain <kjain@linux.ibm.com>
Date:   Wed Mar 1 22:39:17 2023 +0530

    selftests/powerpc/pmu: Fix sample field check in the mmcra_thresh_marked_sample_test

    [ Upstream commit 8a32341cf04ba05974931b4664683c2c9fb84e56 ]

    The testcase verifies the setting of different fields in Monitor Mode
    Control Register A (MMCRA). In the current code, EV_CODE_EXTRACT macro
    is used to extract the "sample" field, which then needs to be further
    processed to fetch rand_samp_elig and rand_samp_mode bits. But the
    current code is not passing valid sample field to EV_CODE_EXTRACT
    macro. Patch addresses this by fixing the input for EV_CODE_EXTRACT.

    Fixes: 29cf373c5766 ("selftests/powerpc/pmu: Add interface test for mmcra register
fields")
    Reported-by: David Binderman <dcb314@hotmail.com>
    Link:
https://lore.kernel.org/r/DB6P189MB0568CF002762C6C43AF6DF169CA89@DB6P189MB0568.EURP189.PROD.
OUTLOOK.COM
    Signed-off-by: Kajol Jain <kjain@linux.ibm.com>
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Link: https://msgid.link/20230301170918.69176-1-kjain@linux.ibm.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f9325f702594c2b8958eaf9f7407dd318314c073
Author: Christophe JAILLET <christophe.jaillet@wanadoo.fr>
Date:   Thu Apr 13 21:33:17 2023 +0200

    fbdev: mmp: Fix deferred clk handling in mmphw_probe()

    [ Upstream commit b3a7a9ab65ae2f2626c7222fb79cdd433f8c5252 ]

    When dev_err_probe() is called, 'ret' holds the value of the previous
    successful devm_request_irq() call.
    'ret' should be assigned with a meaningful value before being used in
    dev_err_probe().

    While at it, use and return "PTR_ERR(ctrl->clk)" instead of a hard-coded
    "-ENOENT" so that -EPROBE_DEFER is handled and propagated correctly.

    Fixes: 81b63420564d ("fbdev: mmp: Make use of the helper function dev_err_probe()")
    Signed-off-by: Christophe JAILLET <christophe.jaillet@wanadoo.fr>
    Signed-off-by: Helge Deller <deller@gmx.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 97aff7ffb6f4c104d8b1e671d131929a1531f309
Author: Albert Huang <huangjie.albert@bytedance.com>
Date:   Wed Mar 29 18:23:00 2023 +0800

    virtio_ring: don't update event idx on get_buf
```

```
    [ Upstream commit 6c0b057cec5eade4c3afec3908821176931a9997 ]

    In virtio_net, if we disable napi_tx, when we trigger a tx interrupt,
    the vq->event_triggered will be set to true. It is then never reset
    until we explicitly call virtqueue_enable_cb_delayed or
    virtqueue_enable_cb_prepare.

    If we disable the napi_tx, virtqueue_enable_cb* will only be called when
    the tx ring is getting relatively empty.

    Since event_triggered is true, VRING_AVAIL_F_NO_INTERRUPT or
    VRING_PACKED_EVENT_FLAG_DISABLE will not be set. As a result we update
    vring_used_event(&vq->split.vring) or vq->packed.vring.driver->off_wrap
    every time we call virtqueue_get_buf_ctx. This causes more interrupts.

    To summarize:
    1) event_triggered was set to true in vring_interrupt()
    2) after this nothing will happen in virtqueue_disable_cb() so
       VRING_AVAIL_F_NO_INTERRUPT is not set in avail_flags_shadow
    3) virtqueue_get_buf_ctx_split() will still think the cb is enabled
       and then it will publish a new event index

    To fix:
    update VRING_AVAIL_F_NO_INTERRUPT or VRING_PACKED_EVENT_FLAG_DISABLE in
    the vq when we call virtqueue_disable_cb even when event_triggered is
    true.

    Tested with iperf:
    iperf3 tcp stream:
    vm1 -----------------> vm2
    vm2 just receives tcp data stream from vm1, and sends acks to vm1,
    there are many tx interrupts in vm2.
    with the patch applied there are just a few tx interrupts.

    v2->v3:
    -update the interrupt disable flag even with the event_triggered is set,
    -instead of checking whether event_triggered is set in
    -virtqueue_get_buf_ctx_{packed/split}, will cause the drivers  which have
    -not called virtqueue_{enable/disable}_cb to miss notifications.

    v3->v4:
    -remove change for
    -"if (vq->packed.event_flags_shadow != VRING_PACKED_EVENT_FLAG_DISABLE)"
    -in virtqueue_disable_cb_packed

    Fixes: 8d622d21d248 ("virtio: fix up virtio_disable_cb")
    Signed-off-by: Albert Huang <huangjie.albert@bytedance.com>
    Signed-off-by: Michael S. Tsirkin <mst@redhat.com>
    Signed-off-by: Jason Wang <jasowang@redhat.com>
    Message-Id: <20230329102300.61000-1-huangjie.albert@bytedance.com>
    Signed-off-by: Michael S. Tsirkin <mst@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit acd13a268ee8c8c332fb99888d9660baf5206228
Author: Eli Cohen <elic@nvidia.com>
Date:   Mon Apr 17 14:03:43 2023 +0300

    vdpa/mlx5: Avoid losing link state updates

    [ Upstream commit c384c2401eed99a2e1f84191e573f15b898babe6 ]

    Current code ignores link state updates if VIRTIO_NET_F_STATUS was not
    negotiated. However, link state updates could be received before feature
    negotiation was completed , therefore causing link state events to be
```

```
    lost, possibly leaving the link state down.

    Modify the code so link state notifier is registered after DRIVER_OK was
    negotiated and carry the registration only if
    VIRTIO_NET_F_STATUS was negotiated.  Unregister the notifier when the
    device is reset.

    Fixes: 033779a708f0 ("vdpa/mlx5: make MTU/STATUS presence conditional on feature bits")
    Acked-by: Jason Wang <jasowang@redhat.com>
    Signed-off-by: Eli Cohen <elic@nvidia.com>
    Message-Id: <20230417110343.138319-1-elic@nvidia.com>
    Signed-off-by: Michael S. Tsirkin <mst@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0f3ef30c1c05502f5de3b73b3715d5994845c1b4
Author: Jishnu Prakash <quic_jprakash@quicinc.com>
Date:   Thu Apr 13 15:38:34 2023 -0700

    spmi: Add a check for remove callback when removing a SPMI driver

    [ Upstream commit b56eef3e16d888883fefab47425036de80dd38fc ]

    When removing a SPMI driver, there can be a crash due to NULL pointer
    dereference if it does not have a remove callback defined. This is
    one such call trace observed when removing the QCOM SPMI PMIC driver:

     dump_backtrace.cfi_jt+0x0/0x8
     dump_stack_lvl+0xd8/0x16c
     panic+0x188/0x498
     __cfi_slowpath+0x0/0x214
     __cfi_slowpath+0x1dc/0x214
     spmi_drv_remove+0x16c/0x1e0
     device_release_driver_internal+0x468/0x79c
     driver_detach+0x11c/0x1a0
     bus_remove_driver+0xc4/0x124
     driver_unregister+0x58/0x84
     cleanup_module+0x1c/0xc24 [qcom_spmi_pmic]
     __do_sys_delete_module+0x3ec/0x53c
     __arm64_sys_delete_module+0x18/0x28
     el0_svc_common+0xdc/0x294
     el0_svc+0x38/0x9c
     el0_sync_handler+0x8c/0xf0
     el0_sync+0x1b4/0x1c0

    If a driver has all its resources allocated through devm_() APIs and
    does not need any other explicit cleanup, it would not require a
    remove callback to be defined. Hence, add a check for remove callback
    presence before calling it when removing a SPMI driver.

    Link: https://lore.kernel.org/r/1671601032-18397-2-git-send-email-
quic_jprakash@quicinc.com
    Fixes: 6f00f8c8635f ("mfd: qcom-spmi-pmic: Use devm_of_platform_populate()")
    Fixes: 5a86bf343976 ("spmi: Linux driver framework for SPMI")
    Signed-off-by: Jishnu Prakash <quic_jprakash@quicinc.com>
    Signed-off-by: Stephen Boyd <sboyd@kernel.org>
    Link: https://lore.kernel.org/r/20230413223834.4084793-7-sboyd@kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 525cb5a56b89abc606a2997ecb890e8777f84948
Author: Philipp Hortmann <philipp.g.hortmann@gmail.com>
Date:   Tue Apr 18 22:02:01 2023 +0200

    staging: rtl8192e: Fix W_DISABLE# does not work after stop/start
```

```
    [ Upstream commit 3fac2397f562eb669ddc2f45867a253f3fc26184 ]

    When loading the driver for rtl8192e, the W_DISABLE# switch is working as
    intended. But when the WLAN is turned off in software and then turned on
    again the W_DISABLE# does not work anymore. Reason for this is that in
    the function _rtl92e_dm_check_rf_ctrl_gpio() the bfirst_after_down is
    checked and returned when true. bfirst_after_down is set true when
    switching the WLAN off in software. But it is not set to false again
    when WLAN is turned on again.

    Add bfirst_after_down = false in _rtl92e_sta_up to reset bit and fix
    above described bug.

    Fixes: 94a799425eee ("From: wlanfae <wlanfae@realtek.com> [PATCH 1/8] rtl8192e: Import
new version of driver from realtek")
    Signed-off-by: Philipp Hortmann <philipp.g.hortmann@gmail.com>
    Link: https://lore.kernel.org/r/20230418200201.GA17398@matrix-ESPRIMO-P710
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e974588e1bc15ef94d0d110d554e135b7fd50f2d
Author: Dhruva Gole <d-gole@ti.com>
Date:   Thu Apr 20 11:12:57 2023 +0530

    spi: cadence-quadspi: use macro DEFINE_SIMPLE_DEV_PM_OPS

    [ Upstream commit be3206e8906e7a93df673ab2e96d69304a008edc ]

    Using this macro makes the code more readable.
    It also inits the members of dev_pm_ops in the following manner
    without us explicitly needing to:

    .suspend = cqspi_suspend, \
    .resume = cqspi_resume, \
    .freeze = cqspi_suspend, \
    .thaw = cqspi_resume, \
    .poweroff = cqspi_suspend, \
    .restore = cqspi_resume

    Also get rid of conditional compilation based on CONFIG_PM_SLEEP because
    it introduces build issues with certain configs when CQSPI_DEV_PM_OPS is
    just NULL.

    Reported-by: kernel test robot <lkp@intel.com>
    Link: https://lore.kernel.org/oe-kbuild-all/202304191900.2fARFQW9-lkp@intel.com/
    Fixes: 140623410536 ("mtd: spi-nor: Add driver for Cadence Quad SPI Flash Controller")
    Signed-off-by: Dhruva Gole <d-gole@ti.com>
    Link: https://lore.kernel.org/r/20230420054257.925092-1-d-gole@ti.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f5fd2fd999b364801e9790c6f69f3fe3f40ed60f
Author: Florian Fainelli <f.fainelli@gmail.com>
Date:   Fri Apr 14 10:02:39 2023 -0700

    serial: 8250: Add missing wakeup event reporting

    [ Upstream commit 0ba9e3a13c6adfa99e32b2576d20820ab10ad48a ]

    An 8250 UART configured as a wake-up source would not have reported
    itself through sysfs as being the source of wake-up, correct that.

    Fixes: b3b708fa2780 ("wake up from a serial port")
    Signed-off-by: Florian Fainelli <f.fainelli@gmail.com>
    Link: https://lore.kernel.org/r/20230414170241.2016255-1-f.fainelli@gmail.com
```

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

commit c6578bb745e22d6171663bb85ed367ad4075a821
Author: Shenwei Wang <shenwei.wang@nxp.com>
Date:   Mon Apr 10 14:55:55 2023 -0500

    tty: serial: fsl_lpuart: adjust buffer length to the intended size

    [ Upstream commit f73fd750552524b06b5d77ebfdd106ccc8fcac61 ]

    Based on the fls function definition provided below, we should not
    subtract 1 to obtain the correct buffer length:

    fls(0) = 0, fls(1) = 1, fls(0x80000000) = 32.

    Fixes: 5887ad43ee02 ("tty: serial: fsl_lpuart: Use cyclic DMA for Rx")
    Signed-off-by: Shenwei Wang <shenwei.wang@nxp.com>
    Link: https://lore.kernel.org/r/20230410195555.1003900-1-shenwei.wang@nxp.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 33c3570b763571cf1488c9dcda302704678cf528
Author: Dan Carpenter <dan.carpenter@linaro.org>
Date:   Wed Apr 19 17:27:03 2023 +0300

    firmware: stratix10-svc: Fix an NULL vs IS_ERR() bug in probe

    [ Upstream commit e1d6ca042e62c2a69513235f8629eb6e62ca79c5 ]

    The svc_create_memory_pool() function returns error pointers.  It never
    returns NULL.  Fix the check.

    Fixes: 7ca5ce896524 ("firmware: add Intel Stratix10 service layer driver")
    Signed-off-by: Dan Carpenter <dan.carpenter@linaro.org>
    Link: https://lore.kernel.org/r/5f9a8cb4-5a4f-460b-9cdc-2fae6c5b7922@kili.mountain
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3a7d4959560a2ee493ef222e3b63d359365f41ec
Author: Chunfeng Yun <chunfeng.yun@mediatek.com>
Date:   Mon Apr 17 10:51:59 2023 +0800

    usb: mtu3: fix kernel panic at qmu transfer done irq handler

    [ Upstream commit d28f4091ea7ec3510fd6a3c6d433234e7a2bef14 ]

    When handle qmu transfer irq, it will unlock @mtu->lock before give back
    request, if another thread handle disconnect event at the same time, and
    try to disable ep, it may lock @mtu->lock and free qmu ring, then qmu
    irq hanlder may get a NULL gpd, avoid the KE by checking gpd's value before
    handling it.

    e.g.
    qmu done irq on cpu0                thread running on cpu1

    qmu_done_tx()
      handle gpd [0]
        mtu3_requ_complete()        mtu3_gadget_ep_disable()
          unlock @mtu->lock
            give back request         lock @mtu->lock
                                        mtu3_ep_disable()
                                          mtu3_gpd_ring_free()
                                      unlock @mtu->lock
          lock @mtu->lock
```

```
        get next gpd [1]

    [1]: goto [0] to handle next gpd, and next gpd may be NULL.

    Fixes: 48e0d3735aa5 ("usb: mtu3: supports new QMU format")
    Signed-off-by: Chunfeng Yun <chunfeng.yun@mediatek.com>
    Link: https://lore.kernel.org/r/20230417025203.18097-3-chunfeng.yun@mediatek.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7df235809993072d771eeaa36bcb247f285eddd9
Author: Yinhao Hu <dddddd@hust.edu.cn>
Date:   Wed Apr 12 13:58:52 2023 +0800

    usb: chipidea: fix missing goto in `ci_hdrc_probe`

    [ Upstream commit d6f712f53b79f5017cdcefafb7a5aea9ec52da5d ]

    From the comment of ci_usb_phy_init, it returns an error code if
    usb_phy_init has failed, and it should do some clean up, not just
    return directly.

    Fix this by goto the error handling.

    Fixes: 74475ede784d ("usb: chipidea: move PHY operation to core")
    Reviewed-by: Dongliang Mu <dzm91@hust.edu.cn>
    Acked-by: Peter Chen <peter.chen@kernel.org>
    Signed-off-by: Yinhao Hu <dddddd@hust.edu.cn>
    Link: https://lore.kernel.org/r/20230412055852.971991-1-dddddd@hust.edu.cn
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 421e64d472318789fbbd45481fecf81db3f57e67
Author: Jon Hunter <jonathanh@nvidia.com>
Date:   Wed Apr 5 19:18:53 2023 +0100

    usb: gadget: tegra-xudc: Fix crash in vbus_draw

    [ Upstream commit 5629d31955297ca47b9283c64fff70f2f34aa528 ]

    Commit ac82b56bda5f ("usb: gadget: tegra-xudc: Add vbus_draw support")
    populated the vbus_draw callback for the Tegra XUDC driver. The function
    tegra_xudc_gadget_vbus_draw(), that was added by this commit, assumes
    that the pointer 'curr_usbphy' has been initialised, which is not always
    the case because this is only initialised when the USB role is updated.
    Fix this crash, by checking that the 'curr_usbphy' is valid before
    dereferencing.

    Fixes: ac82b56bda5f ("usb: gadget: tegra-xudc: Add vbus_draw support")
    Reviewed-by: Thierry Reding <treding@nvidia.com>
    Signed-off-by: Jon Hunter <jonathanh@nvidia.com>
    Link: https://lore.kernel.org/r/20230405181854.42355-1-jonathanh@nvidia.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1cee43150d3fd1d8a4f40d1072a3a679ee4f6c0e
Author: John Paul Adrian Glaubitz <glaubitz@physik.fu-berlin.de>
Date:   Wed Apr 19 13:48:52 2023 +0200

    sh: sq: Fix incorrect element size for allocating bitmap buffer

    [ Upstream commit 80f746e2bd0e1da3fdb49a53570e54a1a225faac ]

    The Store Queue code allocates a bitmap buffer with the size of
    multiple of sizeof(long) in sq_api_init(). While the buffer size
```

```
    is calculated correctly, the code uses the wrong element size to
    allocate the buffer which results in the allocated bitmap buffer
    being too small.

    Fix this by allocating the buffer with kcalloc() with element size
    sizeof(long) instead of kzalloc() whose elements size defaults to
    sizeof(char).

    Fixes: d7c30c682a27 ("sh: Store Queue API rework.")
    Reviewed-by: Geert Uytterhoeven <geert+renesas@glider.be>
    Signed-off-by: John Paul Adrian Glaubitz <glaubitz@physik.fu-berlin.de>
    Link: https://lore.kernel.org/r/20230419114854.528677-1-glaubitz@physik.fu-berlin.de
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit eb4d2f3cf031d814a43d4a65c77f9b99001d2873
Author: Kevin Brodsky <kevin.brodsky@arm.com>
Date:   Tue Apr 11 10:27:47 2023 +0100

    uapi/linux/const.h: prefer ISO-friendly __typeof__

    [ Upstream commit 31088f6f7906253ef4577f6a9b84e2d42447dba0 ]

    typeof is (still) a GNU extension, which means that it cannot be used when
    building ISO C (e.g.  -std=c99).  It should therefore be avoided in uapi
    headers in favour of the ISO-friendly __typeof__.

    Unfortunately this issue could not be detected by
    CONFIG_UAPI_HEADER_TEST=y as the __ALIGN_KERNEL() macro is not expanded in
    any uapi header.

    This matters from a userspace perspective, not a kernel one. uapi
    headers and their contents are expected to be usable in a variety of
    situations, and in particular when building ISO C applications (with
    -std=c99 or similar).

    This particular problem can be reproduced by trying to use the
    __ALIGN_KERNEL macro directly in application code, say:

    #include <linux/const.h>

    int align(int x, int a)
    {
            return __KERNEL_ALIGN(x, a);
    }

    and trying to build that with -std=c99.

    Link: https://lkml.kernel.org/r/20230411092747.3759032-1-kevin.brodsky@arm.com
    Fixes: a79ff731a1b2 ("netfilter: xtables: make XT_ALIGN() usable in exported headers by
exporting __ALIGN_KERNEL()")
    Signed-off-by: Kevin Brodsky <kevin.brodsky@arm.com>
    Reported-by: Ruben Ayrapetyan <ruben.ayrapetyan@arm.com>
    Tested-by: Ruben Ayrapetyan <ruben.ayrapetyan@arm.com>
    Reviewed-by: Petr Vorel <pvorel@suse.cz>
    Tested-by: Petr Vorel <pvorel@suse.cz>
    Reviewed-by: Masahiro Yamada <masahiroy@kernel.org>
    Cc: Sam Ravnborg <sam@ravnborg.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c2346532529b46d4352c3440b0d6372e0bf74173
Author: Florian Fainelli <f.fainelli@gmail.com>
Date:   Thu Apr 6 14:52:51 2023 -0700

    scripts/gdb: raise error with reduced debugging information
```

```
    [ Upstream commit 8af055ae25bff48f57227f5e3d48a4306f3dd1c4 ]

    If CONFIG_DEBUG_INFO_REDUCED is enabled in the kernel configuration, we
    will typically not be able to load vmlinux-gdb.py and will fail with:

    Traceback (most recent call last):
      File "/home/fainelli/work/buildroot/output/arm64/build/linux-custom/vmlinux-gdb.py",
line 25, in <module>
        import linux.utils
      File "/home/fainelli/work/buildroot/output/arm64/build/linux-
custom/scripts/gdb/linux/utils.py", line 131, in <module>
        atomic_long_counter_offset = atomic_long_type.get_type()['counter'].bitpos
    KeyError: 'counter'

    Rather be left wondering what is happening only to find out that reduced
    debug information is the cause, raise an eror.  This was not typically a
    problem until e3c8d33e0d62 ("scripts/gdb: fix 'lx-dmesg' on 32 bits arch")
    but it has since then.

    Link: https://lkml.kernel.org/r/20230406215252.1580538-1-f.fainelli@gmail.com
    Fixes: e3c8d33e0d62 ("scripts/gdb: fix 'lx-dmesg' on 32 bits arch")
    Signed-off-by: Florian Fainelli <f.fainelli@gmail.com>
    Cc: Antonio Borneo <antonio.borneo@foss.st.com>
    Cc: Jan Kiszka <jan.kiszka@siemens.com>
    Cc: John Ogness <john.ogness@linutronix.de>
    Cc: Kieran Bingham <kbingham@kernel.org>
    Cc: Petr Mladek <pmladek@suse.com>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 688fdfc458bfa651dca39c736d39c1b7520af0e8
Author: Lars-Peter Clausen <lars@metafoo.de>
Date:   Thu Apr 13 19:10:22 2023 -0700

    i2c: xiic: xiic_xfer(): Fix runtime PM leak on error path

    [ Upstream commit d663d93bb47e7ab45602b227701022d8aa16040a ]

    The xiic_xfer() function gets a runtime PM reference when the function is
    entered. This reference is released when the function is exited. There is
    currently one error path where the function exits directly, which leads to
    a leak of the runtime PM reference.

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

    Fixes: fdacc3c7405d ("i2c: xiic: Switch from waitqueue to completion")
    Signed-off-by: Lars-Peter Clausen <lars@metafoo.de>
    Reviewed-by: Michal Simek <michal.simek@amd.com>
    Signed-off-by: Wolfram Sang <wsa@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5b14d7c6ba0ba5d167f5ef588ca6dfe1af6dd0aa
Author: Lars-Peter Clausen <lars@metafoo.de>
Date:   Thu Apr 13 19:10:21 2023 -0700

    i2c: cadence: cdns_i2c_master_xfer(): Fix runtime PM leak on error path

    [ Upstream commit ae1664f04f504a998737f5bb563f16b44357bcca ]

    The cdns_i2c_master_xfer() function gets a runtime PM reference when the
    function is entered. This reference is released when the function is
    exited. There is currently one error path where the function exits
    directly, which leads to a leak of the runtime PM reference.
```

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

    Fixes: 1a351b10b967 ("i2c: cadence: Added slave support")
    Signed-off-by: Lars-Peter Clausen <lars@metafoo.de>
    Reviewed-by: Michal Simek <michal.simek@amd.com>
    Signed-off-by: Wolfram Sang <wsa@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1368dbc0a432acf9fc0dcb23bfe52d32ca4c09ab
Author: Dhruva Gole <d-gole@ti.com>
Date:   Mon Apr 17 14:40:27 2023 +0530

    spi: cadence-quadspi: fix suspend-resume implementations

    [ Upstream commit 2087e85bb66ee3652dafe732bb9b9b896229eafc ]

    The cadence QSPI driver misbehaves after performing a full system suspend
    resume:
    ...
    spi-nor spi0.0: resume() failed
    ...
    This results in a flash connected via OSPI interface after system suspend-
    resume to be unusable.
    fix these suspend and resume functions.

    Fixes: 140623410536 ("mtd: spi-nor: Add driver for Cadence Quad SPI Flash Controller")
    Signed-off-by: Dhruva Gole <d-gole@ti.com>
    Link: https://lore.kernel.org/r/20230417091027.966146-3-d-gole@ti.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d38568eec2063cf1ed3fdfa73d793e89001a2537
Author: Konrad Dybcio <konrad.dybcio@linaro.org>
Date:   Mon Apr 17 17:41:08 2023 +0200

    drm/panel: novatek-nt35950: Only unregister DSI1 if it exists

    [ Upstream commit a50be876f4fe2349dc8b056a49d87f69c944570f ]

    Commit 5dd45b66742a ("drm/panel: novatek-nt35950: Improve error handling")
    introduced logic to unregister DSI1 on any sort of probe failure, as
    that's not done automatically by kernel APIs.

    It did not however account for cases where only one DSI host is used.
    Fix that.

    Fixes: 5dd45b66742a ("drm/panel: novatek-nt35950: Improve error handling")
    Reported-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Signed-off-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Reviewed-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Signed-off-by: Neil Armstrong <neil.armstrong@linaro.org>
    Link: https://patchwork.freedesktop.org/patch/msgid/20230417-topic-maple_panel_fixup-v1-
1-07c8db606f5e@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4039b58e16b9153f1ee2d0ac4ab0672b2e963494
Author: Alex Williamson <alex.williamson@redhat.com>
Date:   Thu Apr 13 13:40:42 2023 -0600

    PCI/PM: Extend D3hot delay for NVIDIA HDA controllers

    [ Upstream commit a5a6dd2624698b6e3045c3a1450874d8c790d5d9 ]

    Assignment of NVIDIA Ampere-based GPUs have seen a regression since the
    below referenced commit, where the reduced D3hot transition delay appears
```

```
    to introduce a small window where a D3hot->D0 transition followed by a bus
    reset can wedge the device.  The entire device is subsequently unavailable,
    returning -1 on config space read and is unrecoverable without a host
    reset.

    This has been observed with RTX A2000 and A5000 GPU and audio functions
    assigned to a Windows VM, where shutdown of the VM places the devices in
    D3hot prior to vfio-pci performing a bus reset when userspace releases the
    devices.  The issue has roughly a 2-3% chance of occurring per shutdown.

    Restoring the HDA controller d3hot_delay to the effective value before the
    below commit has been shown to resolve the issue.  NVIDIA confirms this
    change should be safe for all of their HDA controllers.

    Fixes: 3e347969a577 ("PCI/PM: Reduce D3hot delay with usleep_range()")
    Link: https://lore.kernel.org/r/20230413194042.605768-1-alex.williamson@redhat.com
    Reported-by: Zhiyi Guo <zhguo@redhat.com>
    Signed-off-by: Alex Williamson <alex.williamson@redhat.com>
    Signed-off-by: Bjorn Helgaas <bhelgaas@google.com>
    Reviewed-by: Tarun Gupta <targupta@nvidia.com>
    Cc: Abhishek Sahu <abhsahu@nvidia.com>
    Cc: Tarun Gupta <targupta@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1bdb4a5ccab2316935ce4ad4fd4df8d36f0ffc6e
Author: Liliang Ye <yll@hust.edu.cn>
Date:   Mon Apr 3 23:26:47 2023 +0800

    ASoC: fsl_mqs: move of_node_put() to the correct location

    [ Upstream commit 1c34890273a020d61d6127ade3f68ed1cb21c16a ]

    of_node_put() should have been done directly after
    mqs_priv->regmap = syscon_node_to_regmap(gpr_np);
    otherwise it creates a reference leak on the success path.

    To fix this, of_node_put() is moved to the correct location, and change
    all the gotos to direct returns.

    Fixes: a9d273671440 ("ASoC: fsl_mqs: Fix error handling in probe")
    Signed-off-by: Liliang Ye <yll@hust.edu.cn>
    Reviewed-by: Dan Carpenter <error27@gmail.com>
    Link: https://lore.kernel.org/r/20230403152647.17638-1-yll@hust.edu.cn
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6f8c6abc8a11cc511da4c610f0f04aea7f1b0517
Author: Konrad Dybcio <konrad.dybcio@linaro.org>
Date:   Sat Apr 15 13:00:30 2023 +0200

    drm/panel: novatek-nt35950: Improve error handling

    [ Upstream commit 5dd45b66742a1f3cfa9a92dc0ac8714c7708ee6c ]

    In a very peculiar case when probing and registering with the secondary
    DSI host succeeds, but the OF backlight or DSI attachment fails, the
    primary DSI device is automatically cleaned up, but the secondary one
    is not, leading to -EEXIST when the driver core tries to handle
    -EPROBE_DEFER.

    Unregister the DSI1 device manually on failure to prevent that.

    Fixes: 623a3531e9cf ("drm/panel: Add driver for Novatek NT35950 DSI DriverIC panels")
    Signed-off-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Reviewed-by: Neil Armstrong <neil.armstrong@linaro.org>
```

```
    Signed-off-by: Neil Armstrong <neil.armstrong@linaro.org>
    Link: https://patchwork.freedesktop.org/patch/msgid/20230415-konrad-longbois-next-v1-1-
ce695dc9df84@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 61e38e7655ed5fc42e721332e905049e8859be85
Author: Suzuki K Poulose <suzuki.poulose@arm.com>
Date:   Wed Apr 5 10:49:22 2023 +0100

    coresight: etm_pmu: Set the module field

    [ Upstream commit 18996a113f2567aef3057e300e3193ce2df1684c ]

    struct pmu::module must be set to the module owning the PMU driver.
    Set this for the coresight etm_pmu.

    Fixes: 8e264c52e1dab ("coresight: core: Allow the coresight core driver to be built as a
module")
    Signed-off-by: Suzuki K Poulose <suzuki.poulose@arm.com>
    Link: https://lore.kernel.org/r/20230405094922.667834-1-suzuki.poulose@arm.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9cb5490b360a7d6ae67224f383355ba320838778
Author: Pierre Gondois <pierre.gondois@arm.com>
Date:   Fri Apr 14 10:14:50 2023 +0200

    cacheinfo: Check cache properties are present in DT

    [ Upstream commit cde0fbff07eff7e4e0e85fa053fe19a24c86b1e0 ]

    If a Device Tree (DT) is used, the presence of cache properties is
    assumed. Not finding any is not considered. For arm64 platforms,
    cache information can be fetched from the clidr_el1 register.
    Checking whether cache information is available in the DT
    allows to switch to using clidr_el1.

    init_of_cache_level()
    \-of_count_cache_leaves()
    will assume there a 2 cache leaves (L1 data/instruction caches), which
    can be different from clidr_el1 information.

    cache_setup_of_node() tries to read cache properties in the DT.
    If there are none, this is considered a success. Knowing no
    information was available would allow to switch to using clidr_el1.

    Fixes: de0df442ee49 ("cacheinfo: Check 'cache-unified' property to count cache leaves")
    Reported-by: Alexandre Ghiti <alexghiti@rivosinc.com>
    Link: https://lore.kernel.org/all/20230404-hatred-swimmer-6fecdf33b57a@spud/
    Signed-off-by: Pierre Gondois <pierre.gondois@arm.com>
    Reviewed-by: Conor Dooley <conor.dooley@microchip.com>
    Link: https://lore.kernel.org/r/20230414081453.244787-3-pierre.gondois@arm.com
    Signed-off-by: Sudeep Holla <sudeep.holla@arm.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 95fb61da302f58a0393b27feeb96ef41d76876c1
Author: Pierre Gondois <pierre.gondois@arm.com>
Date:   Fri Apr 14 10:14:49 2023 +0200

    cacheinfo: Check sib_leaf in cache_leaves_are_shared()

    [ Upstream commit 7a306e3eabf2b2fd8cffa69b87b32dbf814d79ce ]

    If there is no ACPI/DT information, it is assumed that L1 caches
    are private and L2 (and higher) caches are shared. A cache is
    'shared' between two CPUs if it is accessible from these two
```

```
    CPUs.

    Each CPU owns a representation (i.e. has a dedicated cacheinfo struct)
    of the caches it has access to. cache_leaves_are_shared() tries to
    identify whether two representations are designating the same actual
    cache.

    In cache_leaves_are_shared(), if 'this_leaf' is a L2 cache (or higher)
    and 'sib_leaf' is a L1 cache, the caches are detected as shared as
    only this_leaf's cache level is checked.
    This is leads to setting sib_leaf as being shared with another CPU,
    which is incorrect as this is a L1 cache.

    Check 'sib_leaf->level'. Also update the comment as the function is
    called when populating 'shared_cpu_map'.

    Fixes: f16d1becf96f ("cacheinfo: Use cache identifiers to check if the caches are shared
if available")
    Signed-off-by: Pierre Gondois <pierre.gondois@arm.com>
    Reviewed-by: Conor Dooley <conor.dooley@microchip.com>
    Link: https://lore.kernel.org/r/20230414081453.244787-2-pierre.gondois@arm.com
    Signed-off-by: Sudeep Holla <sudeep.holla@arm.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 33d347b3d7e4b5f5bdda6d871add3203efae2617
Author: Basavaraj Natikar <Basavaraj.Natikar@amd.com>
Date:   Tue Apr 11 21:40:30 2023 +0530

    HID: amd_sfh: Handle "no sensors" enabled for SFH1.1

    [ Upstream commit 8455cbb25927013b3417ab619dced1c0e87708af ]

    Based on num_hid_devices, each sensor device is initialized. If
    "no sensors" is initialized, amd_sfh work initialization and scheduling
    doesn’t make sense and returns EOPNOTSUPP to stop driver probe. Hence,
    add a check for "no sensors" enabled to handle the special case.

    Fixes: 93ce5e0231d7 ("HID: amd_sfh: Implement SFH1.1 functionality")
    Signed-off-by: Basavaraj Natikar <Basavaraj.Natikar@amd.com>
    Signed-off-by: Jiri Kosina <jkosina@suse.cz>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 86a06ac66fe282d67b6f081cc2df342486d24b53
Author: Basavaraj Natikar <Basavaraj.Natikar@amd.com>
Date:   Tue Apr 11 21:40:29 2023 +0530

    HID: amd_sfh: Increase sensor command timeout for SFH1.1

    [ Upstream commit 571dc8f59dd477037bb5a029e8d1b5a4a4d9dd63 ]

    The initialization of SFH1.1 sensors may take some time. Hence, increase
    sensor command timeouts in order to obtain status responses within a
    maximum timeout.

    Fixes: 93ce5e0231d7 ("HID: amd_sfh: Implement SFH1.1 functionality")
    Signed-off-by: Basavaraj Natikar <Basavaraj.Natikar@amd.com>
    Signed-off-by: Jiri Kosina <jkosina@suse.cz>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2ccae855842765216e8129a975b51111e684c46b
Author: Basavaraj Natikar <Basavaraj.Natikar@amd.com>
Date:   Tue Apr 11 21:40:28 2023 +0530

    HID: amd_sfh: Correct the stop all command
```

```
      [ Upstream commit 82c2a0d137794f5ef47982231593a00aee26ce3b ]

      Misinterpreted the stop all command in SHF1.1 firmware. Therefore, it is
      necessary to update the stop all command accordingly to disable all
      sensors.

      Fixes: 93ce5e0231d7 ("HID: amd_sfh: Implement SFH1.1 functionality")
      Signed-off-by: Basavaraj Natikar <Basavaraj.Natikar@amd.com>
      Signed-off-by: Jiri Kosina <jkosina@suse.cz>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 863ba993ee22320b7633f9002676dbde364886a4
  Author: Basavaraj Natikar <Basavaraj.Natikar@amd.com>
  Date:   Tue Apr 11 21:40:27 2023 +0530

      HID: amd_sfh: Add support for shutdown operation

      [ Upstream commit 1353ecaf1830d6d1b74f3225378a9498b4e14fdd ]

      As soon as the system is booted after shutdown, the sensors may remain in
      a weird state and fail to initialize. Therefore, all sensors should be
      turned off during shutdown.

      Fixes: 4f567b9f8141 ("SFH: PCIe driver to add support of AMD sensor fusion hub")
      Signed-off-by: Basavaraj Natikar <Basavaraj.Natikar@amd.com>
      Signed-off-by: Jiri Kosina <jkosina@suse.cz>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 7e1e71d97bb9a944dcb76d2a0539233887464571
  Author: Basavaraj Natikar <Basavaraj.Natikar@amd.com>
  Date:   Tue Apr 11 21:40:26 2023 +0530

      HID: amd_sfh: Fix illuminance value

      [ Upstream commit a33e5e393171ae8384d3381db5cd159ba877cfcb ]

      Illuminance value is actually 32 bits, but is incorrectly trancated to
      16 bits. Hence convert to integer illuminace accordingly to reflect
      correct values.

      Fixes: 93ce5e0231d7 ("HID: amd_sfh: Implement SFH1.1 functionality")
      Signed-off-by: Basavaraj Natikar <Basavaraj.Natikar@amd.com>
      Signed-off-by: Jiri Kosina <jkosina@suse.cz>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit dfa276913370ff55e8608f7b2e7a2d4a449a90af
  Author: Basavaraj Natikar <Basavaraj.Natikar@amd.com>
  Date:   Tue Apr 11 21:40:25 2023 +0530

      HID: amd_sfh: Correct the sensor enable and disable command

      [ Upstream commit 0b9255bf11baa61cd526e6bd24d6c8e6d1eabf8d ]

      In order to start or stop sensors, the firmware command needs to be
      changed to add an additional default subcommand value. For this reason,
      add a subcommand value to enable or disable sensors accordingly.

      Fixes: 93ce5e0231d7 ("HID: amd_sfh: Implement SFH1.1 functionality")
      Signed-off-by: Basavaraj Natikar <Basavaraj.Natikar@amd.com>
      Signed-off-by: Jiri Kosina <jkosina@suse.cz>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit eb8c6b514d761e4490729e92918ae2d6ae24ea85
  Author: Basavaraj Natikar <Basavaraj.Natikar@amd.com>
  Date:   Tue Apr 11 21:40:24 2023 +0530
```

```
    HID: amd_sfh: Correct the structure fields

    [ Upstream commit 7e7fdab79899f62de39c9280fb78f3d3b02ac207 ]

    Misinterpreted sfh_cmd_base structure member fields. Therefore, adjust
    the structure member fields accordingly to reflect functionality.

    Fixes: 93ce5e0231d7 ("HID: amd_sfh: Implement SFH1.1 functionality")
    Signed-off-by: Basavaraj Natikar <Basavaraj.Natikar@amd.com>
    Signed-off-by: Jiri Kosina <jkosina@suse.cz>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3e40722d55805584dc04d8594d912820cafb2432
Author: Aashish Sharma <shraash@google.com>
Date:   Tue Apr 11 06:04:31 2023 +0530

    ASoC: mediatek: common: Fix refcount leak in parse_dai_link_info

    [ Upstream commit beed115c2ce78f990222a29abed042582df4e87c ]

    Add missing of_node_put()s before the returns to balance
    of_node_get()s and of_node_put()s, which may get unbalanced
    in case the for loop 'for_each_available_child_of_node' returns
    early.

    Fixes: 4302187d955f ("ASoC: mediatek: common: add soundcard driver common code")
    Reported-by: kernel test robot <lkp@intel.com>
    Reported-by: Julia Lawall <julia.lawall@inria.fr>
    Link: https://lore.kernel.org/r/202304090504.2K8L6soj-lkp@intel.com/
    Signed-off-by: Aashish Sharma <shraash@google.com>
    Reviewed-by: Guenter Roeck <groeck@chromium.org>
    Reviewed-by: Trevor Wu <trevor.wu@mediatek.com>
    Link: https://lore.kernel.org/r/20230411003431.4048700-1-shraash@google.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4c9d343f9e9d0868dc28ba210a1ee1347ef44dc6
Author: Florian Fainelli <f.fainelli@gmail.com>
Date:   Thu Mar 23 16:16:57 2023 -0700

    scripts/gdb: bail early if there are no generic PD

    [ Upstream commit f19c3c2959e465209ade1a7a699e6cbf4359ce78 ]

    Avoid generating an exception if there are no generic power domain(s)
    registered:

    (gdb) lx-genpd-summary
    domain                          status         children
        /device                                             runtime status
    ----------------------------------------------------------------------
    Python Exception <class 'gdb.error'>: No symbol "gpd_list" in current context.
    Error occurred in Python: No symbol "gpd_list" in current context.
    (gdb) quit

    [f.fainelli@gmail.com: correctly invoke gdb_eval_or_none]
      Link: https://lkml.kernel.org/r/20230327185746.3856407-1-f.fainelli@gmail.com
    Link: https://lkml.kernel.org/r/20230323231659.3319941-1-f.fainelli@gmail.com
    Fixes: 8207d4a88e1e ("scripts/gdb: add lx-genpd-summary command")
    Signed-off-by: Florian Fainelli <f.fainelli@gmail.com>
    Cc: Jan Kiszka <jan.kiszka@siemens.com>
    Cc: Kieran Bingham <kbingham@kernel.org>
    Cc: Leonard Crestez <leonard.crestez@nxp.com>
    Cc: Stephen Boyd <sboyd@kernel.org>
```

```
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit bfb7b98d2d167a598e4adfed65c20da4ad673848
Author: Florian Fainelli <f.fainelli@gmail.com>
Date:   Thu Mar 23 15:52:45 2023 -0700

    scripts/gdb: bail early if there are no clocks

    [ Upstream commit 1d7adbc74c009057ed9dc3112f388e91a9c79acc ]

    Avoid generating an exception if there are no clocks registered:

    (gdb) lx-clk-summary
                                  enable  prepare  protect
       clock                       count    count    count        rate
    ----------------------------------------------------------------------
    Python Exception <class 'gdb.error'>: No symbol "clk_root_list" in
    current context.
    Error occurred in Python: No symbol "clk_root_list" in current context.

    Link: https://lkml.kernel.org/r/20230323225246.3302977-1-f.fainelli@gmail.com
    Fixes: d1e9710b63d8 ("scripts/gdb: initial clk support: lx-clk-summary")
    Signed-off-by: Florian Fainelli <f.fainelli@gmail.com>
    Cc: Jan Kiszka <jan.kiszka@siemens.com>
    Cc: Kieran Bingham <kbingham@kernel.org>
    Cc: Leonard Crestez <leonard.crestez@nxp.com>
    Cc: Stephen Boyd <sboyd@kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 57632904886ddeb86c7a9a67e418786a5017543d
Author: Randy Dunlap <rdunlap@infradead.org>
Date:   Wed Feb 22 19:43:09 2023 -0800

    ia64: salinfo: placate defined-but-not-used warning

    [ Upstream commit 0de155752b152d6bcd96b5b5bf20af336abd183a ]

    When CONFIG_PROC_FS is not set, proc_salinfo_show() is not used.  Mark the
    function as __maybe_unused to quieten the warning message.

    ../arch/ia64/kernel/salinfo.c:584:12: warning: 'proc_salinfo_show' defined but not used
[-Wunused-function]
      584 | static int proc_salinfo_show(struct seq_file *m, void *v)
          |            ^~~~~~~~~~~~~~~~~

    Link: https://lkml.kernel.org/r/20230223034309.13375-1-rdunlap@infradead.org
    Fixes: 3f3942aca6da ("proc: introduce proc_create_single{,_data}")
    Signed-off-by: Randy Dunlap <rdunlap@infradead.org>
    Cc: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0338ea4d0b22959b6ed273c63c047c59cc15c48d
Author: Randy Dunlap <rdunlap@infradead.org>
Date:   Wed Feb 22 19:42:58 2023 -0800

    ia64: mm/contig: fix section mismatch warning/error

    [ Upstream commit 58deeb4ef3b054498747d0929d94ac53ab90981f ]

    alloc_per_cpu_data() is called by find_memory(), which is marked as
    __init.  Therefore alloc_per_cpu_data() can also be marked as __init to
    remedy this modpost problem.
```

```
    WARNING: modpost: vmlinux.o: section mismatch in reference: alloc_per_cpu_data (section:
.text) -> memblock_alloc_try_nid (section: .init.text)

    Link: https://lkml.kernel.org/r/20230223034258.12917-1-rdunlap@infradead.org
    Fixes: 4b9ddc7cf272 ("[IA64] Fix section mismatch in contig.c version of
per_cpu_init()")
    Signed-off-by: Randy Dunlap <rdunlap@infradead.org>
    Cc: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9288c4f1dc860d093098f8667f740c6bb332eee9
Author: Kuppuswamy Sathyanarayanan <sathyanarayanan.kuppuswamy@linux.intel.com>
Date:   Wed Mar 15 16:54:49 2023 -0700

    PCI/EDR: Clear Device Status after EDR error recovery

    [ Upstream commit c441b1e03da6c680a3e12da59c554f454f2ccf5e ]

    During EDR recovery, the OS must clear error status of the port that
    triggered DPC even if firmware retains control of DPC and AER (see the
    implementation note in the PCI Firmware spec r3.3, sec 4.6.12).

    Prior to 068c29a248b6 ("PCI/ERR: Clear PCIe Device Status errors only if
    OS owns AER"), the port Device Status was cleared in this path:

      edr_handle_event
        dpc_process_error(dev)                  # "dev" triggered DPC
        pcie_do_recovery(dev, dpc_reset_link)
          dpc_reset_link                        # exit DPC
          pcie_clear_device_status(dev)         # clear Device Status

    After 068c29a248b6, pcie_do_recovery() no longer clears Device Status when
    firmware controls AER, so the error bit remains set even after recovery.

    Per the "Downstream Port Containment configuration control" bit in the
    returned _OSC Control Field (sec 4.5.1), the OS is allowed to clear error
    status until it evaluates _OST, so clear Device Status in
    edr_handle_event() if the error recovery was successful.

    [bhelgaas: commit log]
    Fixes: 068c29a248b6 ("PCI/ERR: Clear PCIe Device Status errors only if OS owns AER")
    Link: https://lore.kernel.org/r/20230315235449.1279209-1-
sathyanarayanan.kuppuswamy@linux.intel.com
    Reported-by: Tsaur Erwin <erwin.tsaur@intel.com>
    Signed-off-by: Kuppuswamy Sathyanarayanan <sathyanarayanan.kuppuswamy@linux.intel.com>
    Signed-off-by: Bjorn Helgaas <bhelgaas@google.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d11dd69f79aeac459d9705e2df8b6b8b41a06122
Author: Miquel Raynal <miquel.raynal@bootlin.com>
Date:   Tue Apr 4 18:21:09 2023 +0100

    of: Fix modalias string generation

    [ Upstream commit b19a4266c52de78496fe40f0b37580a3b762e67d ]

    The helper generating an OF based modalias (of_device_get_modalias())
    works fine, but due to the use of snprintf() internally it needs a
    buffer one byte longer than what should be needed just for the entire
    string (excluding the '\0'). Most users of this helper are sysfs hooks
    providing the modalias string to users. They all provide a PAGE_SIZE
    buffer which is way above the number of bytes required to fit the
    modalias string and hence do not suffer from this issue.
```

```
    There is another user though, of_device_request_module(), which is only
    called by drivers/usb/common/ulpi.c. This request module function is
    faulty, but maybe because in most cases there is an alternative, ULPI
    driver users have not noticed it.

    In this function, of_device_get_modalias() is called twice. The first
    time without buffer just to get the number of bytes required by the
    modalias string (excluding the null byte), and a second time, after
    buffer allocation, to fill the buffer. The allocation asks for an
    additional byte, in order to store the trailing '\0'. However, the
    buffer *length* provided to of_device_get_modalias() excludes this extra
    byte. The internal use of snprintf() with a length that is exactly the
    number of bytes to be written has the effect of using the last available
    byte to store a '\0', which then smashes the last character of the
    modalias string.

    Provide the actual size of the buffer to of_device_get_modalias() to fix
    this issue.

    Note: the "str[size - 1] = '\0';" line is not really needed as snprintf
    will anyway end the string with a null byte, but there is a possibility
    that this function might be called on a struct device_node without
    compatible, in this case snprintf() would not be executed. So we keep it
    just to avoid possible unbounded strings.

    Cc: Stephen Boyd <sboyd@kernel.org>
    Cc: Peter Chen <peter.chen@kernel.org>
    Fixes: 9c829c097f2f ("of: device: Support loading a module with OF based modalias")
    Signed-off-by: Miquel Raynal <miquel.raynal@bootlin.com>
    Reviewed-by: Rob Herring <robh@kernel.org>
    Signed-off-by: Srinivas Kandagatla <srinivas.kandagatla@linaro.org>
    Link: https://lore.kernel.org/r/20230404172148.82422-2-srinivas.kandagatla@linaro.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ab64bd32b9fac27ff4737d63711b9db5e5462448
Author: Dae R. Jeong <threeearcat@gmail.com>
Date:   Mon Mar 27 21:01:53 2023 +0900

    vmci_host: fix a race condition in vmci_host_poll() causing GPF

    [ Upstream commit ae13381da5ff0e8e084c0323c3cc0a945e43e9c7 ]

    During fuzzing, a general protection fault is observed in
    vmci_host_poll().

    general protection fault, probably for non-canonical address 0xdffffc0000000019: 0000
[#1] PREEMPT SMP KASAN
    KASAN: null-ptr-deref in range [0x00000000000000c8-0x00000000000000cf]
    RIP: 0010:__lock_acquire+0xf3/0x5e00 kernel/locking/lockdep.c:4926
    <- omitting registers ->
    Call Trace:
     <TASK>
     lock_acquire+0x1a4/0x4a0 kernel/locking/lockdep.c:5672
     __raw_spin_lock_irqsave include/linux/spinlock_api_smp.h:110 [inline]
     _raw_spin_lock_irqsave+0xb3/0x100 kernel/locking/spinlock.c:162
     add_wait_queue+0x3d/0x260 kernel/sched/wait.c:22
     poll_wait include/linux/poll.h:49 [inline]
     vmci_host_poll+0xf8/0x2b0 drivers/misc/vmw_vmci/vmci_host.c:174
     vfs_poll include/linux/poll.h:88 [inline]
     do_pollfd fs/select.c:873 [inline]
     do_poll fs/select.c:921 [inline]
     do_sys_poll+0xc7c/0x1aa0 fs/select.c:1015
     __do_sys_ppoll fs/select.c:1121 [inline]
```

```
      __se_sys_ppoll+0x2cc/0x330 fs/select.c:1101
      do_syscall_x64 arch/x86/entry/common.c:51 [inline]
      do_syscall_64+0x4e/0xa0 arch/x86/entry/common.c:82
      entry_SYSCALL_64_after_hwframe+0x46/0xb0

    Example thread interleaving that causes the general protection fault
    is as follows:

    CPU1 (vmci_host_poll)               CPU2 (vmci_host_do_init_context)
    -----                               -----
    // Read uninitialized context
    context = vmci_host_dev->context;
                                        // Initialize context
                                        vmci_host_dev->context = vmci_ctx_create();
                                        vmci_host_dev->ct_type = VMCIOBJ_CONTEXT;

    if (vmci_host_dev->ct_type == VMCIOBJ_CONTEXT) {
        // Dereferencing the wrong pointer
        poll_wait(..., &context->host_context);
    }

    In this scenario, vmci_host_poll() reads vmci_host_dev->context first,
    and then reads vmci_host_dev->ct_type to check that
    vmci_host_dev->context is initialized. However, since these two reads
    are not atomically executed, there is a chance of a race condition as
    described above.

    To fix this race condition, read vmci_host_dev->context after checking
    the value of vmci_host_dev->ct_type so that vmci_host_poll() always
    reads an initialized context.

    Reported-by: Dae R. Jeong <threeearcat@gmail.com>
    Fixes: 8bf503991f87 ("VMCI: host side driver implementation.")
    Signed-off-by: Dae R. Jeong <threeearcat@gmail.com>
    Link: https://lore.kernel.org/r/ZCGFsdBAU4cYww5l@dragonet
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6f4a5877622b85bb94582d733d99620a67c33a55
Author: Christophe Leroy <christophe.leroy@csgroup.eu>
Date:   Sat Apr 1 19:59:46 2023 +0200

    spi: fsl-spi: Fix CPM/QE mode Litte Endian

    [ Upstream commit c20c57d9868d7f9fd1b2904c7801b07e128f6322 ]

    CPM has the same problem as QE so for CPM also use the fix added
    by commit 0398fb70940e ("spi/spi_mpc8xxx: Fix QE mode Litte Endian"):

      CPM mode uses Little Endian so words > 8 bits are byte swapped.
      Workaround this by always enforcing wordsize 8 for 16 and 32 bits
      words. Unfortunately this will not work for LSB transfers
      where wordsize is > 8 bits so disable these for now.

    Also limit the workaround to 16 and 32 bits words because it can
    only work for multiples of 8-bits.

    Signed-off-by: Christophe Leroy <christophe.leroy@csgroup.eu>
    Cc: Joakim Tjernlund <Joakim.Tjernlund@infinera.com>
    Fixes: 0398fb70940e ("spi/spi_mpc8xxx: Fix QE mode Litte Endian")
    Link:
https://lore.kernel.org/r/1b7d3e84b1128f42c1887dd2fb9cdf390f541bc1.1680371809.git.christophe
.leroy@csgroup.eu
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 9f7efe497ec8f406a79d6de256634f6127f266f1
Author: Tharun Kumar P <tharunkumar.pasumarthi@microchip.com>
Date:   Tue Apr 4 22:46:13 2023 +0530

    spi: mchp-pci1xxxx: Fix improper implementation of disabling chip select lines

    [ Upstream commit 45d2af82e0e6f662d0d0db20993b35cb1d8da646 ]

    Hardware does not have support to disable individual chip select lines.
    Disable all chip select lines by using SPI_FORCE_CE bit.

    Fixes: 1cc0cbea7167 ("spi: microchip: pci1xxxx: Add driver for SPI controller of
PCI1XXXX PCIe switch")
    Signed-off-by: Tharun Kumar P <tharunkumar.pasumarthi@microchip.com>
    Link: https://lore.kernel.org/r/20230404171613.1336093-4-
tharunkumar.pasumarthi@microchip.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit d7f4ff0458fc81cbdb7f3eca24bb238d429d066e
Author: Rob Herring <robh@kernel.org>
Date:   Fri Feb 17 14:45:40 2023 -0600

    spi: mpc5xxx-psc: Remove unused platform_data

    [ Upstream commit 04725901d9933b3134e6dee6b5bc1efb67f8d43f ]

    The platform_data for the MPC5xxx PSC SPI controllers is never used, so
    remove it and the resulting code which depends on it.

    Signed-off-by: Rob Herring <robh@kernel.org>
    Link: https://lore.kernel.org/r/20230217-dt-mpc5xxx-spi-v1-1-3be8602fce1e@kernel.org
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Stable-dep-of: 45d2af82e0e6 ("spi: mchp-pci1xxxx: Fix improper implementation of
disabling chip select lines")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e5e0375798f0f8b02d11c0facb320e83ba1a9d75
Author: Tharun Kumar P <tharunkumar.pasumarthi@microchip.com>
Date:   Tue Apr 4 22:46:12 2023 +0530

    spi: mchp-pci1xxxx: Fix SPI transactions not working after suspend and resume

    [ Upstream commit 4266d21669de62cf3fb6774f7d404c1eb95a5ab3 ]

    pci1xxxx_spi_resume API masks SPI interrupt bit which prohibits interrupt
    from coming to the host at the end of the transaction after suspend-resume.
    This patch unmasks this bit at resume.

    Fixes: 1cc0cbea7167 ("spi: microchip: pci1xxxx: Add driver for SPI controller of
PCI1XXXX PCIe switch")
    Signed-off-by: Tharun Kumar P <tharunkumar.pasumarthi@microchip.com>
    Link: https://lore.kernel.org/r/20230404171613.1336093-3-
tharunkumar.pasumarthi@microchip.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit fa22dca65f4c0893d5f9c14912d3ffa723d736f6
Author: Tharun Kumar P <tharunkumar.pasumarthi@microchip.com>
Date:   Tue Apr 4 22:46:11 2023 +0530

    spi: mchp-pci1xxxx: Fix length of SPI transactions not set properly in driver
```

```
    [ Upstream commit 35c8c5e503a82e0a4bf251d32096211eba8c2be6 ]

    In pci1xxxx_spi_transfer_one API, length of SPI transaction gets cleared
    by setting of length mask. Set length of transaction only after masking
    length field.

    Fixes: 1cc0cbea7167 ("spi: microchip: pci1xxxx: Add driver for SPI controller of
PCI1XXXX PCIe switch")
    Signed-off-by: Tharun Kumar P <tharunkumar.pasumarthi@microchip.com>
    Link: https://lore.kernel.org/r/20230404171613.1336093-2-
tharunkumar.pasumarthi@microchip.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a0a74928984977f875a016b1564bddc007b8ace7
Author: Johan Hovold <johan+linaro@kernel.org>
Date:   Mon Mar 13 09:49:53 2023 +0100

    interconnect: qcom: rpm: drop bogus pm domain attach

    [ Upstream commit 72b2720c18ecde92e6a36c4ac897dd5848e3f379 ]

    Any power domain would already have been attached by the platform bus
    code so drop the bogus power domain attach which always succeeds from
    probe.

    This effectively reverts commit 7de109c0abe9 ("interconnect: icc-rpm:
    Add support for bus power domain").

    Fixes: 7de109c0abe9 ("interconnect: icc-rpm: Add support for bus power domain")
    Cc: Yassine Oudjana <y.oudjana@protonmail.com>
    Signed-off-by: Johan Hovold <johan+linaro@kernel.org>
    Tested-by: Konrad Dybcio <konrad.dybcio@linaro.org> # MSM8996 Sony Kagura
    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Link: https://lore.kernel.org/r/20230313084953.24088-3-johan+linaro@kernel.org
    Signed-off-by: Georgi Djakov <djakov@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 49c17fccae36505550c9121891722fff337f148a
Author: Uwe Kleine-König <u.kleine-koenig@pengutronix.de>
Date:   Thu Mar 30 23:03:40 2023 +0200

    spi: qup: Don't skip cleanup in remove's error path

    [ Upstream commit 61f49171a43ab1f80c73c5c88c508770c461e0f2 ]

    Returning early in a platform driver's remove callback is wrong. In this
    case the dma resources are not released in the error path. this is never
    retried later and so this is a permanent leak. To fix this, only skip
    hardware disabling if waking the device fails.

    Fixes: 64ff247a978f ("spi: Add Qualcomm QUP SPI controller support")
    Signed-off-by: Uwe Kleine-König <u.kleine-koenig@pengutronix.de>
    Link: https://lore.kernel.org/r/20230330210341.2459548-2-u.kleine-koenig@pengutronix.de
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6a860009ec21c4412fe92d3ddfc748292ec69e12
Author: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
Date:   Fri Mar 17 16:57:12 2023 +0100

    dt-bindings: serial: snps-dw-apb-uart: correct number of DMAs

    [ Upstream commit dedd376e0c7096daf4171d54957a679b4dfeadbf ]
```

```
    The "minItems" alone does not impose any upper limit of DMAs, so switch
    it to "maxItems" which also implies same value for minItems.

    Fixes: 370f696e4474 ("dt-bindings: serial: snps-dw-apb-uart: add dma & dma-names
properties")
    Signed-off-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Reviewed-by: Conor Dooley <conor.dooley@microchip.com>
    Acked-by: Rob Herring <robh@kernel.org>
    Link: https://lore.kernel.org/r/20230317155712.99654-1-krzysztof.kozlowski@linaro.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 00a1b0360c3dd281fd9b32ab6b7e6f24101f49e3
Author: Randy Dunlap <rdunlap@infradead.org>
Date:   Tue Mar 28 19:15:29 2023 -0700

    linux/vt_buffer.h: allow either builtin or modular for macros

    [ Upstream commit 2b76ffe81e32afd6d318dc4547e2ba8c46207b77 ]

    Fix build errors on ARCH=alpha when CONFIG_MDA_CONSOLE=m.
    This allows the ARCH macros to be the only ones defined.

    In file included from ../drivers/video/console/mdacon.c:37:
    ../arch/alpha/include/asm/vga.h:17:40: error: expected identifier or '(' before
'volatile'
       17 | static inline void scr_writew(u16 val, volatile u16 *addr)
          |                                        ^~~~~~~~
    ../include/linux/vt_buffer.h:24:34: note: in definition of macro 'scr_writew'
       24 | #define scr_writew(val, addr) (*(addr) = (val))
          |                                  ^~~~
    ../include/linux/vt_buffer.h:24:40: error: expected ')' before '=' token
       24 | #define scr_writew(val, addr) (*(addr) = (val))
          |                                        ^
    ../arch/alpha/include/asm/vga.h:17:20: note: in expansion of macro 'scr_writew'
       17 | static inline void scr_writew(u16 val, volatile u16 *addr)
          |                    ^~~~~~~~~~
    ../arch/alpha/include/asm/vga.h:25:29: error: expected identifier or '(' before
'volatile'
       25 | static inline u16 scr_readw(volatile const u16 *addr)
          |                             ^~~~~~~~

    Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
    Signed-off-by: Randy Dunlap <rdunlap@infradead.org>
    Cc: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Cc: Jiri Slaby <jirislaby@kernel.org>
    Cc: dri-devel@lists.freedesktop.org
    Cc: linux-fbdev@vger.kernel.org
    Link: https://lore.kernel.org/r/20230329021529.16188-1-rdunlap@infradead.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4bcf4766d64ee2428b93d9b8a2b9e1e0edea7956
Author: Cristian Ciocaltea <cristian.ciocaltea@collabora.com>
Date:   Tue Mar 28 12:49:01 2023 +0300

    ASoC: es8316: Handle optional IRQ assignment

    [ Upstream commit 39db65a0a17b54915b269d3685f253a4731f344c ]

    The driver is able to work fine without relying on a mandatory interrupt
    being assigned to the I2C device. This is only needed when making use of
    the jack-detect support.

    However, the following warning message is always emitted when there is
```

```
    no such interrupt available:

      es8316 0-0011: Failed to get IRQ 0: -22

    Do not attempt to request an IRQ if it is not available/valid. This also
    ensures the rather misleading message is not displayed anymore.

    Also note the IRQ validation relies on commit dab472eb931bc291 ("i2c /
    ACPI: Use 0 to indicate that device does not have interrupt assigned").

    Fixes: 822257661031 ("ASoC: es8316: Add jack-detect support")
    Signed-off-by: Cristian Ciocaltea <cristian.ciocaltea@collabora.com>
    Reviewed-by: Hans de Goede <hdegoede@redhat.com>
    Link: https://lore.kernel.org/r/20230328094901.50763-1-cristian.ciocaltea@collabora.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b5a119adfa98be9cbd26b0fabc606182310da832
Author: H. Nikolaus Schaller <hns@goldelico.com>
Date:   Thu Mar 9 17:56:31 2023 +0100

    PCI: imx6: Install the fault handler only on compatible match

    [ Upstream commit 5f5ac460dfe7f4e11f99de9870f240e39189cf72 ]

    commit bb38919ec56e ("PCI: imx6: Add support for i.MX6 PCIe controller")
    added a fault hook to this driver in the probe function. So it was only
    installed if needed.

    commit bde4a5a00e76 ("PCI: imx6: Allow probe deferral by reset GPIO")
    moved it from probe to driver init which installs the hook unconditionally
    as soon as the driver is compiled into a kernel.

    When this driver is compiled as a module, the hook is not registered
    until after the driver has been matched with a .compatible and
    loaded.

    commit 415b6185c541 ("PCI: imx6: Fix config read timeout handling")
    extended the fault handling code.

    commit 2d8ed461dbc9 ("PCI: imx6: Add support for i.MX8MQ")
    added some protection for non-ARM architectures, but this does not
    protect non-i.MX ARM architectures.

    Since fault handlers can be triggered on any architecture for different
    reasons, there is no guarantee that they will be triggered only for the
    assumed situation, leading to improper error handling (i.MX6-specific
    imx6q_pcie_abort_handler) on foreign systems.

    I had seen strange L3 imprecise external abort messages several times on
    OMAP4 and OMAP5 devices and couldn't make sense of them until I realized
    they were related to this unused imx6q driver because I had
    CONFIG_PCI_IMX6=y.

    Note that CONFIG_PCI_IMX6=y is useful for kernel binaries that are designed
    to run on different ARM SoC and be differentiated only by device tree
    binaries. So turning off CONFIG_PCI_IMX6 is not a solution.

    Therefore we check the compatible in the init function before registering
    the fault handler.

    Link:
https://lore.kernel.org/r/e1bcfc3078c82b53aa9b78077a89955abe4ea009.1678380991.git.hns@goldel
ico.com
    Fixes: bde4a5a00e76 ("PCI: imx6: Allow probe deferral by reset GPIO")
```

```
    Fixes: 415b6185c541 ("PCI: imx6: Fix config read timeout handling")
    Fixes: 2d8ed461dbc9 ("PCI: imx6: Add support for i.MX8MQ")
    Signed-off-by: H. Nikolaus Schaller <hns@goldelico.com>
    Signed-off-by: Lorenzo Pieralisi <lpieralisi@kernel.org>
    Reviewed-by: Richard Zhu <hongxing.zhu@nxp.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 76ebf77022935f6907f7e3d900c73baec8214584
Author: Daniel Baluta <daniel.baluta@nxp.com>
Date:   Fri Mar 24 14:40:19 2023 +0200

    ASoC: soc-compress: Inherit atomicity from DAI link for Compress FE

    [ Upstream commit 37b58becc1cee4d591024f2056d7ffa99c6089e0 ]

    After commit bbf7d3b1c4f40 ("ASoC: soc-pcm: align BE 'atomicity' with
    that of the FE") BE and FE atomicity must match.

    In the case of Compress PCM there is a mismatch in atomicity between FE
    and BE and we get errors like this:

    [   36.434566]  sai1-wm8960-hifi: dpcm_be_connect: FE is atomic but BE
    is nonatomic, invalid configuration
    [   36.444278]  PCM Deep Buffer: ASoC: can't connect SAI1.OUT

    In order to fix this we must inherit the atomicity from DAI link
    associated with current PCM Compress FE.

    Fixes: bbf7d3b1c4f4 ("ASoC: soc-pcm: align BE 'atomicity' with that of the FE")
    Signed-off-by: Daniel Baluta <daniel.baluta@nxp.com>
    Reviewed-by: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com>
    Link: https://lore.kernel.org/r/20230324124019.30826-1-daniel.baluta@oss.nxp.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 231598b40a070a6bf780c0df1ff5ae3e57102900
Author: Zheng Wang <zyytlz.wz@163.com>
Date:   Mon Mar 20 14:29:31 2023 +0800

    usb: gadget: udc: renesas_usb3: Fix use after free bug in renesas_usb3_remove due to
race condition

    [ Upstream commit 2b947f8769be8b8181dc795fd292d3e7120f5204 ]

    In renesas_usb3_probe, role_work is bound with renesas_usb3_role_work.
    renesas_usb3_start will be called to start the work.

    If we remove the driver which will call usbhs_remove, there may be
    an unfinished work. The possible sequence is as follows:

    CPU0                                    CPU1

                                             renesas_usb3_role_work
    renesas_usb3_remove
    usb_role_switch_unregister
    device_unregister
    kfree(sw)
    //free usb3->role_sw
                                             usb_role_switch_set_role
                                             //use usb3->role_sw

    The usb3->role_sw could be freed under such circumstance and then
    used in usb_role_switch_set_role.

    This bug was found by static analysis. And note that removing a
```

```
    driver is a root-only operation, and should never happen in normal
    case. But the root user may directly remove the device which
    will also trigger the remove function.

    Fix it by canceling the work before cleanup in the renesas_usb3_remove.

    Fixes: 39facfa01c9f ("usb: gadget: udc: renesas_usb3: Add register of usb role switch")
    Signed-off-by: Zheng Wang <zyytlz.wz@163.com>
    Reviewed-by: Yoshihiro Shimoda <yoshihiro.shimoda.uh@renesas.com>
    Link: https://lore.kernel.org/r/20230320062931.505170-1-zyytlz.wz@163.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e3c4de959c9c5d663a12a4d0fd336f757c3739d7
Author: Fabio M. De Francesco <fmdefrancesco@gmail.com>
Date:   Wed Mar 15 13:52:56 2023 +0100

    module/decompress: Never use kunmap() for local un-mappings

    [ Upstream commit 3c17655ab13704582fe25e8ea3200a9b2f8bf20a ]

    Use kunmap_local() to unmap pages locally mapped with kmap_local_page().

    kunmap_local() must be called on the kernel virtual address returned by
    kmap_local_page(), differently from how we use kunmap() which instead
    expects the mapped page as its argument.

    In module_zstd_decompress() we currently map with kmap_local_page() and
    unmap with kunmap(). This breaks the code and so it should be fixed.

    Cc: Piotr Gorski <piotrgorski@cachyos.org>
    Cc: Dmitry Torokhov <dmitry.torokhov@gmail.com>
    Cc: Luis Chamberlain <mcgrof@kernel.org>
    Cc: Stephen Boyd <swboyd@chromium.org>
    Cc: Ira Weiny <ira.weiny@intel.com>
    Fixes: 169a58ad824d ("module/decompress: Support zstd in-kernel decompression")
    Signed-off-by: Fabio M. De Francesco <fmdefrancesco@gmail.com>
    Reviewed-by: Stephen Boyd <swboyd@chromium.org>
    Reviewed-by: Ira Weiny <ira.weiny@intel.com>
    Reviewed-by: Piotr Gorski <piotrgorski@cachyos.org>
    Signed-off-by: Luis Chamberlain <mcgrof@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit bdeb51863f9d18b85afeb75da439105154e1184a
Author: Kunihiko Hayashi <hayashi.kunihiko@socionext.com>
Date:   Wed Mar 22 11:31:01 2023 +0900

    spi: f_ospi: Add missing spi_mem_default_supports_op() helper

    [ Upstream commit bc43c5ec1a97772269785d19f62d32c91ac5fc36 ]

    The .supports_op() callback function returns true by default after
    performing driver-specific checks. Therefore the driver cannot apply
    the buswidth in devicetree.

    Call spi_mem_default_supports_op() helper to handle the buswidth
    in devicetree.

    Fixes: 1b74dd64c861 ("spi: Add Socionext F_OSPI SPI flash controller driver")
    Signed-off-by: Kunihiko Hayashi <hayashi.kunihiko@socionext.com>
    Link: https://lore.kernel.org/r/20230322023101.24490-1-hayashi.kunihiko@socionext.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 47be21a1346cccea2f98567025c1ee1bd2fdb418
```

```
Author: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
Date:   Tue Jan 3 05:11:59 2023 +0200

    interconnect: qcom: osm-l3: drop unuserd header inclusion

    [ Upstream commit f730038fe6a6de170268fd779b2c029aa70a928b ]

    The commit 4529992c9474 ("interconnect: qcom: osm-l3: Use
    platform-independent node ids") made osm-l3 driver use
    platform-independent IDs, removing the need to include platform headers.

    Fixes: 4529992c9474 ("interconnect: qcom: osm-l3: Use platform-independent node ids")
    Signed-off-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Reviewed-by: Bjorn Andersson <andersson@kernel.org>
    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Link: https://lore.kernel.org/r/20230103031159.1060075-1-dmitry.baryshkov@linaro.org
    Signed-off-by: Georgi Djakov <djakov@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 548e2d296c5b12c690d26988c0bb84ece0d86cbc
Author: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
Date:   Tue Jan 3 06:57:17 2023 +0200

    interconnect: qcom: drop obsolete OSM_L3/EPSS defines

    [ Upstream commit 4c4161b4c81b632fc29002c324e996cbb79894e7 ]

    Since Qualcomm platforms have switched to the separate OSM_L3/EPSS
    driver, old related defines became unused. Drop them now.

    Suggested-by: Bjorn Andersson <andersson@kernel.org>
    Fixes: 4529992c9474 ("interconnect: qcom: osm-l3: Use platform-independent node ids")
    Signed-off-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Link: https://lore.kernel.org/r/20230103045717.1079067-1-dmitry.baryshkov@linaro.org
    Signed-off-by: Georgi Djakov <djakov@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b64cb3f085fed296103c91f0db6acad30a021b36
Author: Uwe Kleine-König <u.kleine-koenig@pengutronix.de>
Date:   Mon Mar 6 07:57:32 2023 +0100

    spi: imx: Don't skip cleanup in remove's error path

    [ Upstream commit 11951c9e3f364d7ae3b568a0e52c8335d43066b5 ]

    Returning early in a platform driver's remove callback is wrong. In this
    case the dma resources are not released in the error path. this is never
    retried later and so this is a permanent leak. To fix this, only skip
    hardware disabling if waking the device fails.

    Fixes: d593574aff0a ("spi: imx: do not access registers while clocks disabled")
    Signed-off-by: Uwe Kleine-König <u.kleine-koenig@pengutronix.de>
    Link: https://lore.kernel.org/r/20230306065733.2170662-2-u.kleine-koenig@pengutronix.de
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 77806d7c9bebe40a8cdce2b8d30fbe6511745df8
Author: Uwe Kleine-König <u.kleine-koenig@pengutronix.de>
Date:   Fri Mar 17 09:42:31 2023 +0100

    spi: atmel-quadspi: Free resources even if runtime resume failed in .remove()

    [ Upstream commit 9448bc1dee65f86c0fe64d9dea8b410af0586886 ]
```

```
    An early error exit in atmel_qspi_remove() doesn't prevent the device
    unbind. So this results in an spi controller with an unbound parent
    and unmapped register space (because devm_ioremap_resource() is undone).
    So using the remaining spi controller probably results in an oops.

    Instead unregister the controller unconditionally and only skip hardware
    access and clk disable.

    Also add a warning about resume failing and return zero unconditionally.
    The latter has the only effect to suppress a less helpful error message by
    the spi core.

    Fixes: 4a2f83b7f780 ("spi: atmel-quadspi: add runtime pm support")
    Signed-off-by: Uwe Kleine-König <u.kleine-koenig@pengutronix.de>
    Reviewed-by: Tudor Ambarus <tudor.ambarus@linaro.org>
    Link: https://lore.kernel.org/r/20230317084232.142257-3-u.kleine-koenig@pengutronix.de
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c81e7dd6a1ac54154b3a97838a920669bd6f0d96
Author: Uwe Kleine-König <u.kleine-koenig@pengutronix.de>
Date:   Fri Mar 17 09:42:30 2023 +0100

    spi: atmel-quadspi: Don't leak clk enable count in pm resume

    [ Upstream commit c18bbac353ffed50be134b0a2a059a2bd540c503 ]

    The pm resume call is supposed to enable two clocks. If the second enable
    fails the callback reports failure but doesn't undo the first enable.

    So call clk_disable() for the first clock when clk_enable() for the second
    one fails.

    Fixes: 4a2f83b7f780 ("spi: atmel-quadspi: add runtime pm support")
    Signed-off-by: Uwe Kleine-König <u.kleine-koenig@pengutronix.de>
    Reviewed-by: Tudor Ambarus <tudor.ambarus@linaro.org>
    Link: https://lore.kernel.org/r/20230317084232.142257-2-u.kleine-koenig@pengutronix.de
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 149b5f036faeecd9a23fa5d7c709e66958468dc1
Author: Doug Berger <opendmb@gmail.com>
Date:   Thu Mar 9 11:02:24 2023 -0800

    serial: 8250_bcm7271: Fix arbitration handling

    [ Upstream commit 15ac1122fd6d4bf408a03e6f23c7ad4f60b22f9e ]

    The arbitration of the UART DMA is mishandled for a few
    exceptional cases when probing and releasing the driver.

    It is possible that the DMA register spaces are not defined in
    device tree for an instance of the driver, so attempts to access
    the registers in brcmuart_arbitration() would use NULL pointers.

    It is also possible for the probe function to return an error
    while still holding the UART DMA. This would prevent the UART
    DMA from being claimed by an instance that could use it.

    These errors are addressed by only releasing the UART DMA if it
    is held by this instance (i.e. priv->dma_enabled == 1) and
    directing early error paths in probe to this common release_dma
    handling.

    Fixes: 41a469482de2 ("serial: 8250: Add new 8250-core based Broadcom STB driver")
```

```
    Signed-off-by: Doug Berger <opendmb@gmail.com>
    Acked-by: Florian Fainelli <f.fainelli@gmail.com>
    Link: https://lore.kernel.org/r/20230309190224.687380-1-opendmb@gmail.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 557e54e5981baf7f4a1b0fb0b6aabb8baf3a87e0
Author: Geert Uytterhoeven <geert+renesas@glider.be>
Date:   Mon Mar 13 11:58:37 2023 +0100

    spi: Constify spi parameters of chip select APIs

    [ Upstream commit d2f19eec510424caa55ea949f016ddabe2d8173a ]

    The "spi" parameters of spi_get_chipselect() and spi_get_csgpiod() can
    be const.

    Fixes: 303feb3cc06ac066 ("spi: Add APIs in spi core to set/get spi->chip_select and spi-
>cs_gpiod")
    Signed-off-by: Geert Uytterhoeven <geert+renesas@glider.be>
    Link:
https://lore.kernel.org/r/b112de79e7a1e9095a3b6ff22b639f39e39d7748.1678704562.git.geert+rene
sas@glider.be
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 418515dc45f55f7f6e1b27ea3a96b37dd1200e4c
Author: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
Date:   Sun Mar 12 16:34:28 2023 +0100

    iio: light: max44009: add missing OF device matching

    [ Upstream commit b29c49026c3c05a11f845dba17cad0b3ba06836d ]

    The driver currently matches only via i2c_device_id, but also has
    of_device_id table:

      drivers/iio/light/max44009.c:545:34: error: ‘max44009_of_match’ defined but not used
[-Werror=unused-const-variable=]

    Fixes: 6aef699a7d7e ("iio: light: add driver for MAX44009")
    Signed-off-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Link: https://lore.kernel.org/r/20230312153429.371702-2-krzysztof.kozlowski@linaro.org
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3fb3505636d033bbf7a0851dac63d01732c51d62
Author: Jason Gunthorpe <jgg@ziepe.ca>
Date:   Mon Mar 6 13:27:04 2023 -0400

    iommufd/selftest: Catch overflow of uptr and length

    [ Upstream commit fd8c1a4aee973e87d890a5861e106625a33b2c4e ]

    syzkaller hits a WARN_ON when trying to have a uptr close to UINTPTR_MAX:

      WARNING: CPU: 1 PID: 393 at drivers/iommu/iommufd/selftest.c:403
iommufd_test+0xb19/0x16f0
      Modules linked in:
      CPU: 1 PID: 393 Comm: repro Not tainted 6.2.0-c9c3395d5e3d #1
      Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS rel-1.16.0-0-
gd239552ce722-prebuilt.qemu.org 04/01/2014
      RIP: 0010:iommufd_test+0xb19/0x16f0
      Code: 94 c4 31 ff 44 89 e6 e8 a5 54 17 ff 45 84 e4 0f 85 bb 0b 00 00 41 be fb ff ff ff
e8 31 53 17 ff e9 a0 f7 ff ff e8 27 53 17 ff <0f> 0b 41 be 8
```

```
       RSP: 0018:ffffc90000eabdc0 EFLAGS: 00010246
       RAX: 0000000000000000 RBX: 0000000000000000 RCX: ffffffff8214c487
       RDX: 0000000000000000 RSI: ffff88800f5c8000 RDI: 0000000000000002
       RBP: ffffc90000eabe48 R08: 0000000000000000 R09: 0000000000000001
       R10: 0000000000000001 R11: 0000000000000000 R12: 00000000cd2b0000
       R13: 00000000cd2af000 R14: 0000000000000000 R15: ffffc90000eabe68
       FS:  00007f94d76d5740(0000) GS:ffff88807dd00000(0000) knlGS:0000000000000000
       CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
       CR2: 0000000020000043 CR3: 0000000006880006 CR4: 0000000000770ee0
       PKRU: 55555554
       Call Trace:
        <TASK>
        ? write_comp_data+0x2f/0x90
        iommufd_fops_ioctl+0x1ef/0x310
        __x64_sys_ioctl+0x10e/0x160
        ? __pfx_iommufd_fops_ioctl+0x10/0x10
        do_syscall_64+0x3b/0x90
        entry_SYSCALL_64_after_hwframe+0x72/0xdc

    Check that the user memory range doesn't overflow.

    Fixes: f4b20bb34c83 ("iommufd: Add kernel support for testing iommufd")
    Link: https://lore.kernel.org/r/0-v1-95390ed1df8d+8f-
iommufd_mock_overflow_jgg@nvidia.com
    Reviewed-by: Kevin Tian <kevin.tian@intel.com>
    Reported-by: Pengfei Xu <pengfei.xu@intel.com>
    Link: https://lore.kernel.org/r/Y/hOiilV1wJvu/Hv@xpf.sh.intel.com
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b3f9407f1b8ed7a58e67c918fce06f81dc2d99cd
Author: Marco Pagani <marpagan@redhat.com>
Date:   Wed Mar 1 15:03:08 2023 +0100

    fpga: bridge: fix kernel-doc parameter description

    [ Upstream commit 7ef1a2c1c9dffa177ecc3ea50b7f5ee63a621137 ]

    Fix the kernel-doc description for the "struct fpga_image_info *info"
    parameter of the fpga_bridge_get() function.

    Fixes: 060ac5c8fa7b ("fpga: bridge: kernel-doc fixes")
    Signed-off-by: Marco Pagani <marpagan@redhat.com>
    Reviewed-by: Tom Rix <trix@redhat.com>
    Acked-by: Xu Yilun <yilun.xu@intel.com>
    Link: https://lore.kernel.org/r/20230301140309.512578-1-marpagan@redhat.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit bf1c5f56449b096f7093fbb1c15ee100b4e1f416
Author: Marek Vasut <marex@denx.de>
Date:   Thu Feb 23 05:22:52 2023 +0100

    serial: stm32: Re-assert RTS/DE GPIO in RS485 mode only if more data are transmitted

    [ Upstream commit c47527cbcc3c50800f34b8c684f29721f75de246 ]

    The stm32_usart_transmit_chars() may be called with empty or stopped
    transmit queue, and no XON/OFF character pending. This can happen at
    the end of transmission, where this last call is used to either handle
    the XON/XOFF x_char, or disable TX interrupt if queue is empty or
    stopped.

    If that occurs, do not assert the RS485 RTS/DE GPIO anymore, as the
    GPIO would remain asserted past the end of transmission and that would
```

```
    block the RS485 bus after the transmission.

    Only assert the RS485 RTS/DE GPIO if there is either pending XON/XOFF
    x_char, or at least one character in running transmit queue.

    Fixes: d7c76716169d ("serial: stm32: Use TC interrupt to deassert GPIO RTS in RS485
mode")
    Signed-off-by: Marek Vasut <marex@denx.de>
    Link: https://lore.kernel.org/r/20230223042252.95480-2-marex@denx.de
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 392e6105a5ee73ee5b232f5548fa0376005d83cb
Author: Prashanth K <quic_prashk@quicinc.com>
Date:   Fri Feb 24 11:16:58 2023 +0530

    usb: dwc3: gadget: Change condition for processing suspend event

    [ Upstream commit 4decf4060ecfee1f7a710999fcd421645ac0c419 ]

    Currently we process the suspend interrupt event only if the
    device is in configured state. Consider a case where device
    is not configured and got suspend interrupt, in that case our
    gadget will still use 100mA as composite_suspend didn't happen.
    But battery charging specification (BC1.2) expects a downstream
    device to draw less than 2.5mA when unconnected OR suspended.

    Fix this by removing the condition for processing suspend event,
    and thus composite_resume would set vbus draw to 2.

    Fixes: 72704f876f50 ("dwc3: gadget: Implement the suspend entry event handler")
    Signed-off-by: Prashanth K <quic_prashk@quicinc.com>
    Acked-by: Thinh Nguyen <Thinh.Nguyen@synopsys.com>
    Link: https://lore.kernel.org/r/1677217619-10261-2-git-send-email-
quic_prashk@quicinc.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1142e141054044233344a40f92a49344aa2461e3
Author: Wolfram Sang <wsa+renesas@sang-engineering.com>
Date:   Tue Mar 7 17:30:37 2023 +0100

    usb: host: xhci-rcar: remove leftover quirk handling

    [ Upstream commit 5d67f4861884762ebc2bddb5d667444e45f25782 ]

    Loading V3 firmware does not need a quirk anymore, remove the leftover
    code.

    Fixes: ed8603e11124 ("usb: host: xhci-rcar: Simplify getting the firmware name for R-Car
Gen3")
    Reviewed-by: Geert Uytterhoeven <geert+renesas@glider.be>
    Signed-off-by: Wolfram Sang <wsa+renesas@sang-engineering.com>
    Link: https://lore.kernel.org/r/20230307163041.3815-10-wsa+renesas@sang-engineering.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8c935a960bca431447624d3a981f2c7138140af3
Author: John Stultz <jstultz@google.com>
Date:   Wed Mar 8 20:40:43 2023 +0000

    pstore: Revert pmsg_lock back to a normal mutex

    [ Upstream commit 5239a89b06d6b199f133bf0ffea421683187f257 ]
```

```
    This reverts commit 76d62f24db07f22ccf9bc18ca793c27d4ebef721.

    So while priority inversion on the pmsg_lock is an occasional
    problem that an rt_mutex would help with, in uses where logging
    is writing to pmsg heavily from multiple threads, the pmsg_lock
    can be heavily contended.

    After this change landed, it was reported that cases where the
    mutex locking overhead was commonly adding on the order of 10s
    of usecs delay had suddenly jumped to ~msec delay with rtmutex.

    It seems the slight differences in the locks under this level
    of contention causes the normal mutexes to utilize the spinning
    optimizations, while the rtmutexes end up in the sleeping
    slowpath (which allows additional threads to pile on trying
    to take the lock).

    In this case, it devolves to a worse case senerio where the lock
    acquisition and scheduling overhead dominates, and each thread
    is waiting on the order of ~ms to do ~us of work.

    Obviously, having tons of threads all contending on a single
    lock for logging is non-optimal, so the proper fix is probably
    reworking pstore pmsg to have per-cpu buffers so we don't have
    contention.

    Additionally, Steven Rostedt has provided some furhter
    optimizations for rtmutexes that improves the rtmutex spinning
    path, but at least in my testing, I still see the test tripping
    into the sleeping path on rtmutexes while utilizing the spinning
    path with mutexes.

    But in the short term, lets revert the change to the rt_mutex
    and go back to normal mutexes to avoid a potentially major
    performance regression. And we can work on optimizations to both
    rtmutexes and finer-grained locking for pstore pmsg in the
    future.

    Cc: Wei Wang <wvw@google.com>
    Cc: Midas Chien<midaschieh@google.com>
    Cc: "Chunhui Li (李春辉)" <chunhui.li@mediatek.com>
    Cc: Steven Rostedt <rostedt@goodmis.org>
    Cc: Kees Cook <keescook@chromium.org>
    Cc: Anton Vorontsov <anton@enomsg.org>
    Cc: "Guilherme G. Piccoli" <gpiccoli@igalia.com>
    Cc: Tony Luck <tony.luck@intel.com>
    Cc: kernel-team@android.com
    Fixes: 76d62f24db07 ("pstore: Switch pmsg_lock to an rt_mutex to avoid priority
inversion")
    Reported-by: "Chunhui Li (李春辉)" <chunhui.li@mediatek.com>
    Signed-off-by: John Stultz <jstultz@google.com>
    Signed-off-by: Kees Cook <keescook@chromium.org>
    Link: https://lore.kernel.org/r/20230308204043.2061631-1-jstultz@google.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c9d6ae1e6821524cda48d116c02bc4849d8da648
Author: Hans de Goede <hdegoede@redhat.com>
Date:   Tue Feb 21 15:53:24 2023 +0100

    drivers: staging: rtl8723bs: Fix locking in rtw_scan_timeout_handler()

    [ Upstream commit 3f467036093fedd7e231924327455fc609b5ef02 ]

    Commit cc7ad0d77b51 ("drivers: staging: rtl8723bs: Fix deadlock in
    rtw_surveydone_event_callback()") besides fixing the deadlock also
```

```
    modified rtw_scan_timeout_handler() to use spin_[un]lock_irq()
    instead of spin_[un]lock_bh().

    Disabling the IRQs is not necessary since all code taking this lock
    runs from either user contexts or from softirqs

    rtw_scan_timeout_handler() is the only function taking pmlmepriv->lock
    which uses spin_[un]lock_irq() for this. Switch back to
    spin_[un]lock_bh() to make it consistent with the rest of the code.

    Fixes: cc7ad0d77b51 ("drivers: staging: rtl8723bs: Fix deadlock in
rtw_surveydone_event_callback()")
    Cc: Duoming Zhou <duoming@zju.edu.cn>
    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Link: https://lore.kernel.org/r/20230221145326.7808-2-hdegoede@redhat.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4ab1bace1dd3875371b481ef4301c4671bddea22
Author: Hans de Goede <hdegoede@redhat.com>
Date:   Tue Feb 21 15:53:23 2023 +0100

    drivers: staging: rtl8723bs: Fix locking in _rtw_join_timeout_handler()

    [ Upstream commit 215792eda008f6a1e7ed9d77fa20d582d22bb114 ]

    Commit 041879b12ddb ("drivers: staging: rtl8192bs: Fix deadlock in
    rtw_joinbss_event_prehandle()") besides fixing the deadlock also
    modified _rtw_join_timeout_handler() to use spin_[un]lock_irq()
    instead of spin_[un]lock_bh().

    _rtw_join_timeout_handler() calls rtw_do_join() which takes
    pmlmepriv->scanned_queue.lock using spin_[un]lock_bh(). This
    spin_unlock_bh() call re-enables softirqs which triggers an oops in
    kernel/softirq.c: __local_bh_enable_ip() when it calls
    lockdep_assert_irqs_enabled():

    [  244.506087] WARNING: CPU: 2 PID: 0 at kernel/softirq.c:376
__local_bh_enable_ip+0xa6/0x100
    ...
    [  244.509022] Call Trace:
    [  244.509048]  <IRQ>
    [  244.509100]  _rtw_join_timeout_handler+0x134/0x170 [r8723bs]
    [  244.509468]  ? __pfx__rtw_join_timeout_handler+0x10/0x10 [r8723bs]
    [  244.509772]  ? __pfx__rtw_join_timeout_handler+0x10/0x10 [r8723bs]
    [  244.510076]  call_timer_fn+0x95/0x2a0
    [  244.510200]  __run_timers.part.0+0x1da/0x2d0

    This oops is causd by the switch to spin_[un]lock_irq() which disables
    the IRQs for the entire duration of _rtw_join_timeout_handler().

    Disabling the IRQs is not necessary since all code taking this lock
    runs from either user contexts or from softirqs, switch back to
    spin_[un]lock_bh() to fix this.

    Fixes: 041879b12ddb ("drivers: staging: rtl8192bs: Fix deadlock in
rtw_joinbss_event_prehandle()")
    Cc: Duoming Zhou <duoming@zju.edu.cn>
    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Link: https://lore.kernel.org/r/20230221145326.7808-1-hdegoede@redhat.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 54fa9b1678d4a79f5c5f51a3d720db29341acc23
Author: Lucas Tanure <lucas.tanure@collabora.com>
```

```
Date:   Thu Feb 23 08:43:21 2023 +0000

    ASoC: cs35l41: Only disable internal boost

    [ Upstream commit 4658de99d43cd740e019e7fd124b4128f8f4027f ]

    In error situations, only the internal boost case should be disabled and
    re-enabled.
    Also, for other boost cases re-enabling the boost to the default internal
    boost config is incorrect.

    Fixes: 6450ef559056 ("ASoC: cs35l41: CS35L41 Boosted Smart Amplifier")
    Signed-off-by: Lucas Tanure <lucas.tanure@collabora.com>
    Acked-by: Charles Keepax <ckeepax@opensource.cirrus.com>
    Reviewed-by: David Rhodes <david.rhodes@cirrus.com>
    Link: https://lore.kernel.org/r/20230223084324.9076-2-lucas.tanure@collabora.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ee231701cfc9fc3c3084c52cb011554fc4b3d4b5
Author: Randy Dunlap <rdunlap@infradead.org>
Date:   Sat Feb 25 21:39:46 2023 -0800

    ipmi: ASPEED_BT_IPMI_BMC: select REGMAP_MMIO instead of depending on it

    [ Upstream commit 2a587b9ad052e7e92e508aea90c1e2ae433c1908 ]

    REGMAP is a hidden (not user visible) symbol. Users cannot set it
    directly thru "make *config", so drivers should select it instead of
    depending on it if they need it.

    Consistently using "select" or "depends on" can also help reduce
    Kconfig circular dependency issues.

    Therefore, change the use of "depends on REGMAP_MMIO" to
    "select REGMAP_MMIO", which will also set REGMAP.

    Fixes: eb994594bc22 ("ipmi: bt-bmc: Use a regmap for register access")
    Signed-off-by: Randy Dunlap <rdunlap@infradead.org>
    Cc: Andrew Jeffery <andrew@aj.id.au>
    Cc: Corey Minyard <minyard@acm.org>
    Cc: openipmi-developer@lists.sourceforge.net
    Cc: Arnd Bergmann <arnd@arndb.de>
    Cc: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Message-Id: <20230226053953.4681-2-rdunlap@infradead.org>
    Signed-off-by: Corey Minyard <cminyard@mvista.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 426384dd4980040651536fef5feac4dcc4d7ee4e
Author: Kuniyuki Iwashima <kuniyu@amazon.com>
Date:   Mon Apr 24 15:20:22 2023 -0700

    tcp/udp: Fix memleaks of sk and zerocopy skbs with TX timestamp.

    [ Upstream commit 50749f2dd6854a41830996ad302aef2ffaf011d8 ]

    syzkaller reported [0] memory leaks of an UDP socket and ZEROCOPY
    skbs.  We can reproduce the problem with these sequences:

      sk = socket(AF_INET, SOCK_DGRAM, 0)
      sk.setsockopt(SOL_SOCKET, SO_TIMESTAMPING, SOF_TIMESTAMPING_TX_SOFTWARE)
      sk.setsockopt(SOL_SOCKET, SO_ZEROCOPY, 1)
      sk.sendto(b'', MSG_ZEROCOPY, ('127.0.0.1', 53))
      sk.close()
```

```
    sendmsg() calls msg_zerocopy_alloc(), which allocates a skb, sets
    skb->cb->ubuf.refcnt to 1, and calls sock_hold().  Here, struct
    ubuf_info_msgzc indirectly holds a refcnt of the socket.  When the
    skb is sent, __skb_tstamp_tx() clones it and puts the clone into
    the socket's error queue with the TX timestamp.

    When the original skb is received locally, skb_copy_ubufs() calls
    skb_unclone(), and pskb_expand_head() increments skb->cb->ubuf.refcnt.
    This additional count is decremented while freeing the skb, but struct
    ubuf_info_msgzc still has a refcnt, so __msg_zerocopy_callback() is
    not called.

    The last refcnt is not released unless we retrieve the TX timestamped
    skb by recvmsg().  Since we clear the error queue in inet_sock_destruct()
    after the socket's refcnt reaches 0, there is a circular dependency.
    If we close() the socket holding such skbs, we never call sock_put()
    and leak the count, sk, and skb.

    TCP has the same problem, and commit e0c8bccd40fc ("net: stream:
    purge sk_error_queue in sk_stream_kill_queues()") tried to fix it
    by calling skb_queue_purge() during close().  However, there is a
    small chance that skb queued in a qdisc or device could be put
    into the error queue after the skb_queue_purge() call.

    In __skb_tstamp_tx(), the cloned skb should not have a reference
    to the ubuf to remove the circular dependency, but skb_clone() does
    not call skb_copy_ubufs() for zerocopy skb.  So, we need to call
    skb_orphan_frags_rx() for the cloned skb to call skb_copy_ubufs().

    [0]:
    BUG: memory leak
    unreferenced object 0xffff88800c6d2d00 (size 1152):
      comm "syz-executor392", pid 264, jiffies 4294785440 (age 13.044s)
      hex dump (first 32 bytes):
        00 00 00 00 00 00 00 00 cd af e8 81 00 00 00 00  ................
        02 00 07 40 00 00 00 00 00 00 00 00 00 00 00 00  ...@............
      backtrace:
        [<0000000055636812>] sk_prot_alloc+0x64/0x2a0 net/core/sock.c:2024
        [<0000000054d77b7a>] sk_alloc+0x3b/0x800 net/core/sock.c:2083
        [<0000000066f3c7e0>] inet_create net/ipv4/af_inet.c:319 [inline]
        [<0000000066f3c7e0>] inet_create+0x31e/0xe40 net/ipv4/af_inet.c:245
        [<000000009b83af97>] __sock_create+0x2ab/0x550 net/socket.c:1515
        [<00000000b9b11231>] sock_create net/socket.c:1566 [inline]
        [<00000000b9b11231>] __sys_socket_create net/socket.c:1603 [inline]
        [<00000000b9b11231>] __sys_socket_create net/socket.c:1588 [inline]
        [<00000000b9b11231>] __sys_socket+0x138/0x250 net/socket.c:1636
        [<000000004fb45142>] __do_sys_socket net/socket.c:1649 [inline]
        [<000000004fb45142>] __se_sys_socket net/socket.c:1647 [inline]
        [<000000004fb45142>] __x64_sys_socket+0x73/0xb0 net/socket.c:1647
        [<0000000066999e0e>] do_syscall_x64 arch/x86/entry/common.c:50 [inline]
        [<0000000066999e0e>] do_syscall_64+0x38/0x90 arch/x86/entry/common.c:80
        [<0000000017f238c1>] entry_SYSCALL_64_after_hwframe+0x63/0xcd

    BUG: memory leak
    unreferenced object 0xffff888017633a00 (size 240):
      comm "syz-executor392", pid 264, jiffies 4294785440 (age 13.044s)
      hex dump (first 32 bytes):
        00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
        00 00 00 00 00 00 00 00 00 00 2d 6d 0c 80 88 ff ff  ..........-m.....
      backtrace:
        [<000000002b1c4368>] __alloc_skb+0x229/0x320 net/core/skbuff.c:497
        [<00000000143579a6>] alloc_skb include/linux/skbuff.h:1265 [inline]
        [<00000000143579a6>] sock_omalloc+0xaa/0x190 net/core/sock.c:2596
        [<00000000be626478>] msg_zerocopy_alloc net/core/skbuff.c:1294 [inline]
        [<00000000be626478>] msg_zerocopy_realloc+0x1ce/0x7f0 net/core/skbuff.c:1370
```

```
          [<00000000cbfc9870>] __ip_append_data+0x2adf/0x3b30 net/ipv4/ip_output.c:1037
          [<0000000089869146>] ip_make_skb+0x26c/0x2e0 net/ipv4/ip_output.c:1652
          [<00000000098015c2>] udp_sendmsg+0x1bac/0x2390 net/ipv4/udp.c:1253
          [<0000000045e0e95e>] inet_sendmsg+0x10a/0x150 net/ipv4/af_inet.c:819
          [<000000008d31bfde>] sock_sendmsg_nosec net/socket.c:714 [inline]
          [<000000008d31bfde>] sock_sendmsg+0x141/0x190 net/socket.c:734
          [<0000000021e21aa4>] __sys_sendto+0x243/0x360 net/socket.c:2117
          [<00000000ac0af00c>] __do_sys_sendto net/socket.c:2129 [inline]
          [<00000000ac0af00c>] __se_sys_sendto net/socket.c:2125 [inline]
          [<00000000ac0af00c>] __x64_sys_sendto+0xe1/0x1c0 net/socket.c:2125
          [<0000000066999e0e>] do_syscall_x64 arch/x86/entry/common.c:50 [inline]
          [<0000000066999e0e>] do_syscall_64+0x38/0x90 arch/x86/entry/common.c:80
          [<0000000017f238c1>] entry_SYSCALL_64_after_hwframe+0x63/0xcd

    Fixes: f214f915e7db ("tcp: enable MSG_ZEROCOPY")
    Fixes: b5947e5d1e71 ("udp: msg_zerocopy")
    Reported-by: syzbot <syzkaller@googlegroups.com>
    Signed-off-by: Kuniyuki Iwashima <kuniyu@amazon.com>
    Reviewed-by: Willem de Bruijn <willemb@google.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 201046eff77bc8d97ebc08507313eba38b6ca59d
Author: Gencen Gan <gangecen@hust.edu.cn>
Date:   Mon Apr 24 23:28:01 2023 +0800

    net: amd: Fix link leak when verifying config failed

    [ Upstream commit d325c34d9e7e38d371c0a299d415e9b07f66a1fb ]

    After failing to verify configuration, it returns directly without
    releasing link, which may cause memory leak.

    Paolo Abeni thinks that the whole code of this driver is quite
    "suboptimal" and looks unmainatained since at least ~15y, so he
    suggests that we could simply remove the whole driver, please
    take it into consideration.

    Simon Horman suggests that the fix label should be set to
    "Linux-2.6.12-rc2" considering that the problem has existed
    since the driver was introduced and the commit above doesn't
    seem to exist in net/net-next.

    Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
    Signed-off-by: Gan Gecen <gangecen@hust.edu.cn>
    Reviewed-by: Dongliang Mu <dzm91@hust.edu.cn>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 09edb81948b279e3a153b3ce903350b8225aec45
Author: Kuniyuki Iwashima <kuniyu@amazon.com>
Date:   Fri Apr 21 11:52:55 2023 -0700

    netlink: Use copy_to_user() for optval in netlink_getsockopt().

    [ Upstream commit d913d32cc2707e9cd24fe6fa6d7d470e9c728980 ]

    Brad Spencer provided a detailed report [0] that when calling getsockopt()
    for AF_NETLINK, some SOL_NETLINK options set only 1 byte even though such
    options require at least sizeof(int) as length.

    The options return a flag value that fits into 1 byte, but such behaviour
    confuses users who do not initialise the variable before calling
    getsockopt() and do not strictly check the returned value as char.
```

```
    Currently, netlink_getsockopt() uses put_user() to copy data to optlen and
    optval, but put_user() casts the data based on the pointer, char *optval.
    As a result, only 1 byte is set to optval.

    To avoid this behaviour, we need to use copy_to_user() or cast optval for
    put_user().

    Note that this changes the behaviour on big-endian systems, but we document
    that the size of optval is int in the man page.

      $ man 7 netlink
      ...
      Socket options
           To set or get a netlink socket option, call getsockopt(2) to read
           or setsockopt(2) to write the option with the option level argument
           set to SOL_NETLINK.  Unless otherwise noted, optval is a pointer to
           an int.

    Fixes: 9a4595bc7e67 ("[NETLINK]: Add set/getsockopt options to support more than 32
groups")
    Fixes: be0c22a46cfb ("netlink: add NETLINK_BROADCAST_ERROR socket option")
    Fixes: 38938bfe3489 ("netlink: add NETLINK_NO_ENOBUFS socket flag")
    Fixes: 0a6a3a23ea6e ("netlink: add NETLINK_CAP_ACK socket option")
    Fixes: 2d4bc93368f5 ("netlink: extended ACK reporting")
    Fixes: 89d35528d17d ("netlink: Add new socket option to enable strict checking on
dumps")
    Reported-by: Brad Spencer <bspencer@blackberry.com>
    Link: https://lore.kernel.org/netdev/ZD7VkNWFfp22kTDt@datsun.rim.net/
    Signed-off-by: Kuniyuki Iwashima <kuniyu@amazon.com>
    Reviewed-by: Johannes Berg <johannes@sipsolutions.net>
    Link: https://lore.kernel.org/r/20230421185255.94606-1-kuniyu@amazon.com
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d8d7ce037d9a8f1f0714ece268c4c2c50845bbc3
Author: Liu Jian <liujian56@huawei.com>
Date:   Fri Apr 14 18:30:06 2023 +0800

    Revert "Bluetooth: btsdio: fix use after free bug in btsdio_remove due to unfinished
work"

    [ Upstream commit db2bf510bd5d57f064d9e1db395ed86a08320c54 ]

    This reverts commit 1e9ac114c4428fdb7ff4635b45d4f46017e8916f.

    This patch introduces a possible null-ptr-def problem. Revert it. And the
    fixed bug by this patch have resolved by commit 73f7b171b7c0 ("Bluetooth:
    btsdio: fix use after free bug in btsdio_remove due to race condition").

    Fixes: 1e9ac114c442 ("Bluetooth: btsdio: fix use after free bug in btsdio_remove due to
unfinished work")
    Signed-off-by: Liu Jian <liujian56@huawei.com>
    Signed-off-by: Luiz Augusto von Dentz <luiz.von.dentz@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 12972ae63ae7803de00e39e73b5111e43ea78a40
Author: Marc Dionne <marc.dionne@auristor.com>
Date:   Fri Apr 21 17:16:17 2023 +0100

    rxrpc: Fix error when reading rxrpc tokens

    [ Upstream commit fadfc57cc8047080a123b16f288b7138524fb1e2 ]

    When converting from ASSERTCMP to WARN_ON, the tested condition must
    be inverted, which was missed for this case.
```

```
    This would cause an EIO error when trying to read an rxrpc token, for
    instance when trying to display tokens with AuriStor's "tokens" command.

    Fixes: 84924aac08a4 ("rxrpc: Fix checker warning")
    Signed-off-by: Marc Dionne <marc.dionne@auristor.com>
    Signed-off-by: David Howells <dhowells@redhat.com>
    cc: "David S. Miller" <davem@davemloft.net>
    cc: Eric Dumazet <edumazet@google.com>
    cc: Jakub Kicinski <kuba@kernel.org>
    cc: Paolo Abeni <pabeni@redhat.com>
    cc: linux-afs@lists.infradead.org
    cc: netdev@vger.kernel.org
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9e3c96aed8fe32907e0a4bca05aad457629a820c
Author: Ziyang Xuan <william.xuanziyang@huawei.com>
Date:   Thu Apr 20 20:40:35 2023 +0800

    ipv4: Fix potential uninit variable access bug in __ip_make_skb()

    [ Upstream commit 99e5acae193e369b71217efe6f1dad42f3f18815 ]

    Like commit ea30388baebc ("ipv6: Fix an uninit variable access bug in
    __ip6_make_skb()"). icmphdr does not in skb linear region under the
    scenario of SOCK_RAW socket. Access icmp_hdr(skb)->type directly will
    trigger the uninit variable access bug.

    Use a local variable icmp_type to carry the correct value in different
    scenarios.

    Fixes: 96793b482540 ("[IPV4]: Add ICMPMsgStats MIB (RFC 4293)")
    Reviewed-by: Willem de Bruijn <willemb@google.com>
    Signed-off-by: Ziyang Xuan <william.xuanziyang@huawei.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8d68de656ad2c55ca9befde665756e080dab4920
Author: Ivan Vecera <ivecera@redhat.com>
Date:   Thu Apr 20 20:36:33 2023 +0200

    net/sched: cls_api: Initialize miss_cookie_node when action miss is not used

    [ Upstream commit 2cc8a008d62f3c04eeb7ec6fe59e542802bb8df3 ]

    Function tcf_exts_init_ex() sets exts->miss_cookie_node ptr only
    when use_action_miss is true so it assumes in other case that
    the field is set to NULL by the caller. If not then the field
    contains garbage and subsequent tcf_exts_destroy() call results
    in a crash.
    Ensure that the field .miss_cookie_node pointer is NULL when
    use_action_miss parameter is false to avoid this potential scenario.

    Fixes: 80cd22c35c90 ("net/sched: cls_api: Support hardware miss to tc action")
    Signed-off-by: Ivan Vecera <ivecera@redhat.com>
    Reviewed-by: Pedro Tammela <pctammela@mojatatu.com>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Link: https://lore.kernel.org/r/20230420183634.1139391-1-ivecera@redhat.com
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2322462d6f9ad4874f4e3c63df3b5cc00cb1acbd
Author: Davide Caratti <dcaratti@redhat.com>
Date:   Thu Apr 20 16:59:46 2023 +0200
```

```
    net/sched: sch_fq: fix integer overflow of "credit"

    [ Upstream commit 7041101ff6c3073fd8f2e99920f535b111c929cb ]

    if sch_fq is configured with "initial quantum" having values greater than
    INT_MAX, the first assignment of "credit" does signed integer overflow to
    a very negative value.
    In this situation, the syzkaller script provided by Cristoph triggers the
    CPU soft-lockup warning even with few sockets. It's not an infinite loop,
    but "credit" wasn't probably meant to be minus 2Gb for each new flow.
    Capping "initial quantum" to INT_MAX proved to fix the issue.

    v2: validation of "initial quantum" is done in fq_policy, instead of open
        coding in fq_change() _ suggested by Jakub Kicinski

    Reported-by: Christoph Paasch <cpaasch@apple.com>
    Link: https://github.com/multipath-tcp/mptcp_net-next/issues/377
    Fixes: afe4fd062416 ("pkt_sched: fq: Fair Queue packet scheduler")
    Reviewed-by: Eric Dumazet <edumazet@google.com>
    Signed-off-by: Davide Caratti <dcaratti@redhat.com>
    Link:
https://lore.kernel.org/r/7b3a3c7e36d03068707a021760a194a8eb5ad41a.1682002300.git.dcaratti@r
edhat.com
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2a6b1ad842f09e18a7e99777287130f8d903c81f
Author: Dan Carpenter <dan.carpenter@linaro.org>
Date:   Thu Apr 20 15:36:07 2023 +0300

    net: dpaa: Fix uninitialized variable in dpaa_stop()

    [ Upstream commit 461bb5b97049a149278f2c27a3aa12af16da6a2e ]

    The return value is not initialized on the success path.

    Fixes: 901bdff2f529 ("net: fman: Change return type of disable to void")
    Signed-off-by: Dan Carpenter <dan.carpenter@linaro.org>
    Acked-by: Madalin Bucur <madalin.bucur@oss.nxp.com>
    Reviewed-by: Sean Anderson <sean.anderson@seco.com>
    Link: https://lore.kernel.org/r/8c9dc377-8495-495f-a4e5-4d2d0ee12f0c@kili.mountain
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit bd8cdec60fb62098d3e6fac16836797a789b9ca6
Author: Florian Westphal <fw@strlen.de>
Date:   Thu Apr 13 17:13:19 2023 +0200

    netfilter: nf_tables: don't write table validation state without mutex

    [ Upstream commit 9a32e9850686599ed194ccdceb6cd3dd56b2d9b9 ]

    The ->cleanup callback needs to be removed, this doesn't work anymore as
    the transaction mutex is already released in the ->abort function.

    Just do it after a successful validation pass, this either happens
    from commit or abort phases where transaction mutex is held.

    Fixes: f102d66b335a ("netfilter: nf_tables: use dedicated mutex to guard transactions")
    Signed-off-by: Florian Westphal <fw@strlen.de>
    Signed-off-by: Pablo Neira Ayuso <pablo@netfilter.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 83781c60017c3e219cea6cca0141823ae6dbd050
```

```
Author: Stanislav Fomichev <sdf@google.com>
Date:   Tue Apr 18 15:53:38 2023 -0700

    bpf: Don't EFAULT for getsockopt with optval=NULL

    [ Upstream commit 00e74ae0863827d944e36e56a4ce1e77e50edb91 ]

    Some socket options do getsockopt with optval=NULL to estimate the size
    of the final buffer (which is returned via optlen). This breaks BPF
    getsockopt assumptions about permitted optval buffer size. Let's enforce
    these assumptions only when non-NULL optval is provided.

    Fixes: 0d01da6afc54 ("bpf: implement getsockopt and setsockopt hooks")
    Reported-by: Martin KaFai Lau <martin.lau@kernel.org>
    Signed-off-by: Stanislav Fomichev <sdf@google.com>
    Signed-off-by: Daniel Borkmann <daniel@iogearbox.net>
    Link:
https://lore.kernel.org/bpf/ZD7Js4fj5YyI2oLd@google.com/T/#mb68daf700f87a9244a15d01d00c3f0e5
b08f49f7
    Link: https://lore.kernel.org/bpf/20230418225343.553806-2-sdf@google.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 642b7bbc5a6569e85005d867fa315ccb91056377
Author: Alexei Starovoitov <ast@kernel.org>
Date:   Thu Apr 20 18:49:01 2023 -0700

    bpf: Fix race between btf_put and btf_idr walk.

    [ Upstream commit acf1c3d68e9a31f10d92bc67ad4673cdae5e8d92 ]

    Florian and Eduard reported hard dead lock:
    [   58.433327]  _raw_spin_lock_irqsave+0x40/0x50
    [   58.433334]  btf_put+0x43/0x90
    [   58.433338]  bpf_find_btf_id+0x157/0x240
    [   58.433353]  btf_parse_fields+0x921/0x11c0

    This happens since btf->refcount can be 1 at the time of btf_put() and
    btf_put() will call btf_free_id() which will try to grab btf_idr_lock
    and will dead lock.
    Avoid the issue by doing btf_put() without locking.

    Fixes: 3d78417b60fb ("bpf: Add bpf_btf_find_by_name_kind() helper.")
    Fixes: 1e89106da253 ("bpf: Add bpf_core_add_cands() and wire it into
bpf_core_apply_relo_insn().")
    Reported-by: Florian Westphal <fw@strlen.de>
    Reported-by: Eduard Zingerman <eddyz87@gmail.com>
    Signed-off-by: Alexei Starovoitov <ast@kernel.org>
    Signed-off-by: Daniel Borkmann <daniel@iogearbox.net>
    Tested-by: Eduard Zingerman <eddyz87@gmail.com>
    Link: https://lore.kernel.org/bpf/20230421014901.70908-1-alexei.starovoitov@gmail.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1559484d837eafa3101424e59304afcfcc0c797d
Author: Yan Wang <rk.code@outlook.com>
Date:   Wed Apr 19 22:13:46 2023 +0800

    net: stmmac:fix system hang when setting up tag_8021q VLAN for DSA ports

    [ Upstream commit 35226750f7ab9d49140d95bc7d38a2a9b0f4fdfc ]

    The system hang because of dsa_tag_8021q_port_setup()->
                                    stmmac_vlan_rx_add_vid().

    I found in stmmac_drv_probe() that cailing pm_runtime_put()
    disabled the clock.
```

```
    First, when the kernel is compiled with CONFIG_PM=y,The stmmac's
    resume/suspend is active.

    Secondly,stmmac as DSA master,the dsa_tag_8021q_port_setup() function
    will callback stmmac_vlan_rx_add_vid when DSA dirver starts. However,
    The system is hanged for the stmmac_vlan_rx_add_vid() accesses its
    registers after stmmac's clock is closed.

    I would suggest adding the pm_runtime_resume_and_get() to the
    stmmac_vlan_rx_add_vid().This guarantees that resuming clock output
    while in use.

    Fixes: b3dcb3127786 ("net: stmmac: correct clocks enabled in stmmac_vlan_rx_kill_vid()")
    Reviewed-by: Jacob Keller <jacob.e.keller@intel.com>
    Signed-off-by: Yan Wang <rk.code@outlook.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e519b229bc02e7697f2dca529d9e5e48eefd4240
Author: Vlad Buslov <vladbu@nvidia.com>
Date:   Thu Apr 13 15:06:59 2023 +0200

    Revert "net/mlx5e: Don't use termination table when redundant"

    [ Upstream commit 081abcacaf0a9505c0d4cc9780b12e6ce5d33630 ]

    This reverts commit 14624d7247fcd0f3114a6f5f17b3c8d1020fbbb7.

    The termination table usage is requires for DMFS steering mode as firmware
    doesn't support mixed table destinations list which causes following
    syndrome with hairpin rules:

    [81922.283225] mlx5_core 0000:08:00.0: mlx5_cmd_out_err:803:(pid 25977):
SET_FLOW_TABLE_ENTRY(0x936) op_mod(0x0) failed, status bad parameter(0x3), syndrome
(0xaca205), err(-22)

    Fixes: 14624d7247fc ("net/mlx5e: Don't use termination table when redundant")
    Signed-off-by: Vlad Buslov <vladbu@nvidia.com>
    Reviewed-by: Roi Dayan <roid@nvidia.com>
    Reviewed-by: Maor Dickman <maord@nvidia.com>
    Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 17eb3ecc67cf97013bfdc1a4f767d243656668d5
Author: Aya Levin <ayal@nvidia.com>
Date:   Sun Mar 12 16:37:36 2023 +0200

    net/mlx5e: Nullify table pointer when failing to create

    [ Upstream commit 1b540decd03acd736693aabb6cb46c71fca38ae6 ]

    On failing to create promisc flow steering table, the pointer is
    returned with an error. Nullify it so unloading the driver won't try to
    destroy a non existing table.

    Failing to create promisc table may happen over BF devices when the ARM
    side is going through a firmware tear down. The host side start a
    reload flow. While the driver unloads, it tries to remove the promisc
    table. Remove WARN in this state as it is a valid error flow.

    Fixes: 1c46d7409f30 ("net/mlx5e: Optimize promiscuous mode")
    Signed-off-by: Aya Levin <ayal@nvidia.com>
    Reviewed-by: Tariq Toukan <tariqt@nvidia.com>
    Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
```

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

commit cdd3a93fb02fac732cbb1e1b468bc99d9e5a51b6
Author: Moshe Shemesh <moshe@nvidia.com>
Date:   Sun Apr 2 06:49:53 2023 +0300

    net/mlx5: Use recovery timeout on sync reset flow

    [ Upstream commit dfad99750c0f83b0242572a573afa2c055f85b36 ]

    Use the same timeout for sync reset flow and health recovery flow, since
    the former involves driver's recovery from firmware reset, which is
    similar to health recovery. Otherwise, in some cases, such as a firmware
    upgrade on the DPU, the firmware pre-init bit may not be ready within
    current timeout and the driver will abort loading back after reset.

    Signed-off-by: Moshe Shemesh <moshe@nvidia.com>
    Fixes: 37ca95e62ee2 ("net/mlx5: Increase FW pre-init timeout for health recovery")
    Reviewed-by: Maher Sanalla <msanalla@nvidia.com>
    Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit aced4f01a2c9d619fba76cf5467467d091284539
Author: Moshe Shemesh <moshe@nvidia.com>
Date:   Sun Apr 9 10:44:31 2023 +0300

    Revert "net/mlx5: Remove "recovery" arg from mlx5_load_one() function"

    [ Upstream commit 21608a2cf38e9f743636b5f86507ffbca18ecee7 ]

    This reverts commit 5977ac3910f1cbaf44dca48179118b25c206ac29.

    Revert this patch as we need the "recovery" arg back in mlx5_load_one()
    function. This arg will be used in the next patch for using recovery
    timeout during sync reset flow.

    Signed-off-by: Moshe Shemesh <moshe@nvidia.com>
    Reviewed-by: Maher Sanalla <msanalla@nvidia.com>
    Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
    Stable-dep-of: dfad99750c0f ("net/mlx5: Use recovery timeout on sync reset flow")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 31f53648d3c9e7b64fda063ee0f12b5c833332a1
Author: Roi Dayan <roid@nvidia.com>
Date:   Wed Mar 29 15:24:32 2023 +0300

    net/mlx5e: Fix error flow in representor failing to add vport rx rule

    [ Upstream commit 0a6b069cc60d68d33b4f6e7dd7f1adc3ec749766 ]

    On representor init rx error flow the flow steering pointer is being
    released so mlx5e_attach_netdev() doesn't have a valid fs pointer
    in its error flow. Make sure the pointer is nullified when released
    and add a check in mlx5e_fs_cleanup() to verify fs is not null
    as representor cleanup callback would be called anyway.

    Fixes: af8bbf730068 ("net/mlx5e: Convert mlx5e_flow_steering member of mlx5e_priv to
pointer")
    Signed-off-by: Roi Dayan <roid@nvidia.com>
    Reviewed-by: Maor Dickman <maord@nvidia.com>
    Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ef199a15edc39a1065530ed10091249e3f5810ca
Author: Chris Mi <cmi@nvidia.com>
```

```
Date:   Tue Apr 4 03:45:04 2023 +0300

    net/mlx5: Release tunnel device after tc update skb

    [ Upstream commit 4fbef0f8ea6350eaea89b1e3175f9325252913ac ]

    The cited commit causes a regression. Tunnel device is not released
    after tc update skb if skb needs to be freed. The following error
    message will be printed:

      unregister_netdevice: waiting for vxlan1 to become free. Usage count = 11

    Fix it by releasing tunnel device if skb needs to be freed.

    Fixes: 93a1ab2c545b ("net/mlx5: Refactor tc miss handling to a single function")
    Signed-off-by: Chris Mi <cmi@nvidia.com>
    Reviewed-by: Maor Dickman <maord@nvidia.com>
    Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 984819fa3f5354fdf5465035ec3d4bba2830d5f7
Author: Chris Mi <cmi@nvidia.com>
Date:   Fri Mar 10 09:56:08 2023 +0200

    net/mlx5: E-switch, Don't destroy indirect table in split rule

    [ Upstream commit 4c8189302567f75099a336b0efcff8291ec86ff4 ]

    Source port rewrite (forward to ovs internal port or statck device) isn't
    supported in the rule of split action. So there is no indirect table in
    split rule. The cited commit destroyes indirect table in split rule. The
    indirect table for other rules will be destroyed wrongly. It will cause
    traffic loss.

    Fix it by removing the destroy function in split rule. And also remove
    the destroy function in error flow.

    Fixes: 10742efc20a4 ("net/mlx5e: VF tunnel TX traffic offloading")
    Signed-off-by: Chris Mi <cmi@nvidia.com>
    Reviewed-by: Roi Dayan <roid@nvidia.com>
    Reviewed-by: Maor Dickman <maord@nvidia.com>
    Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 244ee2b05997441f5a5b506fcddf44d26bb83497
Author: Chris Mi <cmi@nvidia.com>
Date:   Fri Mar 10 11:06:48 2023 +0200

    net/mlx5: E-switch, Create per vport table based on devlink encap mode

    [ Upstream commit fd745f4c0abe41ebb09d11bf622b054a0f3e7b49 ]

    Currently when creating per vport table, create flags are hardcoded.
    Devlink encap mode is set based on user input and HW capability.
    Create per vport table based on devlink encap mode.

    Fixes: c796bb7cd230 ("net/mlx5: E-switch, Generalize per vport table API")
    Signed-off-by: Chris Mi <cmi@nvidia.com>
    Reviewed-by: Roi Dayan <roid@nvidia.com>
    Reviewed-by: Maor Dickman <maord@nvidia.com>
    Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3db903a71f1f4bbf30baae166a4a49f2e8aceb61
Author: Vlad Buslov <vladbu@nvidia.com>
```

```
Date:   Fri Mar 24 17:52:17 2023 +0100

    net/mlx5e: Release the label when replacing existing ct entry

    [ Upstream commit 8ac04a28144cfa89b61be518268233742c23d88d ]

    Cited commit doesn't release the label mapping when replacing existing ct
    entry which leads to following memleak report:

    unreferenced object 0xffff8881854cf280 (size 96):
      comm "kworker/u48:74", pid 23093, jiffies 4296664564 (age 175.944s)
      hex dump (first 32 bytes):
        00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
        00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
      backtrace:
        [<000000002722d368>] __kmalloc+0x4b/0x1c0
        [<00000000cc44e18f>] mapping_add+0x6e8/0xc90 [mlx5_core]
        [<000000003ad942a7>] mlx5_get_label_mapping+0x66/0xe0 [mlx5_core]
        [<00000000266308ac>] mlx5_tc_ct_entry_create_mod_hdr+0x1c4/0xf50 [mlx5_core]
        [<000000009a768b4f>] mlx5_tc_ct_entry_add_rule+0x16f/0xaf0 [mlx5_core]
        [<00000000a178f3e5>] mlx5_tc_ct_block_flow_offload_add+0x10cb/0x1f90 [mlx5_core]
        [<000000007b46c496>] mlx5_tc_ct_block_flow_offload+0x14a/0x630 [mlx5_core]
        [<00000000a9a18ac5>] nf_flow_offload_tuple+0x1a3/0x390 [nf_flow_table]
        [<00000000d0881951>] flow_offload_work_handler+0x257/0xd30 [nf_flow_table]
        [<000000009e4935a4>] process_one_work+0x7c2/0x13e0
        [<00000000f5cd36a7>] worker_thread+0x59d/0xec0
        [<00000000baed1daf>] kthread+0x28f/0x330
        [<0000000063d282a4>] ret_from_fork+0x1f/0x30

    Fix the issue by correctly releasing the label mapping.

    Fixes: 94ceffb48eac ("net/mlx5e: Implement CT entry update")
    Signed-off-by: Vlad Buslov <vladbu@nvidia.com>
    Reviewed-by: Roi Dayan <roid@nvidia.com>
    Reviewed-by: Paul Blakey <paulb@nvidia.com>
    Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2d57a514f9ab7d2d40f49b02d93edfcec8c78a9e
Author: Vlad Buslov <vladbu@nvidia.com>
Date:   Wed Mar 22 09:52:26 2023 +0100

    net/mlx5e: Don't clone flow post action attributes second time

    [ Upstream commit e9fce818fe003b6c527f25517b9ac08eb4661b5d ]

    The code already clones post action attributes in
    mlx5e_clone_flow_attr_for_post_act(). Creating another copy in
    mlx5e_tc_post_act_add() is a erroneous leftover from original
    implementation. Instead, assign handle->attribute to post_attr provided by
    the caller. Note that cloning the attribute second time is not just
    wasteful but also causes issues like second copy not being properly updated
    in neigh update code which leads to following use-after-free:

    Feb 21 09:02:00 c-237-177-40-045 kernel: BUG: KASAN: use-after-free in
mlx5_cmd_set_fte+0x200d/0x24c0 [mlx5_core]
    Feb 21 09:02:00 c-237-177-40-045 kernel:  kasan_report+0xbb/0x1a0
    Feb 21 09:02:00 c-237-177-40-045 kernel:  kasan_save_stack+0x1e/0x40
    Feb 21 09:02:00 c-237-177-40-045 kernel:  kasan_set_track+0x21/0x30
    Feb 21 09:02:00 c-237-177-40-045 kernel:  __kasan_kmalloc+0x7a/0x90
    Feb 21 09:02:00 c-237-177-40-045 kernel:  kasan_save_stack+0x1e/0x40
    Feb 21 09:02:00 c-237-177-40-045 kernel:  kasan_set_track+0x21/0x30
    Feb 21 09:02:00 c-237-177-40-045 kernel:  kasan_save_free_info+0x2a/0x40
    Feb 21 09:02:00 c-237-177-40-045 kernel:  ____kasan_slab_free+0x11a/0x1b0
    Feb 21 09:02:00 c-237-177-40-045 kernel: page dumped because: kasan: bad access detected
```

```
    Feb 21 09:02:00 c-237-177-40-045 kernel: mlx5_core 0000:08:00.0: mlx5_cmd_out_err:803:
(pid 8833): SET_FLOW_TABLE_ENTRY(0x936) op_mod(0x0) failed, status bad resource state(0x9),
syndrome (0xf2ff71), err(-22)
    Feb 21 09:02:00 c-237-177-40-045 kernel: mlx5_core 0000:08:00.0 enp8s0f0: Failed to add
post action rule
    Feb 21 09:02:00 c-237-177-40-045 kernel: mlx5_core 0000:08:00.0:
mlx5e_tc_encap_flows_add:190:(pid 8833): Failed to update flow post acts, -22
    Feb 21 09:02:00 c-237-177-40-045 kernel: Call Trace:
    Feb 21 09:02:00 c-237-177-40-045 kernel:  <TASK>
    Feb 21 09:02:00 c-237-177-40-045 kernel:  dump_stack_lvl+0x57/0x7d
    Feb 21 09:02:00 c-237-177-40-045 kernel:  print_report+0x170/0x471
    Feb 21 09:02:00 c-237-177-40-045 kernel:  ? mlx5_cmd_set_fte+0x200d/0x24c0 [mlx5_core]
    Feb 21 09:02:00 c-237-177-40-045 kernel:  kasan_report+0xbb/0x1a0
    Feb 21 09:02:00 c-237-177-40-045 kernel:  ? mlx5_cmd_set_fte+0x200d/0x24c0 [mlx5_core]
    Feb 21 09:02:00 c-237-177-40-045 kernel:  mlx5_cmd_set_fte+0x200d/0x24c0 [mlx5_core]
    Feb 21 09:02:00 c-237-177-40-045 kernel:  ? __module_address.part.0+0x62/0x200
    Feb 21 09:02:00 c-237-177-40-045 kernel:  ? mlx5_cmd_stub_create_flow_table+0xd0/0xd0
[mlx5_core]
    Feb 21 09:02:00 c-237-177-40-045 kernel:  ? __raw_spin_lock_init+0x3b/0x110
    Feb 21 09:02:00 c-237-177-40-045 kernel:  mlx5_cmd_create_fte+0x80/0xb0 [mlx5_core]
    Feb 21 09:02:00 c-237-177-40-045 kernel:  add_rule_fg+0xe80/0x19c0 [mlx5_core]
    --
    Feb 21 09:02:00 c-237-177-40-045 kernel: Allocated by task 13476:
    Feb 21 09:02:00 c-237-177-40-045 kernel:  kasan_save_stack+0x1e/0x40
    Feb 21 09:02:00 c-237-177-40-045 kernel:  kasan_set_track+0x21/0x30
    Feb 21 09:02:00 c-237-177-40-045 kernel:  __kasan_kmalloc+0x7a/0x90
    Feb 21 09:02:00 c-237-177-40-045 kernel:  mlx5_packet_reformat_alloc+0x7b/0x230
[mlx5_core]
    Feb 21 09:02:00 c-237-177-40-045 kernel:  mlx5e_tc_tun_create_header_ipv4+0x977/0xf10
[mlx5_core]
    Feb 21 09:02:00 c-237-177-40-045 kernel:  mlx5e_attach_encap+0x15b4/0x1e10 [mlx5_core]
    Feb 21 09:02:00 c-237-177-40-045 kernel:  post_process_attr+0x305/0xa30 [mlx5_core]
    Feb 21 09:02:00 c-237-177-40-045 kernel:  mlx5e_tc_add_fdb_flow+0x4c0/0xcf0 [mlx5_core]
    Feb 21 09:02:00 c-237-177-40-045 kernel:  __mlx5e_add_fdb_flow+0x7cf/0xe90 [mlx5_core]
    Feb 21 09:02:00 c-237-177-40-045 kernel:  mlx5e_configure_flower+0xcaa/0x4b90
[mlx5_core]
    Feb 21 09:02:00 c-237-177-40-045 kernel:  mlx5e_rep_setup_tc_cls_flower+0x99/0x1b0
[mlx5_core]
    Feb 21 09:02:00 c-237-177-40-045 kernel:  mlx5e_rep_setup_tc_cb+0x133/0x1e0 [mlx5_core]
    --
    Feb 21 09:02:00 c-237-177-40-045 kernel: Freed by task 8833:
    Feb 21 09:02:00 c-237-177-40-045 kernel:  kasan_save_stack+0x1e/0x40
    Feb 21 09:02:00 c-237-177-40-045 kernel:  kasan_set_track+0x21/0x30
    Feb 21 09:02:00 c-237-177-40-045 kernel:  kasan_save_free_info+0x2a/0x40
    Feb 21 09:02:00 c-237-177-40-045 kernel:  ____kasan_slab_free+0x11a/0x1b0
    Feb 21 09:02:00 c-237-177-40-045 kernel:  __kmem_cache_free+0x1de/0x400
    Feb 21 09:02:00 c-237-177-40-045 kernel:  mlx5_packet_reformat_dealloc+0xad/0x100
[mlx5_core]
    Feb 21 09:02:00 c-237-177-40-045 kernel:  mlx5e_tc_encap_flows_del+0x3c0/0x500
[mlx5_core]
    Feb 21 09:02:00 c-237-177-40-045 kernel:  mlx5e_rep_update_flows+0x40c/0xa80 [mlx5_core]
    Feb 21 09:02:00 c-237-177-40-045 kernel:  mlx5e_rep_neigh_update+0x473/0x7a0 [mlx5_core]
    Feb 21 09:02:00 c-237-177-40-045 kernel:  process_one_work+0x7c2/0x1310
    Feb 21 09:02:00 c-237-177-40-045 kernel:  worker_thread+0x59d/0xec0
    Feb 21 09:02:00 c-237-177-40-045 kernel:  kthread+0x28f/0x330

    Fixes: 8300f225268b ("net/mlx5e: Create new flow attr for multi table actions")
    Signed-off-by: Vlad Buslov <vladbu@nvidia.com>
    Reviewed-by: Roi Dayan <roid@nvidia.com>
    Signed-off-by: Saeed Mahameed <saeedm@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d2ebc4d67bff4a9efa05bdf9494f4dec5c23db1d
Author: Joe Damato <jdamato@fastly.com>
Date:   Sun Apr 16 19:12:23 2023 +0000
```

```
    ixgbe: Enable setting RSS table to default values

    [ Upstream commit e85d3d55875f7a1079edfbc4e4e98d6f8aea9ac7 ]

    ethtool uses `ETHTOOL_GRXRINGS` to compute how many queues are supported
    by RSS. The driver should return the smaller of either:
      - The maximum number of RSS queues the device supports, OR
      - The number of RX queues configured

    Prior to this change, running `ethtool -X $iface default` fails if the
    number of queues configured is larger than the number supported by RSS,
    even though changing the queue count correctly resets the flowhash to
    use all supported queues.

    Other drivers (for example, i40e) will succeed but the flow hash will
    reset to support the maximum number of queues supported by RSS, even if
    that amount is smaller than the configured amount.

    Prior to this change:

    $ sudo ethtool -L eth1 combined 20
    $ sudo ethtool -x eth1
    RX flow hash indirection table for eth1 with 20 RX ring(s):
        0:      0     1     2     3     4     5     6     7
        8:      8     9    10    11    12    13    14    15
       16:      0     1     2     3     4     5     6     7
       24:      8     9    10    11    12    13    14    15
       32:      0     1     2     3     4     5     6     7
    ...

    You can see that the flowhash was correctly set to use the maximum
    number of queues supported by the driver (16).

    However, asking the NIC to reset to "default" fails:

    $ sudo ethtool -X eth1 default
    Cannot set RX flow hash configuration: Invalid argument

    After this change, the flowhash can be reset to default which will use
    all of the available RSS queues (16) or the configured queue count,
    whichever is smaller.

    Starting with eth1 which has 10 queues and a flowhash distributing to
    all 10 queues:

    $ sudo ethtool -x eth1
    RX flow hash indirection table for eth1 with 10 RX ring(s):
        0:      0     1     2     3     4     5     6     7
        8:      8     9     0     1     2     3     4     5
       16:      6     7     8     9     0     1     2     3
    ...

    Increasing the queue count to 48 resets the flowhash to distribute to 16
    queues, as it did before this patch:

    $ sudo ethtool -L eth1 combined 48
    $ sudo ethtool -x eth1
    RX flow hash indirection table for eth1 with 16 RX ring(s):
        0:      0     1     2     3     4     5     6     7
        8:      8     9    10    11    12    13    14    15
       16:      0     1     2     3     4     5     6     7
    ...

    Due to the other bugfix in this series, the flowhash can be set to use
```

```
    queues 0-5:

    $ sudo ethtool -X eth1 equal 5
    $ sudo ethtool -x eth1
    RX flow hash indirection table for eth1 with 16 RX ring(s):
        0:      0     1     2     3     4     0     1     2
        8:      3     4     0     1     2     3     4     0
       16:      1     2     3     4     0     1     2     3
    ...

    Due to this bugfix, the flowhash can be reset to default and use 16
    queues:

    $ sudo ethtool -X eth1 default
    $ sudo ethtool -x eth1
    RX flow hash indirection table for eth1 with 16 RX ring(s):
        0:      0     1     2     3     4     5     6     7
        8:      8     9    10    11    12    13    14    15
       16:      0     1     2     3     4     5     6     7
    ...

    Fixes: 91cd94bfe4f0 ("ixgbe: add basic support for setting and getting nfc controls")
    Signed-off-by: Joe Damato <jdamato@fastly.com>
    Reviewed-by: Sridhar Samudrala <sridhar.samudrala@intel.com>
    Tested-by: Pucha Himasekhar Reddy <himasekharx.reddy.pucha@intel.com> (A Contingent
worker at Intel)
    Signed-off-by: Tony Nguyen <anthony.l.nguyen@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cb1d102451ba48318fad6784cb5ce18f49c11efa
Author: Joe Damato <jdamato@fastly.com>
Date:   Sun Apr 16 19:12:22 2023 +0000

    ixgbe: Allow flow hash to be set via ethtool

    [ Upstream commit 4f3ed1293feb9502dc254b05802faf1ad3317ac6 ]

    ixgbe currently returns `EINVAL` whenever the flowhash it set by ethtool
    because the ethtool code in the kernel passes a non-zero value for hfunc
    that ixgbe should allow.

    When ethtool is called with `ETHTOOL_SRXFHINDIR`,
    `ethtool_set_rxfh_indir` will call ixgbe's set_rxfh function
    with `ETH_RSS_HASH_NO_CHANGE`. This value should be accepted.

    When ethtool is called with `ETHTOOL_SRSSH`, `ethtool_set_rxfh` will
    call ixgbe's set_rxfh function with `rxfh.hfunc`, which appears to be
    hardcoded in ixgbe to always be `ETH_RSS_HASH_TOP`. This value should
    also be accepted.

    Before this patch:

    $ sudo ethtool -L eth1 combined 10
    $ sudo ethtool -X eth1 default
    Cannot set RX flow hash configuration: Invalid argument

    After this patch:

    $ sudo ethtool -L eth1 combined 10
    $ sudo ethtool -X eth1 default
    $ sudo ethtool -x eth1
    RX flow hash indirection table for eth1 with 10 RX ring(s):
        0:      0     1     2     3     4     5     6     7
        8:      8     9     0     1     2     3     4     5
       16:      6     7     8     9     0     1     2     3
```

```
        24:      4     5     6     7     8     9     0     1
        ...

    Fixes: 1c7cf0784e4d ("ixgbe: support for ethtool set_rxfh")
    Signed-off-by: Joe Damato <jdamato@fastly.com>
    Reviewed-by: Sridhar Samudrala <sridhar.samudrala@intel.com>
    Tested-by: Pucha Himasekhar Reddy <himasekharx.reddy.pucha@intel.com> (A Contingent
worker at Intel)
    Signed-off-by: Tony Nguyen <anthony.l.nguyen@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2371e1ecd445baf793a74db00ea6b2a2bc13c4c0
Author: Zhengchao Shao <shaozhengchao@huawei.com>
Date:   Tue Apr 18 14:54:50 2023 +0800

    net: libwx: fix memory leak in wx_setup_rx_resources

    [ Upstream commit e315e7b83a22043bffee450437d7089ef373cbf6 ]

    When wx_alloc_page_pool() failed in wx_setup_rx_resources(), it doesn't
    release DMA buffer. Add dma_free_coherent() in the error path to release
    the DMA buffer.

    Fixes: 850b971110b2 ("net: libwx: Allocate Rx and Tx resources")
    Signed-off-by: Zhengchao Shao <shaozhengchao@huawei.com>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Link: https://lore.kernel.org/r/20230418065450.2268522-1-shaozhengchao@huawei.com
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 118d45787fec9b325f1eab2f30a0831aca976f54
Author: Johannes Berg <johannes.berg@intel.com>
Date:   Tue Apr 18 12:28:08 2023 +0300

    wifi: iwlwifi: mvm: fix potential memory leak

    [ Upstream commit 457d7fb03e6c3d73fbb509bd85fc4b02d1ab405e ]

    If we do get multiple notifications from firmware, then
    we might have allocated 'notif', but don't free it. Fix
    that by checking for duplicates before allocation.

    Fixes: 4da46a06d443 ("wifi: iwlwifi: mvm: Add support for wowlan info notification")
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link:
https://lore.kernel.org/r/20230418122405.116758321cc4.I8bdbcbb38c89ac637eaa20dda58fa9165b258
93a@changeid
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c21315e6d2ec37dc7896e2a4e5bce8b435a598b4
Author: Haim Dreyfuss <haim.dreyfuss@intel.com>
Date:   Thu Apr 13 21:40:27 2023 +0300

    wifi: iwlwifi: mvm: support wowlan info notification version 2

    [ Upstream commit 905d50ddbc83bef0d7f3386e7f3472b0324b405b ]

    As part of version 2 we don't need to have wake_packet_bufsize
    and wake_packet_length. The first one is already calculated by the driver,
    the latter is sent as part of the wake packet notification.

    Signed-off-by: Haim Dreyfuss <haim.dreyfuss@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
```

```
    Link:
https://lore.kernel.org/r/20230413213309.3b53213b10d4.Ibf2f15aca614def2d262dd267d1aad65931b5
8f1@changeid
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Stable-dep-of: 457d7fb03e6c ("wifi: iwlwifi: mvm: fix potential memory leak")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b830ba20b43be52eae7d4087b61a0079dec56820
Author: Johannes Berg <johannes.berg@intel.com>
Date:   Tue Apr 18 12:28:05 2023 +0300

    wifi: iwlwifi: fw: fix memory leak in debugfs

    [ Upstream commit 3d90d2f4a018fe8cfd65068bc6350b6222be4852 ]

    Fix a memory leak that occurs when reading the fw_info
    file all the way, since we return NULL indicating no
    more data, but don't free the status tracking object.

    Fixes: 36dfe9ac6e8b ("iwlwifi: dump api version in yaml format")
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link:
https://lore.kernel.org/r/20230418122405.239e501b3b8d.I4268f87809ef91209cbcd748eee0863195e70
fa2@changeid
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f612ae1ab4793701caf39386fb3b7f4b3ef44e48
Author: Tzung-Bi Shih <tzungbi@kernel.org>
Date:   Wed Apr 19 13:15:26 2023 +0800

    netfilter: conntrack: fix wrong ct->timeout value

    [ Upstream commit 73db1b8f2bb6725b7391e85aab41fdf592b3c0c1 ]

    (struct nf_conn)->timeout is an interval before the conntrack
    confirmed.  After confirmed, it becomes a timestamp.

    It is observed that timeout of an unconfirmed conntrack:
    - Set by calling ctnetlink_change_timeout(). As a result,
      `nfct_time_stamp` was wrongly added to `ct->timeout` twice.
    - Get by calling ctnetlink_dump_timeout(). As a result,
      `nfct_time_stamp` was wrongly subtracted.

    Call Trace:
     <TASK>
     dump_stack_lvl
     ctnetlink_dump_timeout
     __ctnetlink_glue_build
     ctnetlink_glue_build
     __nfqnl_enqueue_packet
     nf_queue
     nf_hook_slow
     ip_mc_output
     ? __pfx_ip_finish_output
     ip_send_skb
     ? __pfx_dst_output
     udp_send_skb
     udp_sendmsg
     ? __pfx_ip_generic_getfrag
     sock_sendmsg

    Separate the 2 cases in:
    - Setting `ct->timeout` in __nf_ct_set_timeout().
```

```
      - Getting `ct->timeout` in ctnetlink_dump_timeout().

      Pablo appends:

      Update ctnetlink to set up the timeout _after_ the IPS_CONFIRMED flag is
      set on, otherwise conntrack creation via ctnetlink breaks.

      Note that the problem described in this patch occurs since the
      introduction of the nfnetlink_queue conntrack support, select a
      sufficiently old Fixes: tag for -stable kernel to pick up this fix.

      Fixes: a4b4766c3ceb ("netfilter: nfnetlink_queue: rename related to nfqueue attaching
  conntrack info")
      Signed-off-by: Tzung-Bi Shih <tzungbi@kernel.org>
      Signed-off-by: Pablo Neira Ayuso <pablo@netfilter.org>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit fc3f262d1644a0ba92438f008a275b31d312115d
  Author: Ryder Lee <ryder.lee@mediatek.com>
  Date:   Wed Apr 19 13:58:40 2023 +0800

      wifi: mt76: mt7996: fill txd by host driver

      [ Upstream commit 3b522cadedfe6e9e0e8193d7d4ab5aa8d0c73209 ]

      The hardware SDO has issue to fill txd for the moment, so fallback to
      driver filling method.

      Fixes: 98686cd21624 (wifi: mt76: mt7996: add driver for MediaTek Wi-Fi 7 (802.11be)
  devices)
      Signed-off-by: Ryder Lee <ryder.lee@mediatek.com>
      Signed-off-by: Felix Fietkau <nbd@nbd.name>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 22ee0f52da958dda69bc1cc1d91146cab04a52cb
  Author: Pablo Neira Ayuso <pablo@netfilter.org>
  Date:   Tue Apr 18 23:31:26 2023 +0200

      netfilter: conntrack: restore IPS_CONFIRMED out of nf_conntrack_hash_check_insert()

      [ Upstream commit 2cdaa3eefed83082923cf219c8b6a314e622da74 ]

      e6d57e9ff0ae ("netfilter: conntrack: fix rmmod double-free race")
      consolidates IPS_CONFIRMED bit set in nf_conntrack_hash_check_insert().
      However, this breaks ctnetlink:

       # conntrack -I -p tcp --timeout 123 --src 1.2.3.4 --dst 5.6.7.8 --state ESTABLISHED --
  sport 1 --dport 4 -u SEEN_REPLY
       conntrack v1.4.6 (conntrack-tools): Operation failed: Device or resource busy

      This is a partial revert of the aforementioned commit to restore
      IPS_CONFIRMED.

      Fixes: e6d57e9ff0ae ("netfilter: conntrack: fix rmmod double-free race")
      Reported-by: Stéphane Graber <stgraber@stgraber.org>
      Tested-by: Stéphane Graber <stgraber@stgraber.org>
      Signed-off-by: Pablo Neira Ayuso <pablo@netfilter.org>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit c8c97524b628c2bb40cf1de88713ea811e18ce61
  Author: Johannes Berg <johannes.berg@intel.com>
  Date:   Mon Apr 17 11:41:33 2023 +0300

      wifi: iwlwifi: mvm: check firmware response size
```

```
    [ Upstream commit 13513cec93ac9902d0b896976d8bab3758a9881c ]

    Check the firmware response size for responses to the
    memory read/write command in debugfs before using it.

    Fixes: 2b55f43f8e47 ("iwlwifi: mvm: Add mem debugfs entry")
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link:
https://lore.kernel.org/r/20230417113648.0d56fcaf68ee.I70e9571f3ed7263929b04f8fabad23c9b999e
4ea@changeid
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 628d53866ac49a83f3ee29054907da5a1d914206
Author: Ryder Lee <ryder.lee@mediatek.com>
Date:   Fri Apr 14 04:23:29 2023 +0800

    wifi: mt76: connac: fix txd multicast rate setting

    [ Upstream commit 3d2892e05086d09aecf14ea64b2debbf495e313c ]

    The vif->bss_conf.mcast_rate should be applied to multicast data frame
    only.

    Fixes: 182071cdd594 ("mt76: connac: move connac2_mac_write_txwi in mt76_connac module")
    Signed-off-by: Ryder Lee <ryder.lee@mediatek.com>
    Signed-off-by: Felix Fietkau <nbd@nbd.name>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 88ec7baa4d81e5c45e2410940218c7630e51b320
Author: Quan Zhou <quan.zhou@mediatek.com>
Date:   Thu Apr 13 05:11:14 2023 +0800

    wifi: mt76: mt7921e: stop chip reset worker in unregister hook

    [ Upstream commit 3d78c46423c6567ed25ca033e086865b1b4d5ae1 ]

    If the chip reset worker is triggered during the remove process, the chip
    DMA may not be properly pushed back to the idle state. This can lead to
    corruption of the DMA flow due to the chip reset. Therefore, it is
    necessary to stop the chip reset before the DMA is finalized.

    To avoid resetting the chip after the reset worker is cancelled, use
    __mt7921_mcu_drv_pmctrl() instead of mt7921_mcu_drv_pmctrl(). It is safe to
    ignore the pm mutex because the pm worker and wake worker have already been
    cancelled.

    Fixes: 033ae79b3830 ("mt76: mt7921: refactor init.c to be bus independent")
    Co-developed-by: Sean Wang <sean.wang@mediatek.com>
    Signed-off-by: Sean Wang <sean.wang@mediatek.com>
    Co-developed-by: Deren Wu <deren.wu@mediatek.com>
    Signed-off-by: Deren Wu <deren.wu@mediatek.com>
    Co-developed-by: Wang Zhao <wang.zhao@mediatek.com>
    Signed-off-by: Wang Zhao <wang.zhao@mediatek.com>
    Signed-off-by: Quan Zhou <quan.zhou@mediatek.com>
    Signed-off-by: Felix Fietkau <nbd@nbd.name>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 34307143a8686d6ab585965bfd0a7c19ec3e0309
Author: Quan Zhou <quan.zhou@mediatek.com>
Date:   Thu Apr 13 05:11:13 2023 +0800

    wifi: mt76: mt7921e: improve reliability of dma reset
```

```
    [ Upstream commit 87714bf6ed1589813e473db5471e6e9857755764 ]

    The hardware team has advised the driver that it is necessary to first put
    WFDMA into an idle state before resetting the WFDMA. Otherwise, the WFDMA
    may enter an unknown state where it cannot be polled with the right state
    successfully. To ensure that the DMA can work properly while a stressful
    cold reboot test was being made, we have reordered the programming sequence
    in the driver based on the hardware team's guidance.

    The patch would modify the WFDMA disabling flow from

    "DMA reset -> disabling DMASHDL -> disabling WFDMA -> polling and waiting
    until DMA idle" to "disabling WFDMA -> polling and waiting for DMA idle ->
    disabling DMASHDL -> DMA reset.

    Where he polling and waiting until WFDMA is idle is coordinated with the
    operation of disabling WFDMA. Even while WFDMA is being disabled, it can
    still handle Tx/Rx requests. The additional polling allows sufficient time
    for WFDMA to process the last T/Rx request. When the idle state of WFDMA is
    reached, it is a reliable indication that DMASHDL is also idle to ensure it
    is safe to disable it and perform the DMA reset.

    Fixes: 0a1059d0f060 ("mt76: mt7921: move mt7921_dma_reset in dma.c")
    Co-developed-by: Sean Wang <sean.wang@mediatek.com>
    Signed-off-by: Sean Wang <sean.wang@mediatek.com>
    Co-developed-by: Deren Wu <deren.wu@mediatek.com>
    Signed-off-by: Deren Wu <deren.wu@mediatek.com>
    Co-developed-by: Wang Zhao <wang.zhao@mediatek.com>
    Signed-off-by: Wang Zhao <wang.zhao@mediatek.com>
    Signed-off-by: Quan Zhou <quan.zhou@mediatek.com>
    Signed-off-by: Felix Fietkau <nbd@nbd.name>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 16e8f6bc44e001372a110faec2fd36eca619ce3d
Author: Jiefeng Li <jiefeng_li@hust.edu.cn>
Date:   Wed Apr 12 14:22:34 2023 +0800

    wifi: mt76: mt7921: fix missing unwind goto in `mt7921u_probe`

    [ Upstream commit 5c47cdebbaeb7724df6f9f46917c93e53f791547 ]

    `mt7921u_dma_init` can only return zero or negative number according to its
    definition. When it returns non-zero number, there exists an error and this
    function should handle this error rather than return directly.

    Fixes: 0d2afe09fad5 ("mt76: mt7921: add mt7921u driver")
    Signed-off-by: Jiefeng Li <jiefeng_li@hust.edu.cn>
    Reviewed-by: Dongliang Mu <dzm91@hust.edu.cn>
    Reviewed-by: Sridhar Samudrala <sridhar.samudrala@intel.com>
    Signed-off-by: Felix Fietkau <nbd@nbd.name>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ec4d97e8eddcfa9f63f2f62adec5fb4f941ba2ef
Author: Sean Wang <sean.wang@mediatek.com>
Date:   Wed Apr 12 05:23:11 2023 +0800

    mt76: mt7921: fix kernel panic by accessing unallocated eeprom.data

    [ Upstream commit 12db28c3ef31f719bd18fa186a40bb152e6a527c ]

    The MT7921 driver no longer uses eeprom.data, but the relevant code has not
    been removed completely since
    commit 16d98b548365 ("mt76: mt7921: rely on mcu_get_nic_capability").
    This could result in potential invalid memory access.
```

```
    To fix the kernel panic issue in mt7921, it is necessary to avoid accessing
    unallocated eeprom.data which can lead to invalid memory access.

    Furthermore, it is possible to entirely eliminate the
    mt7921_mcu_parse_eeprom function and solely depend on
    mt7921_mcu_parse_response to divide the RxD header.

    [2.702735] BUG: kernel NULL pointer dereference, address: 0000000000000550
    [2.702740] #PF: supervisor write access in kernel mode
    [2.702741] #PF: error_code(0x0002) - not-present page
    [2.702743] PGD 0 P4D 0
    [2.702747] Oops: 0002 [#1] PREEMPT SMP NOPTI
    [2.702755] RIP: 0010:mt7921_mcu_parse_response+0x147/0x170 [mt7921_common]
    [2.702758] RSP: 0018:ffffae7c00fef828 EFLAGS: 00010286
    [2.702760] RAX: ffffa367f57be024 RBX: ffffa367cc7bf500 RCX: 0000000000000000
    [2.702762] RDX: 0000000000000550 RSI: 0000000000000000 RDI: ffffa367cc7bf500
    [2.702763] RBP: ffffae7c00fef840 R08: ffffa367cb167000 R09: 0000000000000005
    [2.702764] R10: 0000000000000000 R11: ffffffffc04702e4 R12: ffffa367e8329f40
    [2.702766] R13: 0000000000000000 R14: 0000000000000001 R15: ffffa367e8329f40
    [2.702768] FS:  000079ee6cf20c40(0000) GS:ffffa36b2f940000(0000) knlGS:0000000000000000
    [2.702769] CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    [2.702775] CR2: 0000000000000550 CR3: 00000001233c6004 CR4: 0000000000770ee0
    [2.702776] PKRU: 55555554
    [2.702777] Call Trace:
    [2.702782]  mt76_mcu_skb_send_and_get_msg+0xc3/0x11e [mt76 <HASH:1bc4 5>]
    [2.702785]  mt7921_run_firmware+0x241/0x853 [mt7921_common <HASH:6a2f 6>]
    [2.702789]  mt7921e_mcu_init+0x2b/0x56 [mt7921e <HASH:d290 7>]
    [2.702792]  mt7921_register_device+0x2eb/0x5a5 [mt7921_common <HASH:6a2f 6>]
    [2.702795]  ? mt7921_irq_tasklet+0x1d4/0x1d4 [mt7921e <HASH:d290 7>]
    [2.702797]  mt7921_pci_probe+0x2d6/0x319 [mt7921e <HASH:d290 7>]
    [2.702799]  pci_device_probe+0x9f/0x12a

    Fixes: 16d98b548365 ("mt76: mt7921: rely on mcu_get_nic_capability")
    Signed-off-by: Sean Wang <sean.wang@mediatek.com>
    Signed-off-by: Felix Fietkau <nbd@nbd.name>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e291a6c683d7e3eced3788aed2671a548f7bf04a
Author: Ming Yen Hsieh <mingyen.hsieh@mediatek.com>
Date:   Thu Mar 23 21:26:12 2023 +0800

    wifi: mt76: fix 6GHz high channel not be scanned

    [ Upstream commit 23792cedaff02351b57bddd8957fc917fa88f2e0 ]

    mt76 scan command only support 64 channels currently. If the
    channel count is larger than 64(for 2+5+6GHz), some channels will
    not be scanned. Hence change the scan type to full channel scan
    in case of the command cannot include proper list for chip.

    Fixes: 399090ef9605 ("mt76: mt76_connac: move hw_scan and sched_scan routine in
mt76_connac_mcu module")
    Reported-by: Ben Greear <greearb@candelatech.com>
    Tested-by: Isaac Konikoff <konikofi@candelatech.com>
    Signed-off-by: Ming Yen Hsieh <mingyen.hsieh@mediatek.com>
    Signed-off-by: Deren Wu <deren.wu@mediatek.com>
    Signed-off-by: Felix Fietkau <nbd@nbd.name>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 05a9124291220dfaa60d0b2bf096b1050f8934e3
Author: Quan Zhou <quan.zhou@mediatek.com>
Date:   Sat Mar 18 15:41:12 2023 +0800

    wifi: mt76: mt7921e: fix probe timeout after reboot
```

```
    [ Upstream commit c397fc1e6365a2a9e5540a85b2c1d4ea412aa0e2 ]

    In system warm reboot scene, due to the polling timeout(now 1000us)
    is too short to wait dma idle in time, it may make driver probe fail
    with error code -ETIMEDOUT. Meanwhile, we also found the dma may take
    around 70ms to enter idle state. Change the polling idle timeout to
    100ms to avoid the probabilistic probe fail.

    Tested pass with 5000 times warm reboot on x86 platform.

    [4.477496] pci 0000:01:00.0: attach allowed to drvr mt7921e [internal device]
    [4.478306] mt7921e 0000:01:00.0: ASIC revision: 79610010
    [4.480063] mt7921e: probe of 0000:01:00.0 failed with error -110

    Fixes: 0a1059d0f060 ("mt76: mt7921: move mt7921_dma_reset in dma.c")
    Signed-off-by: Quan Zhou <quan.zhou@mediatek.com>
    Signed-off-by: Deren Wu <deren.wu@mediatek.com>
    Signed-off-by: Felix Fietkau <nbd@nbd.name>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ee6718f5179e760a7c4e035fcc7e96f19ea849a1
Author: Neil Chen <yn.chen@mediatek.com>
Date:   Tue Mar 7 19:22:49 2023 +0800

    wifi: mt76: mt7921: use driver flags rather than mac80211 flags to mcu

    [ Upstream commit a5af1481a85679a9812f728ee5cf64ee7edbe669 ]

    FIF_* flags from mac80211 is not ABI. mt7921 should not pass it into mcu
    directly. Remap FIF_* to driver defined flags as mcu command input.

    Fixes: c222f77fd421 ("wifi: mt76: mt7921: fix rx filter incorrect by drv/fw
inconsistent")
    Signed-off-by: Neil Chen <yn.chen@mediatek.com>
    Signed-off-by: Deren Wu <deren.wu@mediatek.com>
    Signed-off-by: Felix Fietkau <nbd@nbd.name>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3f89cf1948dc9b1eb0e2dc243fd78d8988fbb6b2
Author: StanleyYP Wang <StanleyYP.Wang@mediatek.com>
Date:   Fri Mar 3 16:35:58 2023 +0800

    wifi: mt76: mt7996: fix eeprom tx path bitfields

    [ Upstream commit 72fc0df3006ce5c109f9c68f0724e44c47b4ec7b ]

    Swap the tx path bitfields of band1 and band2 to read correct setting.

    Fixes: 98686cd21624 ("wifi: mt76: mt7996: add driver for MediaTek Wi-Fi 7 (802.11be)
devices")
    Signed-off-by: StanleyYP Wang <StanleyYP.Wang@mediatek.com>
    Signed-off-by: Shayne Chen <shayne.chen@mediatek.com>
    Signed-off-by: Felix Fietkau <nbd@nbd.name>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a3d731a413ef65db99130f9a0c6172fe2c8392a9
Author: Peter Chiu <chui-hao.chiu@mediatek.com>
Date:   Fri Mar 3 16:35:55 2023 +0800

    wifi: mt76: mt7996: fix pointer calculation in ie countdown event

    [ Upstream commit 8b14ce24a0297175bc4ebdf26d45a22b5a33847f ]

    Fix the tail and data pointers. The rxd->len in mt7996_mcu_rxd does not
    include the length of general rxd. It only includes the length of
```

```
    firmware event rxd. Use skb->length to get the correct length.

    Fixes: 98686cd21624 ("wifi: mt76: mt7996: add driver for MediaTek Wi-Fi 7 (802.11be)
devices")
    Signed-off-by: Peter Chiu <chui-hao.chiu@mediatek.com>
    Signed-off-by: Shayne Chen <shayne.chen@mediatek.com>
    Signed-off-by: Felix Fietkau <nbd@nbd.name>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a7f0965851f036d4e146ca90cad96a82199be1ba
Author: Shayne Chen <shayne.chen@mediatek.com>
Date:   Fri Mar 3 16:35:53 2023 +0800

    wifi: mt76: mt7996: let non-bufferable MMPDUs use correct hw queue

    [ Upstream commit d0b6f86fdbefa62fd4ad2acd1aea6c45f9b518ba ]

    Pass qid into mt7996_mac_write_txwi() to let the tx descriptor of
    non-bufferable MMPDUs be filled with correct hw queue index.

    Fixes: 98686cd21624 ("wifi: mt76: mt7996: add driver for MediaTek Wi-Fi 7 (802.11be)
devices")
    Signed-off-by: Shayne Chen <shayne.chen@mediatek.com>
    Signed-off-by: Felix Fietkau <nbd@nbd.name>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5d447e3a41a7ad6e1d2a90eafcf770d74694928f
Author: Kang Chen <void0red@gmail.com>
Date:   Mon Feb 27 22:48:23 2023 +0800

    wifi: mt76: handle failure of vzalloc in mt7615_coredump_work

    [ Upstream commit 9e47dd9f64a47ae00ca0123017584c37209ee900 ]

    vzalloc may fails, dump might be null and will cause
    illegal address access later.

    Link: https://lore.kernel.org/all/Y%2Fy5Asxw3T3m4jCw@lore-desk
    Fixes: d2bf7959d9c0 ("mt76: mt7663: introduce coredump support")
    Signed-off-by: Kang Chen <void0red@gmail.com>
    Signed-off-by: Felix Fietkau <nbd@nbd.name>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 39fca8759eaf439719d86025f4609bb6ffac1a5f
Author: Howard Hsu <howard-yh.hsu@mediatek.com>
Date:   Thu Feb 23 14:24:57 2023 +0800

    wifi: mt76: mt7915: rework init flow in mt7915_thermal_init()

    [ Upstream commit 9c97df11dfe68b548a47744ba9fa6beddcfba2bc ]

    If kernel do not enable CONFIG_HWMON, it may cause thermal
    initialization to be done with temperature value 0 and then can not
    transmit. This commit fixes it by setting trigger/restore temperature
    before checking CONFIG_HWMON.

    Fixes: 7d12b38ab6f6 ("wifi: mt76: mt7915: call mt7915_mcu_set_thermal_throttling() only
after init_work")
    Signed-off-by: Howard Hsu <howard-yh.hsu@mediatek.com>
    Signed-off-by: Felix Fietkau <nbd@nbd.name>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 690b437f1008945451653b77bb0bf28ee30779f3
Author: Lorenz Brun <lorenz@brun.one>
Date:   Sat Feb 18 12:29:45 2023 +0100
```

```
    wifi: mt76: mt7915: expose device tree match table

    [ Upstream commit 90fb69212c60e26ef70ed0e8532b116c7649ac88 ]

    On MT7986 the WiFi driver currently does not get automatically loaded,
    requiring manual modprobing because the device tree compatibles are not
    exported into metadata.

    Add the missing MODULE_DEVICE_TABLE macro to fix this.

    Fixes: 99ad32a4ca3a2 ("mt76: mt7915: add support for MT7986")
    Signed-off-by: Lorenz Brun <lorenz@brun.one>
    Reviewed-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Signed-off-by: Felix Fietkau <nbd@nbd.name>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit deffb99a802ff039e422e5e4d5ee81355195b893
Author: Ryder Lee <ryder.lee@mediatek.com>
Date:   Sat Feb 18 01:46:54 2023 +0800

    wifi: mt76: mt7996: fix radiotap bitfield

    [ Upstream commit 63a3724632464559d1e98d90d2c63d43ec6ef6f5 ]

    The 11be generation's radiotap bitfields were wrongly copy-and-pasted
    from 11ax driver, so fix them accordingly.

    Fixes: 98686cd21624 ("wifi: mt76: mt7996: add driver for MediaTek Wi-Fi 7 (802.11be)
devices")
    Signed-off-by: Ryder Lee <ryder.lee@mediatek.com>
    Signed-off-by: Felix Fietkau <nbd@nbd.name>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cc022a673e26afc52a669cee03dee25f068b7143
Author: Dan Carpenter <error27@gmail.com>
Date:   Thu Feb 16 15:15:05 2023 +0300

    wifi: mt76: mt7915: unlock on error in mt7915_thermal_temp_store()

    [ Upstream commit cdc215c2c8d74b3c8886650e979b47f16c1f7f92 ]

    Drop the lock before returning -EINVAL.

    Fixes: ecaccdae7a7e ("wifi: mt76: mt7915: rework mt7915_thermal_temp_store()")
    Signed-off-by: Dan Carpenter <error27@gmail.com>
    Acked-by: Lorenzo Bianconi <lorenzo@kernel.org>
    Signed-off-by: Felix Fietkau <nbd@nbd.name>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 95e41ac30a0d0b3637f0b3c93235a7a107a2fb7f
Author: Deren Wu <deren.wu@mediatek.com>
Date:   Tue Feb 14 10:49:57 2023 +0800

    wifi: mt76: mt7921: fix PCI DMA hang after reboot

    [ Upstream commit 9270270d62191b7549296721e8d5f3dc0df01563 ]

    mt7921 just stop some workers and clean up chip status before reboot.
    In stress test, there are working activities still running at the period
    of .shutdown callback and that would cause some hosts cannot recover
    DMA after reboot. To avoid the floating state in reboot, we use
    mt7921_pci_remove() to fully deinit all resources.

    Fixes: f23a0cea8bd6 ("wifi: mt76: mt7921e: add pci .shutdown() support")
```

```
    Signed-off-by: Deren Wu <deren.wu@mediatek.com>
    Reviewed-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Signed-off-by: Felix Fietkau <nbd@nbd.name>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit fe317f8ec6678d05b482a023bc9af6f472f79d49
Author: Deren Wu <deren.wu@mediatek.com>
Date:   Sat Feb 11 09:01:58 2023 +0800

    wifi: mt76: mt7921: fix wrong command to set STA channel

    [ Upstream commit fcc51acfebb85dbc3ab1bea3ce4997d7c0a3a38d ]

    Should not use AND operator to check vif type NL80211_IFTYPE_MONITOR, and
    that will cause we go into sniffer command for both STA and MONITOR
    mode. However, the sniffer command would set channel properly (with some
    extra options), the STA mode still works even if using the wrong
    command.

    Fix vif type check to make sure we using the right command to update
    channel.

    Fixes: 914189af23b8 ("wifi: mt76: mt7921: fix channel switch fail in monitor mode")
    Signed-off-by: Deren Wu <deren.wu@mediatek.com>
    Signed-off-by: Felix Fietkau <nbd@nbd.name>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 779e206020d913cf0102640a5efb4aad5faa1e07
Author: Deren Wu <deren.wu@mediatek.com>
Date:   Sat Feb 4 14:53:04 2023 +0800

    wifi: mt76: remove redundent MCU_UNI_CMD_* definitions

    [ Upstream commit 532f0482fc577a78c0086893ab39acb406ac3e65 ]

    clear redundent definitions only

    Fixes: 5b55b6da982c ("wifi: mt76: mt7921: add unified ROC cmd/event support")
    Signed-off-by: Deren Wu <deren.wu@mediatek.com>
    Signed-off-by: Felix Fietkau <nbd@nbd.name>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit eefd297b508dfe995cce1e83e104d47d2933c2f2
Author: Emmanuel Grumbach <emmanuel.grumbach@intel.com>
Date:   Sun Apr 16 15:47:38 2023 +0300

    wifi: iwlwifi: make the loop for card preparation effective

    [ Upstream commit 28965ec0b5d9112585f725660e2ff13218505ace ]

    Since we didn't reset t to 0, only the first iteration of the loop
    did checked the ready bit several times.
    From the second iteration and on, we just tested the bit once and
    continued to the next iteration.

    Reported-and-tested-by: Lorenzo Zolfanelli <lorenzo@zolfa.nl>
    Link: https://bugzilla.kernel.org/show_bug.cgi?id=216452
    Fixes: 289e5501c314 ("iwlwifi: fix the preparation of the card")
    Signed-off-by: Emmanuel Grumbach <emmanuel.grumbach@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link:
https://lore.kernel.org/r/20230416154301.615b683ab9c8.Ic52c3229d3345b0064fa34263293db095d88d
af8@changeid
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit a368fa2096a8d254fd5ce24f7a359b1b2248d1e6
Author: Pavel Begunkov <asml.silence@gmail.com>
Date:   Thu Apr 13 15:28:05 2023 +0100

    io_uring/rsrc: use nospec'ed indexes

    [ Upstream commit 953c37e066f05a3dca2d74643574b8dfe8a83983 ]

    We use array_index_nospec() for registered buffer indexes, but don't use
    it while poking into rsrc tags, fix that.

    Fixes: 634d00df5e1cf ("io_uring: add full-fledged dynamic buffers support")
    Signed-off-by: Pavel Begunkov <asml.silence@gmail.com>
    Link:
https://lore.kernel.org/r/f02fafc5a9c0dd69be2b0618c38831c078232ff0.1681395792.git.asml.silen
ce@gmail.com
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 127f0d4bc288baf2c8bd846fc57aa04fa6b708cf
Author: Jan Kara <jack@suse.cz>
Date:   Wed Mar 29 17:49:32 2023 +0200

    jdb2: Don't refuse invalidation of already invalidated buffers

    [ Upstream commit bd159398a2d2234de07d310132865706964aaaa7 ]

    When invalidating buffers under the partial tail page,
    jbd2_journal_invalidate_folio() returns -EBUSY if the buffer is part of
    the committing transaction as we cannot safely modify buffer state.
    However if the buffer is already invalidated (due to previous
    invalidation attempts from ext4_wait_for_tail_page_commit()), there's
    nothing to do and there's no point in returning -EBUSY. This fixes
    occasional warnings from ext4_journalled_invalidate_folio() triggered by
    generic/051 fstest when blocksize < pagesize.

    Fixes: 53e872681fed ("ext4: fix deadlock in journal_unmap_buffer()")
    Signed-off-by: Jan Kara <jack@suse.cz>
    Link: https://lore.kernel.org/r/20230329154950.19720-1-jack@suse.cz
    Signed-off-by: Theodore Ts'o <tytso@mit.edu>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0abe0d3805caff54934367eab7c3fc9def028220
Author: Tom Rix <trix@redhat.com>
Date:   Fri Apr 14 13:11:58 2023 +0300

    wifi: iwlwifi: fw: move memset before early return

    [ Upstream commit 8ce437dd5b2e4adef13aa4ecce07392f9966b1ab ]

    Clang static analysis reports this representative issue
    dbg.c:1455:6: warning: Branch condition evaluates to
    a garbage value
      if (!rxf_data.size)
           ^~~~~~~~~~~~~~

    This check depends on iwl_ini_get_rxf_data() to clear
    rxf_data but the function can return early without
    doing the clear.  So move the memset before the early
    return.

    Fixes: cc9b6012d34b ("iwlwifi: yoyo: use hweight_long instead of bit manipulating")
    Signed-off-by: Tom Rix <trix@redhat.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
```

```
    Link:
https://lore.kernel.org/r/20230414130637.872a7175f1ff.I33802a77a91998276992b088fbe25f61c87c3
3ac@changeid
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 013bf8aca5d76f81be5e9b3289262aaf6daa70a6
Author: Tom Rix <trix@redhat.com>
Date:   Fri Apr 14 13:11:57 2023 +0300

    wifi: iwlwifi: mvm: initialize seq variable

    [ Upstream commit 11e94d2bcd88dea5d9ce99555b6b172f5232d3e2 ]

    Clang static analysis reports this issue
    d3.c:567:22: warning: The left operand of '>' is
      a garbage value
      if (seq.tkip.iv32 > cur_rx_iv32)
          ~~~~~~~~~~~~~ ^

    seq is never initialized. Call ieee80211_get_key_rx_seq() to
    initialize seq.

    Fixes: 0419e5e672d6 ("iwlwifi: mvm: d3: separate TKIP data from key iteration")
    Signed-off-by: Tom Rix <trix@redhat.com>
    Reviewed-by: Nick Desaulniers <ndesaulniers@google.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link:
https://lore.kernel.org/r/20230414130637.6dd372f84f93.If1f708c90e6424a935b4eba3917dfb7582e0d
d0a@changeid
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4ad494f1d7c120ed6d970c4758139cd733f40569
Author: Daniel Gabay <daniel.gabay@intel.com>
Date:   Thu Apr 13 21:40:34 2023 +0300

    wifi: iwlwifi: yoyo: Fix possible division by zero

    [ Upstream commit ba30415118eee374a08b39a0460a1d1e52c24a25 ]

    Don't allow buffer allocation TLV with zero req_size since it
    leads later to division by zero in iwl_dbg_tlv_alloc_fragments().
    Also, NPK/SRAM locations are allowed to have zero buffer req_size,
    don't discard them.

    Fixes: a9248de42464 ("iwlwifi: dbg_ini: add TLV allocation new API support")
    Signed-off-by: Daniel Gabay <daniel.gabay@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link:
https://lore.kernel.org/r/20230413213309.5d6688ed74d8.I5c2f3a882b50698b708d54f4524dc5bdf11e3
d32@changeid
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit db407b0196d623e715f194af49e99295eacd0f3b
Author: Daniel Gabay <daniel.gabay@intel.com>
Date:   Thu Apr 13 21:40:33 2023 +0300

    wifi: iwlwifi: yoyo: skip dump correctly on hw error

    [ Upstream commit 11195ab0d6f3202cf7af1a4c69570f59c377d8ad ]

    When NIC is in a bad state, reading data will return 28 bits as
    0xa5a5a5a and the lowest 4 bits are not fixed value.
```

```
    Mask these bits in a few places to skip the dump correctly.

    Fixes: 89639e06d0f3 ("iwlwifi: yoyo: support for new DBGI_SRAM region")
    Signed-off-by: Daniel Gabay <daniel.gabay@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link:
https://lore.kernel.org/r/20230413213309.df6c0663179d.I36d8487b2419c6fefa65e5514855d94327c3b
1eb@changeid
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4dbeeda1f8303862dc6a27a0da2746032c560000
Author: Ayala Beker <ayala.beker@intel.com>
Date:   Thu Apr 13 21:40:22 2023 +0300

    wifi: iwlwifi: mvm: don't drop unencrypted MCAST frames

    [ Upstream commit 8e5a26360cbe29b896b6758518280d41c3704d43 ]

    MCAST frames are filtered out by the driver if we are not
    authorized yet.
    Fix it to filter out only protected frames.

    Fixes: 147eb05f24e6 ("iwlwifi: mvm: always tell the firmware to accept MCAST frames in
BSS")
    Signed-off-by: Ayala Beker <ayala.beker@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link:
https://lore.kernel.org/r/20230413213309.9cedcc27db60.I8fb7057981392660da482dd215e85c15946d3
f4b@changeid
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3c9a4070ba8ab542ee6555532ba5a52da3bddc68
Author: Yu Kuai <yukuai3@huawei.com>
Date:   Tue Mar 14 09:22:58 2023 +0800

    md/raid10: don't call bio_start_io_acct twice for bio which experienced read error

    [ Upstream commit 7cddb055bfda5f7b0be931e8ea750fc28bc18a27 ]

    handle_read_error() will resumit r10_bio by raid10_read_request(), which
    will call bio_start_io_acct() again, while bio_end_io_acct() will only
    be called once.

    Fix the problem by don't account io again from handle_read_error().

    Fixes: 528bc2cf2fcc ("md/raid10: enable io accounting")
    Suggested-by: Song Liu <song@kernel.org>
    Signed-off-by: Yu Kuai <yukuai3@huawei.com>
    Signed-off-by: Song Liu <song@kernel.org>
    Link: https://lore.kernel.org/r/20230314012258.2395894-1-yukuai1@huaweicloud.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ec473e82e10d39a02eb59b0b95e546119a3bdb79
Author: Yu Kuai <yukuai3@huawei.com>
Date:   Fri Mar 10 15:38:55 2023 +0800

    md/raid10: fix memleak of md thread

    [ Upstream commit f0ddb83da3cbbf8a1f9087a642c448ff52ee9abd ]

    In raid10_run(), if setup_conf() succeed and raid10_run() failed before
    setting 'mddev->thread', then in the error path 'conf->thread' is not
```

```
    freed.

    Fix the problem by setting 'mddev->thread' right after setup_conf().

    Fixes: 43a521238aca ("md-cluster: choose correct label when clustered layout is not
supported")
    Signed-off-by: Yu Kuai <yukuai3@huawei.com>
    Signed-off-by: Song Liu <song@kernel.org>
    Link: https://lore.kernel.org/r/20230310073855.1337560-7-yukuai1@huaweicloud.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e2fec8d95353a48634b085011626ba3ec8ab8b1c
Author: Yu Kuai <yukuai3@huawei.com>
Date:   Fri Mar 10 15:38:54 2023 +0800

    md/raid10: fix memleak for 'conf->bio_split'

    [ Upstream commit c9ac2acde53f5385de185bccf6aaa91cf9ac1541 ]

    In the error path of raid10_run(), 'conf' need be freed, however,
    'conf->bio_split' is missed and memory will be leaked.

    Since there are 3 places to free 'conf', factor out a helper to fix the
    problem.

    Fixes: fc9977dd069e ("md/raid10: simplify the splitting of requests.")
    Signed-off-by: Yu Kuai <yukuai3@huawei.com>
    Signed-off-by: Song Liu <song@kernel.org>
    Link: https://lore.kernel.org/r/20230310073855.1337560-6-yukuai1@huaweicloud.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4f82e7e07cdaf2947d71968e3d6b73370a217093
Author: Yu Kuai <yukuai3@huawei.com>
Date:   Fri Mar 10 15:38:53 2023 +0800

    md/raid10: fix leak of 'r10bio->remaining' for recovery

    [ Upstream commit 26208a7cffd0c7cbf14237ccd20c7270b3ffeb7e ]

    raid10_sync_request() will add 'r10bio->remaining' for both rdev and
    replacement rdev. However, if the read io fails, recovery_request_write()
    returns without issuing the write io, in this case, end_sync_request()
    is only called once and 'remaining' is leaked, cause an io hang.

    Fix the problem by decreasing 'remaining' according to if 'bio' and
    'repl_bio' is valid.

    Fixes: 24afd80d99f8 ("md/raid10: handle recovery of replacement devices.")
    Signed-off-by: Yu Kuai <yukuai3@huawei.com>
    Signed-off-by: Song Liu <song@kernel.org>
    Link: https://lore.kernel.org/r/20230310073855.1337560-5-yukuai1@huaweicloud.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0f96dc405a975b48ad659d0a24748a236bab05e7
Author: Li Nan <linan122@huawei.com>
Date:   Wed Feb 22 12:09:59 2023 +0800

    md/raid10: fix task hung in raid10d

    [ Upstream commit 72c215ed8731c88b2d7e09afc51fffc207ae47b8 ]

    commit fe630de009d0 ("md/raid10: avoid deadlock on recovery.") allowed
    normal io and sync io to exist at the same time. Task hung will occur as
    below:
```

```
      T1                      T2              T3              T4
      raid10d
       handle_read_error
        allow_barrier
         conf->nr_pending--
          -> 0
                              //submit sync io
                              raid10_sync_request
                               raise_barrier
                                ->will not be blocked
                                ...
                              //submit to drivers
        raid10_read_request
         wait_barrier
          conf->nr_pending++
           -> 1
                                                 //retry read fail
                                                 raid10_end_read_request
                                                  reschedule_retry
                                                   add to retry_list
                                                   conf->nr_queued++
                                                    -> 1
                                                                 //sync io fail
                                                                 end_sync_read
                                                                  __end_sync_read
                                                                   reschedule_retry
                                                                    add to retry_list
                                                                     conf->nr_queued++
                                                                      -> 2
       ...
       handle_read_error
       get form retry_list
       conf->nr_queued--
        freeze_array
         wait nr_pending == nr_queued+1
              ->1           ->2
         //task hung
```

retry read and sync io will be added to retry_list(nr_queued->2) if they
fails. raid10d() called handle_read_error() and hung in freeze_array().
nr_queued will not decrease because raid10d is blocked, nr_pending will
not increase because conf->barrier is not released.

Fix it by moving allow_barrier() after raid10_read_request().
raise_barrier() will wait for nr_waiting to become 0. Therefore, sync io
and regular io will not be issued at the same time.

Also remove the check of nr_queued in stop_waiting_barrier. It can be 0
but don't need to be blocking. Remove the check for MD_RECOVERY_RUNNING as
the check is redundent.

Fixes: fe630de009d0 ("md/raid10: avoid deadlock on recovery.")
Signed-off-by: Li Nan <linan122@huawei.com>
Signed-off-by: Song Liu <song@kernel.org>
Link: https://lore.kernel.org/r/20230222041000.3341651-2-linan666@huaweicloud.com
Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9c4af9874c2136399c76e7d1489ed2cb3d03c810
Author: Chao Yu <chao@kernel.org>
Date: Wed Apr 5 22:45:36 2023 +0800

    f2fs: fix to check return value of inc_valid_block_count()

    [Upstream commit 935fc6fa6466cf18dd72dd1518ebc7bfc4cd58a4]

```
    In __replace_atomic_write_block(), we missed to check return value
    of inc_valid_block_count(), for extreme testcase that f2fs image is
    run out of space, it may cause inconsistent status in between SIT
    table and total valid block count.

    Cc: Daeho Jeong <daehojeong@google.com>
    Fixes: 3db1de0e582c ("f2fs: change the current atomic write way")
    Signed-off-by: Chao Yu <chao@kernel.org>
    Signed-off-by: Jaegeuk Kim <jaegeuk@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 988799a264ece63ad10e1319094c2c11c942a061
Author: Chao Yu <chao@kernel.org>
Date:   Wed Apr 5 22:44:53 2023 +0800

    f2fs: fix to check return value of f2fs_do_truncate_blocks()

    [ Upstream commit b851ee6ba3cc212641e622ebcf92b950c7bafa07 ]

    Otherwise, if truncation on cow_inode failed, remained data may
    pollute current transaction of atomic write.

    Cc: Daeho Jeong <daehojeong@google.com>
    Fixes: a46bebd502fe ("f2fs: synchronize atomic write aborts")
    Signed-off-by: Chao Yu <chao@kernel.org>
    Signed-off-by: Jaegeuk Kim <jaegeuk@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ca3373087f333ef7795a9905b079b3fe7df2c3ba
Author: Daniel Borkmann <daniel@iogearbox.net>
Date:   Thu Apr 13 20:28:42 2023 +0200

    bpf, sockmap: Revert buggy deadlock fix in the sockhash and sockmap

    [ Upstream commit 8c5c2a4898e3d6bad86e29d471e023c8a19ba799 ]

    syzbot reported a splat and bisected it to recent commit ed17aa92dc56 ("bpf,
    sockmap: fix deadlocks in the sockhash and sockmap"):

      [...]
      WARNING: CPU: 1 PID: 9280 at kernel/softirq.c:376 __local_bh_enable_ip+0xbe/0x130
kernel/softirq.c:376
      Modules linked in:
      CPU: 1 PID: 9280 Comm: syz-executor.1 Not tainted 6.2.0-syzkaller-13249-gd319f344561d
#0
      Hardware name: Google Google Compute Engine/Google Compute Engine, BIOS Google
03/30/2023
      RIP: 0010:__local_bh_enable_ip+0xbe/0x130 kernel/softirq.c:376
      [...]
      Call Trace:
      <TASK>
      spin_unlock_bh include/linux/spinlock.h:395 [inline]
      sock_map_del_link+0x2ea/0x510 net/core/sock_map.c:165
      sock_map_unref+0xb0/0x1d0 net/core/sock_map.c:184
      sock_hash_delete_elem+0x1ec/0x2a0 net/core/sock_map.c:945
      map_delete_elem kernel/bpf/syscall.c:1536 [inline]
      __sys_bpf+0x2edc/0x53e0 kernel/bpf/syscall.c:5053
      __do_sys_bpf kernel/bpf/syscall.c:5166 [inline]
      __se_sys_bpf kernel/bpf/syscall.c:5164 [inline]
      __x64_sys_bpf+0x79/0xc0 kernel/bpf/syscall.c:5164
      do_syscall_x64 arch/x86/entry/common.c:50 [inline]
      do_syscall_64+0x39/0xb0 arch/x86/entry/common.c:80
      entry_SYSCALL_64_after_hwframe+0x63/0xcd
      RIP: 0033:0x7fe8f7c8c169
      </TASK>
```

```
       [...]

    Revert for now until we have a proper solution.

    Fixes: ed17aa92dc56 ("bpf, sockmap: fix deadlocks in the sockhash and sockmap")
    Reported-by: syzbot+49f6cef45247ff249498@syzkaller.appspotmail.com
    Cc: Hsin-Wei Hung <hsinweih@uci.edu>
    Cc: Xin Liu <liuxin350@huawei.com>
    Cc: John Fastabend <john.fastabend@gmail.com>
    Signed-off-by: Daniel Borkmann <daniel@iogearbox.net>
    Link: https://lore.kernel.org/bpf/000000000000f1db9605f939720e@google.com/
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b1ddc42127ebfc769da2198604a8c294c62a2782
Author: Avraham Stern <avraham.stern@intel.com>
Date:   Thu Apr 13 10:44:15 2023 +0300

    wifi: iwlwifi: mvm: don't set CHECKSUM_COMPLETE for unsupported protocols

    [ Upstream commit 217f3c52f00d3419ecdd38a99a7eceecb11679b2 ]

    On Bz devices, CHECKSUM_COMPLETE was set for unsupported protocols
    which results in a warning. Fix it.

    Fixes: b6f5b647f694 ("iwlwifi: mvm: handle RX checksum on Bz devices")
    Signed-off-by: Avraham Stern <avraham.stern@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link:
https://lore.kernel.org/r/20230413102635.a2a35286f0ca.I50daa9445a6465514c44f5096c32adef64beb
a5f@changeid
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 754fe7cae5682594c004cba9963a25970df5e0dc
Author: Avraham Stern <avraham.stern@intel.com>
Date:   Thu Apr 13 10:44:14 2023 +0300

    wifi: iwlwifi: trans: don't trigger d3 interrupt twice

    [ Upstream commit 277f56a141fc54ef7f9e09dba65fb2e12021411d ]

    When the IPC registers are used for sleep control, setting
    the IPC sleep bit already triggers an interrupt to the fw, so
    there is no need to also set the doorbell. Setting also the
    doorbell triggers the sleep interrupt twice which lead to
    an assert.

    Fixes: af08571d3925 ("iwlwifi: pcie: support Bz suspend/resume trigger")
    Signed-off-by: Avraham Stern <avraham.stern@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link:
https://lore.kernel.org/r/20230413102635.b5f2f6e44d38.I4cb5b6ad4914db47a714e731c4c8b4db679ca
bce@changeid
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3fdaefcd3ad5c895c56df98cd5a04598223d4088
Author: Johannes Berg <johannes.berg@intel.com>
Date:   Thu Apr 13 10:44:08 2023 +0300

    wifi: iwlwifi: mvm: fix A-MSDU checks

    [ Upstream commit d34d11aea2d5d71a66f2f90507c3f7202b964175 ]

    Since Gl A-step devices use the old checksum hardware,
```

```
    we shouldn't use the Bz code to check for A-MSDU
    combining ability; fix that.

    Fixes: ec18e7d4d20d ("wifi: iwlwifi: mvm: use old checksum for Bz A-step")
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link:
https://lore.kernel.org/r/20230413102635.8c445b943fee.Ibf772102ca712f59e2ee0cdd4c344011fcf44
5aa@changeid
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cd154cda8b4a7a6bcad565b52f81927b08c06470
Author: Johannes Berg <johannes.berg@intel.com>
Date:   Thu Apr 13 10:44:01 2023 +0300

    wifi: iwlwifi: debug: fix crash in __iwl_err()

    [ Upstream commit 634c7b1bd08ca322537ab389f8cbd7bb543b5e45 ]

    In __iwl_err(), if we rate-limit the message away, then
    vaf.va is still NULL-initialized by the time we get to
    the tracing code, which then crashes. When it doesn't
    get rate-limited out, it's still wrong to reuse the old
    args2 that was already printed, which is why we bother
    making a copy in the first place.

    Assign vaf.va properly to fix this.

    Fixes: e5f1cc98cc1b ("iwlwifi: allow rate-limited error messages")
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link:
https://lore.kernel.org/r/20230413102635.e27134c6bcd4.Ib3894cd2ba7a5ad5e75912a7634f146ceaa56
9e2@changeid
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 95432ccc853e5087ded9c0605eee0fa44bc7f3ff
Author: Christoph Hellwig <hch@lst.de>
Date:   Thu Apr 13 08:40:38 2023 +0200

    blk-mq: don't plug for head insertions in blk_execute_rq_nowait

    [ Upstream commit 50947d7fe9fa6abe3ddc40769dfb02a51c58edb6 ]

    Plugs never insert at head, so don't plug for head insertions.

    Fixes: 1c2d2fff6dc0 ("block: wire-up support for passthrough plugging")
    Signed-off-by: Christoph Hellwig <hch@lst.de>
    Reviewed-by: Bart Van Assche <bvanassche@acm.org>
    Reviewed-by: Damien Le Moal <dlemoal@kernel.org>
    Link: https://lore.kernel.org/r/20230413064057.707578-2-hch@lst.de
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7cb30dd356ab8b3e6a2f7cf0f3ac29bf884fd8dc
Author: Song Liu <song@kernel.org>
Date:   Wed Apr 12 14:04:22 2023 -0700

    selftests/bpf: Fix leaked bpf_link in get_stackid_cannot_attach

    [ Upstream commit c1e07a80cf23d3a6e96172bc9a73bfa912a9fcbc ]

    skel->links.oncpu is leaked in one case. This causes test perf_branches
```

```
    fails when it runs after get_stackid_cannot_attach:

    ./test_progs -t get_stackid_cannot_attach,perf_branches
    84      get_stackid_cannot_attach:OK
    test_perf_branches_common:PASS:test_perf_branches_load 0 nsec
    test_perf_branches_common:PASS:attach_perf_event 0 nsec
    test_perf_branches_common:PASS:set_affinity 0 nsec
    check_good_sample:FAIL:output not valid no valid sample from prog
    146/1   perf_branches/perf_branches_hw:FAIL
    146/2   perf_branches/perf_branches_no_hw:OK
    146     perf_branches:FAIL

    All error logs:
    test_perf_branches_common:PASS:test_perf_branches_load 0 nsec
    test_perf_branches_common:PASS:attach_perf_event 0 nsec
    test_perf_branches_common:PASS:set_affinity 0 nsec
    check_good_sample:FAIL:output not valid no valid sample from prog
    146/1   perf_branches/perf_branches_hw:FAIL
    146     perf_branches:FAIL
    Summary: 1/1 PASSED, 0 SKIPPED, 1 FAILED

    Fix this by adding the missing bpf_link__destroy().

    Fixes: 346938e9380c ("selftests/bpf: Add get_stackid_cannot_attach")
    Signed-off-by: Song Liu <song@kernel.org>
    Signed-off-by: Daniel Borkmann <daniel@iogearbox.net>
    Link: https://lore.kernel.org/bpf/20230412210423.900851-3-song@kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c7bc957b7a11c5863b49f888e5e171ad641f545f
Author: Song Liu <song@kernel.org>
Date:   Wed Apr 12 14:04:21 2023 -0700

    selftests/bpf: Use read_perf_max_sample_freq() in perf_event_stackmap

    [ Upstream commit de6d014a09bf12a9a8959d60c0a1d4a41d394a89 ]

    Currently, perf_event sample period in perf_event_stackmap is set too low
    that the test fails randomly. Fix this by using the max sample frequency,
    from read_perf_max_sample_freq().

    Move read_perf_max_sample_freq() to testing_helpers.c. Replace the CHECK()
    with if-printf, as CHECK is not available in testing_helpers.c.

    Fixes: 1da4864c2b20 ("selftests/bpf: Add callchain_stackid")
    Signed-off-by: Song Liu <song@kernel.org>
    Signed-off-by: Daniel Borkmann <daniel@iogearbox.net>
    Link: https://lore.kernel.org/bpf/20230412210423.900851-2-song@kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e870bcf4ec4e36d6f198fd06f2e013c0c046e86d
Author: Ming Lei <ming.lei@redhat.com>
Date:   Wed Apr 12 16:49:04 2023 +0800

    nvme-fcloop: fix "inconsistent {IN-HARDIRQ-W} -> {HARDIRQ-ON-W} usage"

    [ Upstream commit 4f86a6ff6fbd891232dda3ca97fd1b9630b59809 ]

    fcloop_fcp_op() could be called from flush request's ->end_io(flush_end_io) in
    which the spinlock of fq->mq_flush_lock is grabbed with irq saved/disabled.

    So fcloop_fcp_op() can't call spin_unlock_irq(&tfcp_req->reqlock) simply
    which enables irq unconditionally.

    Fixes the warning by switching to spin_lock_irqsave()/spin_unlock_irqrestore()
```

```
    Fixes: c38dbbfab1bc ("nvme-fcloop: fix inconsistent lock state warnings")
    Reported-by: Yi Zhang <yi.zhang@redhat.com>
    Signed-off-by: Ming Lei <ming.lei@redhat.com>
    Reviewed-by: Ewan D. Milne <emilne@redhat.com>
    Tested-by: Yi Zhang <yi.zhang@redhat.com>
    Signed-off-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b3cd4ca963ecaa2e5aca5e25a8aaaeac51029edf
Author: Keith Busch <kbusch@kernel.org>
Date:   Wed Apr 5 14:57:20 2023 -0700

    nvme: fix async event trace event

    [ Upstream commit 6622b76fe922b94189499a90ccdb714a4a8d0773 ]

    Mixing AER Event Type and Event Info has masking clashes. Just print the
    event type, but also include the event info of the AER result in the
    trace.

    Fixes: 09bd1ff4b15143b ("nvme-core: add async event trace helper")
    Reported-by: Nate Thornton <nate.thornton@samsung.com>
    Reviewed-by: Sagi Grimberg <sagi@grimberg.me>
    Reviewed-by: Minwoo Im <minwoo.im@samsung.com>
    Reviewed-by: Chaitanya Kulkarni <kch@nvidia.com>
    Signed-off-by: Keith Busch <kbusch@kernel.org>
    Signed-off-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8fea8b33923b3adc9b4ad25a475b2d9d926444f0
Author: Damien Le Moal <damien.lemoal@opensource.wdc.com>
Date:   Wed Mar 15 19:59:38 2023 +0900

    nvmet: fix I/O Command Set specific Identify Controller

    [ Upstream commit a5a6ab0950b46ab1ef4a5c83c80234018b81b38a ]

    For an identify command with cns set to NVME_ID_CNS_CS_CTRL, the NVMe
    2.0 specification states that:

    If the I/O Command Set specified by the CSI field does not have an
    Identify Controller data structure, then the controller shall return
    a zero filled data structure. If the host requests a data structure for
    an I/O Command Set that the controller does not support, the controller
    shall abort the command with a status code of Invalid Field in Command.

    However, the current implementation of this identify command in
    nvmet_execute_identify() only handles the ZNS command set, returning an
    error for the NVM command set, which is not compliant with the
    specifications as we do support this command set.

    Fix this by:
    1) Renaming nvmet_execute_identify_cns_cs_ctrl() to
       nvmet_execute_identify_ctrl_zns() to continue handling the
       ZNS command set as is.
    2) Introduce a nvmet_execute_identify_ctrl_ns() helper to handle the
       NVM command set, returning a zero filled nvme_id_ctrl_nvm data
       structure.
    3) Modify nvmet_execute_identify() to call these helpers based on
       the csi specified, returning an error for unsupported command sets.

    Fixes: aaf2e048af27 ("nvmet: add ZBD over ZNS backend support")
    Signed-off-by: Damien Le Moal <damien.lemoal@opensource.wdc.com>
    Reviewed-by: Chaitanya Kulkarni <kch@nvidia.com>
```

```
    Tested-by: Chaitanya Kulkarni <kch@nvidia.com>
    Signed-off-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9fceb5718e12ad3533e0e07a0b6f70d50992bc21
Author: Damien Le Moal <damien.lemoal@opensource.wdc.com>
Date:   Wed Mar 15 19:59:37 2023 +0900

    nvmet: fix Identify Active Namespace ID list handling

    [ Upstream commit 97416f67d55fb8b866ff1815ca7ef26b6dfa6a5e ]

    The identify command with cns set to NVME_ID_CNS_NS_ACTIVE_LIST does
    not depend on the command set. The execution of this command should
    thus not look at the csi field specified in the command. Simplify
    nvmet_execute_identify() to directly call
    nvmet_execute_identify_nslist() without the csi switch-case.

    Fixes: ab5d0b38c047 ("nvmet: add Command Set Identifier support")
    Signed-off-by: Damien Le Moal <damien.lemoal@opensource.wdc.com>
    Reviewed-by: Chaitanya Kulkarni <kch@nvidia.com>
    Tested-by: Chaitanya Kulkarni <kch@nvidia.com>
    Signed-off-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8247cae4f9343c379ff882f1f6d405b5d1ad9e5a
Author: Damien Le Moal <damien.lemoal@opensource.wdc.com>
Date:   Wed Mar 15 19:59:36 2023 +0900

    nvmet: fix Identify Controller handling

    [ Upstream commit 62904b3b333e7f3c0f879dc3513295eee5765c9f ]

    The identify command with cns set to NVME_ID_CNS_CTRL does not depend on
    the command set. The execution of this command should thus not look at
    the csi specified in the command. Simplify nvmet_execute_identify() to
    directly call nvmet_execute_identify_ctrl() without the csi switch-case.

    Fixes: ab5d0b38c047 ("nvmet: add Command Set Identifier support")
    Signed-off-by: Damien Le Moal <damien.lemoal@opensource.wdc.com>
    Reviewed-by: Chaitanya Kulkarni <kch@nvidia.com>
    Tested-by: Chaitanya Kulkarni <kch@nvidia.com>
    Signed-off-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit af9b701667d1ae11138c8e9aa82a10a2694eac09
Author: Damien Le Moal <damien.lemoal@opensource.wdc.com>
Date:   Wed Mar 15 19:59:35 2023 +0900

    nvmet: fix Identify Namespace handling

    [ Upstream commit 8c098aa00118c35108f0c19bd3cdc45e11574948 ]

    The identify command with cns set to NVME_ID_CNS_NS does not directly
    depend on the command set. The NVMe specifications is rather confusing
    here as it appears that this command only applies to the NVM command
    set. However, footnote 8 of Figure 273 in the NVMe 2.0 base
    specifications clearly state that this command applies to NVM command
    sets that support logical blocks, that is, NVM and ZNS. Both the NVM and
    ZNS command set specifications also list this identify as mandatory.

    The command handling should thus not look at the csi field since it is
    defined as unused for this command. Given that we do not support the
    KV command set, simply remove the csi switch-case for that command
    handling and call directly nvmet_execute_identify_ns() in
```

```
    nvmet_execute_identify().

    Fixes: ab5d0b38c047 ("nvmet: add Command Set Identifier support")
    Signed-off-by: Damien Le Moal <damien.lemoal@opensource.wdc.com>
    Reviewed-by: Chaitanya Kulkarni <kch@nvidia.com>
    Tested-by: Chaitanya Kulkarni <kch@nvidia.com>
    Signed-off-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c60c11a0ad0f94b6d458af3a267d4827a1a30a36
Author: Damien Le Moal <damien.lemoal@opensource.wdc.com>
Date:   Tue Mar 14 15:20:36 2023 +0900

    nvmet: fix error handling in nvmet_execute_identify_cns_cs_ns()

    [ Upstream commit ab76e7206b672b2e8818cb121a04506956d6b223 ]

    Nvme specifications state that:

    If the I/O Command Set associated with the namespace identified by the
    NSID field does not support the Identify Namespace data structure
    specified by the CSI field, the controller shall abort the command with
    a status code of Invalid Field in Command.

    In other words, if nvmet_execute_identify_cns_cs_ns() is called for a
    target with a block device that is not zoned, we should not return any
    data and set the status to NVME_SC_INVALID_FIELD.

    While at it, it is also better to revalidate the ns block devie *before*
    checking if the block device is zoned, to ensure that
    nvmet_execute_identify_cns_cs_ns() operates against updated device
    characteristics.

    Fixes: aaf2e048af27 ("nvmet: add ZBD over ZNS backend support")
    Signed-off-by: Damien Le Moal <damien.lemoal@opensource.wdc.com>
    Reviewed-by: Keith Busch <kbusch@kernel.org>
    Reviewed-by: Sagi Grimberg <sagi@grimberg.me>
    Reviewed-by: Chaitanya Kulkarni <kch@nvidia.com>
    Signed-off-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d2b8cf384c39ddb6fdb9e9034cd9b9887e09e735
Author: Xin Liu <liuxin350@huawei.com>
Date:   Thu Apr 6 20:26:22 2023 +0800

    bpf, sockmap: fix deadlocks in the sockhash and sockmap

    [ Upstream commit ed17aa92dc56b6d8883e4b7a8f1c6fbf5ed6cd29 ]

    When huang uses sched_switch tracepoint, the tracepoint
    does only one thing in the mounted ebpf program, which
    deletes the fixed elements in sockhash ([0])

    It seems that elements in sockhash are rarely actively
    deleted by users or ebpf program. Therefore, we do not
    pay much attention to their deletion. Compared with hash
    maps, sockhash only provides spin_lock_bh protection.
    This causes it to appear to have self-locking behavior
    in the interrupt context.


[0]:https://lore.kernel.org/all/CABcoxUayum5oOqFMMqAeWuS8+EzojquSOSyDA3J_2omY=2EeAg@mail.gma
il.com/

    Reported-by: Hsin-Wei Hung <hsinweih@uci.edu>
```

```
    Fixes: 604326b41a6f ("bpf, sockmap: convert to generic sk_msg interface")
    Signed-off-by: Xin Liu <liuxin350@huawei.com>
    Acked-by: John Fastabend <john.fastabend@gmail.com>
    Link: https://lore.kernel.org/r/20230406122622.109978-1-liuxin350@huawei.com
    Signed-off-by: Alexei Starovoitov <ast@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a0f305537a5fba27ccee720966ec7b2fe5fdf230
Author: P Praneesh <quic_ppranees@quicinc.com>
Date:   Fri Mar 24 16:57:01 2023 +0200

    wifi: ath11k: fix writing to unintended memory region

    [ Upstream commit 756a7f90878f0866fd2fe167ef37e90b47326b96 ]

    While initializing spectral, the magic value is getting written to the
    invalid memory address leading to random boot-up crash. This occurs
    due to the incorrect index increment in ath11k_dbring_fill_magic_value
    function. Fix it by replacing the existing logic with memset32 to ensure
    there is no invalid memory access.

    Tested-on: QCN9074 hw1.0 PCI WLAN.HK.2.4.0.1-01838-QCAHKSWPL_SILICONZ-1

    Fixes: d3d358efc553 ("ath11k: add spectral/CFR buffer validation support")
    Signed-off-by: P Praneesh <quic_ppranees@quicinc.com>
    Signed-off-by: Kalle Valo <quic_kvalo@quicinc.com>
    Link: https://lore.kernel.org/r/20230321052900.16895-1-quic_ppranees@quicinc.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ca5adb0becc56def0553dc652548c8cbd290ad84
Author: Sebastian Reichel <sebastian.reichel@collabora.com>
Date:   Fri Apr 7 18:11:29 2023 +0200

    net: ethernet: stmmac: dwmac-rk: fix optional phy regulator handling

    [ Upstream commit db21973263f8c56750cb610f1d5e8bee00a513b9 ]

    The usual devm_regulator_get() call already handles "optional"
    regulators by returning a valid dummy and printing a warning
    that the dummy regulator should be described properly. This
    code open coded the same behaviour, but masked any errors that
    are not -EPROBE_DEFER and is quite noisy.

    This change effectively unmasks and propagates regulators errors
    not involving -ENODEV, downgrades the error print to warning level
    if no regulator is specified and captures the probe defer message
    for /sys/kernel/debug/devices_deferred.

    Fixes: 2e12f536635f ("net: stmmac: dwmac-rk: Use standard devicetree property for phy
regulator")
    Signed-off-by: Sebastian Reichel <sebastian.reichel@collabora.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 548a0fd1320dafd0fe99d17b871d0c14883acbe7
Author: Sebastian Reichel <sebastian.reichel@collabora.com>
Date:   Fri Apr 7 18:11:28 2023 +0200

    net: ethernet: stmmac: dwmac-rk: rework optional clock handling

    [ Upstream commit ea449f7fa0bf3fcd02e04a770b9ff707bf5e8f96 ]

    The clock requesting code is quite repetitive. Fix this by requesting
    the clocks via devm_clk_bulk_get_optional. The optional variant has been
    used, since this is effectively what the old code did. The exact clocks
```

```
    required depend on the platform and configuration. As a side effect
    this change adds correct -EPROBE_DEFER handling.

    Suggested-by: Jakub Kicinski <kuba@kernel.org>
    Suggested-by: Andrew Lunn <andrew@lunn.ch>
    Fixes: 7ad269ea1a2b ("GMAC: add driver for Rockchip RK3288 SoCs integrated GMAC")
    Signed-off-by: Sebastian Reichel <sebastian.reichel@collabora.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7e5a54d1f00725a739dcd20f616d82eff4f764bd
Author: Shuchang Li <lishuchang@hust.edu.cn>
Date:   Tue Apr 4 15:21:32 2023 +0800

    scsi: lpfc: Fix ioremap issues in lpfc_sli4_pci_mem_setup()

    [ Upstream commit 91a0c0c1413239d0548b5aac4c82f38f6d53a91e ]

    When if_type equals zero and pci_resource_start(pdev, PCI_64BIT_BAR4)
    returns false, drbl_regs_memmap_p is not remapped. This passes a NULL
    pointer to iounmap(), which can trigger a WARN() on certain arches.

    When if_type equals six and pci_resource_start(pdev, PCI_64BIT_BAR4)
    returns true, drbl_regs_memmap_p may has been remapped and
    ctrl_regs_memmap_p is not remapped. This is a resource leak and passes a
    NULL pointer to iounmap().

    To fix these issues, we need to add null checks before iounmap(), and
    change some goto labels.

    Fixes: 1351e69fc6db ("scsi: lpfc: Add push-to-adapter support to sli4")
    Signed-off-by: Shuchang Li <lishuchang@hust.edu.cn>
    Link: https://lore.kernel.org/r/20230404072133.1022-1-lishuchang@hust.edu.cn
    Reviewed-by: Justin Tee <justin.tee@broadcom.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 051cb7751cad4547b7ff00a46c9667a1894aed5f
Author: Feng Zhou <zhoufeng.zf@bytedance.com>
Date:   Mon Apr 10 16:59:07 2023 +0800

    bpf/btf: Fix is_int_ptr()

    [ Upstream commit 91f2dc6838c19342f7f2993627c622835cc24890 ]

    When tracing a kernel function with arg type is u32*, btf_ctx_access()
    would report error: arg2 type INT is not a struct.

    The commit bb6728d75611 ("bpf: Allow access to int pointer arguments
    in tracing programs") added support for int pointer, but did not skip
    modifiers before checking it's type. This patch fixes it.

    Fixes: bb6728d75611 ("bpf: Allow access to int pointer arguments in tracing programs")
    Co-developed-by: Chengming Zhou <zhouchengming@bytedance.com>
    Signed-off-by: Chengming Zhou <zhouchengming@bytedance.com>
    Signed-off-by: Feng Zhou <zhoufeng.zf@bytedance.com>
    Signed-off-by: Daniel Borkmann <daniel@iogearbox.net>
    Acked-by: Jiri Olsa <jolsa@kernel.org>
    Link: https://lore.kernel.org/bpf/20230410085908.98493-2-zhoufeng.zf@bytedance.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d03333ce0ef076d804b98ece5ead497e7e040d3a
Author: Gregory Greenman <gregory.greenman@intel.com>
Date:   Mon Apr 10 17:07:21 2023 +0300
```

```
    wifi: iwlwifi: fix duplicate entry in iwl_dev_info_table

    [ Upstream commit fc3c2f0ed86b65dff4b6844c59c597b977cae533 ]

    There're two identical entries for ax1650 device in
    iwl_dev_info_table. Remove one of the duplicate entries.

    Fixes: 953e66a7238b ("iwlwifi: add new ax1650 killer device")
    Signed-off-by: Gregory Greenman <gregory.greenman@intel.com>
    Link: https://lore.kernel.org/r/20230410140721.897683-2-gregory.greenman@intel.com
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit af4ce124d7bd74cb839bbdaccffbb416771a56b5
Author: Chao Yu <chao@kernel.org>
Date:   Mon Apr 10 10:14:02 2023 +0800

    f2fs: fix to avoid use-after-free for cached IPU bio

    [ Upstream commit 5cdb422c839134273866208dad5360835ddb9794 ]

    xfstest generic/019 reports a bug:

    kernel BUG at mm/filemap.c:1619!
    RIP: 0010:folio_end_writeback+0x8a/0x90
    Call Trace:
     end_page_writeback+0x1c/0x60
     f2fs_write_end_io+0x199/0x420
     bio_endio+0x104/0x180
     submit_bio_noacct+0xa5/0x510
     submit_bio+0x48/0x80
     f2fs_submit_write_bio+0x35/0x300
     f2fs_submit_merged_ipu_write+0x2a0/0x2b0
     f2fs_write_single_data_page+0x838/0x8b0
     f2fs_write_cache_pages+0x379/0xa30
     f2fs_write_data_pages+0x30c/0x340
     do_writepages+0xd8/0x1b0
     __writeback_single_inode+0x44/0x370
     writeback_sb_inodes+0x233/0x4d0
     __writeback_inodes_wb+0x56/0xf0
     wb_writeback+0x1dd/0x2d0
     wb_workfn+0x367/0x4a0
     process_one_work+0x21d/0x430
     worker_thread+0x4e/0x3c0
     kthread+0x103/0x130
     ret_from_fork+0x2c/0x50

    The root cause is: after cp_error is set, f2fs_submit_merged_ipu_write()
    in f2fs_write_single_data_page() tries to flush IPU bio in cache, however
    f2fs_submit_merged_ipu_write() missed to check validity of @bio parameter,
    result in submitting random cached bio which belong to other IO context,
    then it will cause use-after-free issue, fix it by adding additional
    validity check.

    Fixes: 0b20fcec8651 ("f2fs: cache global IPU bio")
    Signed-off-by: Chao Yu <chao@kernel.org>
    Signed-off-by: Jaegeuk Kim <jaegeuk@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b8f83d5ae1568049121e89cc30bfbf0c00e35f60
Author: Kal Conley <kal.conley@dectris.com>
Date:   Thu Apr 6 01:59:18 2023 +0200

    xsk: Fix unaligned descriptor validation
```

```
    [ Upstream commit d769ccaf957fe7391f357c0a923de71f594b8a2b ]

    Make sure unaligned descriptors that straddle the end of the UMEM are
    considered invalid. Currently, descriptor validation is broken for
    zero-copy mode which only checks descriptors at page granularity.
    For example, descriptors in zero-copy mode that overrun the end of the
    UMEM but not a page boundary are (incorrectly) considered valid. The
    UMEM boundary check needs to happen before the page boundary and
    contiguity checks in xp_desc_crosses_non_contig_pg(). Do this check in
    xp_unaligned_validate_desc() instead like xp_check_unaligned() already
    does.

    Fixes: 2b43470add8c ("xsk: Introduce AF_XDP buffer allocation API")
    Signed-off-by: Kal Conley <kal.conley@dectris.com>
    Acked-by: Magnus Karlsson <magnus.karlsson@intel.com>
    Link: https://lore.kernel.org/r/20230405235920.7305-2-kal.conley@dectris.com
    Signed-off-by: Martin KaFai Lau <martin.lau@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 580e9deb87774ece4c515c508879395f2610db9c
Author: Herbert Xu <herbert@gondor.apana.org.au>
Date:   Tue Mar 28 11:35:23 2023 +0800

    crypto: drbg - Only fail when jent is unavailable in FIPS mode

    [ Upstream commit 686cd976b6ddedeeb1a1fb09ba53a891d3cc9a03 ]

    When jent initialisation fails for any reason other than ENOENT,
    the entire drbg fails to initialise, even when we're not in FIPS
    mode.  This is wrong because we can still use the kernel RNG when
    we're not in FIPS mode.

    Change it so that it only fails when we are in FIPS mode.

    Fixes: 57225e679788 ("crypto: drbg - Use callback API for random readiness")
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Reviewed-by: Stephan Mueller <smueller@chronox.de>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9ae978209c1cb53c8976f30b2b751e922cbf9345
Author: Quentin Monnet <quentin@isovalent.com>
Date:   Wed Apr 5 14:21:15 2023 +0100

    bpftool: Fix bug for long instructions in program CFG dumps

    [ Upstream commit 67cf52cdb6c8fa6365d29106555dacf95c9fd374 ]

    When dumping the control flow graphs for programs using the 16-byte long
    load instruction, we need to skip the second part of this instruction
    when looking for the next instruction to process. Otherwise, we end up
    printing "BUG_ld_00" from the kernel disassembler in the CFG.

    Fixes: efcef17a6d65 ("tools: bpftool: generate .dot graph from CFG information")
    Signed-off-by: Quentin Monnet <quentin@isovalent.com>
    Link: https://lore.kernel.org/r/20230405132120.59886-3-quentin@isovalent.com
    Signed-off-by: Alexei Starovoitov <ast@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5523c148658dc4acda0d9a8e52cd1c071fe77cad
Author: YiFei Zhu <zhuyifei@google.com>
Date:   Wed Apr 5 19:33:54 2023 +0000

    selftests/bpf: Wait for receive in cg_storage_multi test
```

```
    [ Upstream commit 5af607a861d43ffff830fc1890033e579ec44799 ]

    In some cases the loopback latency might be large enough, causing
    the assertion on invocations to be run before ingress prog getting
    executed. The assertion would fail and the test would flake.

    This can be reliably reproduced by arbitrarily increasing the
    loopback latency (thanks to [1]):
      tc qdisc add dev lo root handle 1: htb default 12
      tc class add dev lo parent 1:1 classid 1:12 htb rate 20kbps ceil 20kbps
      tc qdisc add dev lo parent 1:12 netem delay 100ms

    Fix this by waiting on the receive end, instead of instantly
    returning to the assert. The call to read() will wait for the
    default SO_RCVTIMEO timeout of 3 seconds provided by
    start_server().

    [1] https://gist.github.com/kstevens715/4598301

    Reported-by: Martin KaFai Lau <martin.lau@linux.dev>
    Link: https://lore.kernel.org/bpf/9c5c8b7e-1d89-a3af-5400-14fde81f4429@linux.dev/
    Fixes: 3573f384014f ("selftests/bpf: Test CGROUP_STORAGE behavior on shared egress +
ingress")
    Acked-by: Stanislav Fomichev <sdf@google.com>
    Signed-off-by: YiFei Zhu <zhuyifei@google.com>
    Link: https://lore.kernel.org/r/20230405193354.1956209-1-zhuyifei@google.com
    Signed-off-by: Martin KaFai Lau <martin.lau@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4aae4d8ef374461b0ab6f96c3d520430c88d7d12
Author: Kal Conley <kal.conley@dectris.com>
Date:   Mon Apr 3 14:03:59 2023 +0200

    selftests: xsk: Deflakify STATS_RX_DROPPED test

    [ Upstream commit 68e7322142f5e731af222892d384d311835db0f1 ]

    Fix flaky STATS_RX_DROPPED test. The receiver calls getsockopt after
    receiving the last (valid) packet which is not the final packet sent in
    the test (valid and invalid packets are sent in alternating fashion with
    the final packet being invalid). Since the last packet may or may not
    have been dropped already, both outcomes must be allowed.

    This issue could also be fixed by making sure the last packet sent is
    valid. This alternative is left as an exercise to the reader (or the
    benevolent maintainers of this file).

    This problem was quite visible on certain setups. On one machine this
    failure was observed 50% of the time.

    Also, remove a redundant assignment of pkt_stream->nb_pkts. This field
    is already initialized by __pkt_stream_alloc.

    Fixes: 27e934bec35b ("selftests: xsk: make stat tests not spin on getsockopt")
    Signed-off-by: Kal Conley <kal.conley@dectris.com>
    Acked-by: Magnus Karlsson <magnus.karlsson@intel.com>
    Link: https://lore.kernel.org/r/20230403120400.31018-1-kal.conley@dectris.com
    Signed-off-by: Martin KaFai Lau <martin.lau@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d175e8d41e6c924521c9f1501a907261b8ae6751
Author: Kal Conley <kal.conley@dectris.com>
Date:   Wed Apr 5 10:29:04 2023 +0200

    selftests: xsk: Disable IPv6 on VETH1
```

```
    [ Upstream commit f2b50f17268390567bc0e95642170d88f336c8f4 ]

    This change fixes flakiness in the BIDIRECTIONAL test:

        # [is_pkt_valid] expected length [60], got length [90]
        not ok 1 FAIL: SKB BUSY-POLL BIDIRECTIONAL

    When IPv6 is enabled, the interface will periodically send MLDv1 and
    MLDv2 packets. These packets can cause the BIDIRECTIONAL test to fail
    since it uses VETH0 for RX.

    For other tests, this was not a problem since they only receive on VETH1
    and IPv6 was already disabled on VETH0.

    Fixes: a89052572ebb ("selftests/bpf: Xsk selftests framework")
    Signed-off-by: Kal Conley <kal.conley@dectris.com>
    Link: https://lore.kernel.org/r/20230405082905.6303-1-kal.conley@dectris.com
    Signed-off-by: Martin KaFai Lau <martin.lau@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ff4d8040593916795ebd17282f6b2fd53c62d463
Author: Kal Conley <kal.conley@dectris.com>
Date:   Mon Apr 3 16:50:46 2023 +0200

    selftests: xsk: Use correct UMEM size in testapp_invalid_desc

    [ Upstream commit 7a2050df244e2c9a4e90882052b7907450ad10ed ]

    Avoid UMEM_SIZE macro in testapp_invalid_desc which is incorrect when
    the frame size is not XSK_UMEM__DEFAULT_FRAME_SIZE. Also remove the
    macro since it's no longer being used.

    Fixes: 909f0e28207c ("selftests: xsk: Add tests for 2K frame size")
    Signed-off-by: Kal Conley <kal.conley@dectris.com>
    Acked-by: Magnus Karlsson <magnus.karlsson@intel.com>
    Link: https://lore.kernel.org/r/20230403145047.33065-2-kal.conley@dectris.com
    Signed-off-by: Martin KaFai Lau <martin.lau@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2e4654a60fee4aaa0ed60e2fd667e3cded98922e
Author: Simon Horman <horms@kernel.org>
Date:   Mon Apr 3 17:43:16 2023 +0200

    net: qrtr: correct types of trace event parameters

    [ Upstream commit 054fbf7ff8143d35ca7d3bb5414bb44ee1574194 ]

    The arguments passed to the trace events are of type unsigned int,
    however the signature of the events used __le32 parameters.

    I may be missing the point here, but sparse flagged this and it
    does seem incorrect to me.

      net/qrtr/ns.c: note: in included file (through include/trace/trace_events.h,
include/trace/define_trace.h, include/trace/events/qrtr.h):
      ./include/trace/events/qrtr.h:11:1: warning: cast to restricted __le32
      ./include/trace/events/qrtr.h:11:1: warning: restricted __le32 degrades to integer
      ./include/trace/events/qrtr.h:11:1: warning: restricted __le32 degrades to integer
      ... (a lot more similar warnings)
      net/qrtr/ns.c:115:47:    expected restricted __le32 [usertype] service
      net/qrtr/ns.c:115:47:    got unsigned int service
      net/qrtr/ns.c:115:61: warning: incorrect type in argument 2 (different base types)
      ... (a lot more similar warnings)
```

```
    Fixes: dfddb54043f0 ("net: qrtr: Add tracepoint support")
    Reviewed-by: Manivannan Sadhasivam <mani@kernel.org>
    Signed-off-by: Simon Horman <horms@kernel.org>
    Link: https://lore.kernel.org/r/20230402-qrtr-trace-types-v1-1-92ad55008dd3@kernel.org
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0023068d0800d91c3a61ecb7c118ff617257e222
Author: Qilin Tan <qilin.tan@mediatek.com>
Date:   Fri Mar 31 17:26:56 2023 +0800

    f2fs: fix iostat lock protection

    [ Upstream commit 144f1cd40bf91fb3ac1d41806470756ce774f389 ]

    Made iostat lock irq safe to avoid potentinal deadlock.

    Deadlock scenario:
    f2fs_attr_store
      -> f2fs_sbi_store
      -> _sbi_store
      -> spin_lock(sbi->iostat_lock)
        <interrupt request>
        -> scsi_end_request
        -> bio_endio
        -> f2fs_dio_read_end_io
        -> f2fs_update_iostat
        -> spin_lock_irqsave(sbi->iostat_lock)  ===> Dead lock here

    Fixes: 61803e984307 ("f2fs: fix iostat related lock protection")
    Fixes: a1e09b03e6f5 ("f2fs: use iomap for direct I/O")
    Signed-off-by: Qilin Tan <qilin.tan@mediatek.com>
    Reviewed-by: Chao Yu <chao@kernel.org>
    Signed-off-by: Jaegeuk Kim <jaegeuk@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c41de5c67dfb1b17294635fc403180a62e87bf42
Author: Dave Marchevsky <davemarchevsky@fb.com>
Date:   Mon Apr 3 13:00:27 2023 -0700

    bpf: Fix struct_meta lookup for bpf_obj_free_fields kfunc call

    [ Upstream commit f6a6a5a976288e4d0d94eb1c6c9e983e8e5cdb31 ]

    bpf_obj_drop_impl has a void return type. In check_kfunc_call, the "else
    if" which sets insn_aux->kptr_struct_meta for bpf_obj_drop_impl is
    surrounded by a larger if statement which checks btf_type_is_ptr. As a
    result:

      * The bpf_obj_drop_impl-specific code will never execute
      * The btf_struct_meta input to bpf_obj_drop is always NULL
      * __bpf_obj_drop_impl will always see a NULL btf_record when called
        from BPF program, and won't call bpf_obj_free_fields
      * program-allocated kptrs which have fields that should be cleaned up
        by bpf_obj_free_fields may instead leak resources

    This patch adds a btf_type_is_void branch to the larger if and moves
    special handling for bpf_obj_drop_impl there, fixing the issue.

    Fixes: ac9f06050a35 ("bpf: Introduce bpf_obj_drop")
    Cc: Kumar Kartikeya Dwivedi <memxor@gmail.com>
    Signed-off-by: Dave Marchevsky <davemarchevsky@fb.com>
    Link: https://lore.kernel.org/r/20230403200027.2271029-1-davemarchevsky@fb.com
    Signed-off-by: Alexei Starovoitov <ast@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit c079b05ccbadd83d4411904581c216428e4494aa
Author: Andrii Nakryiko <andrii@kernel.org>
Date:   Wed Mar 8 10:41:14 2023 -0800

    bpf: factor out fetching basic kfunc metadata

    [ Upstream commit 07236eab7a3139da97aef9f5f21f403be82a82ea ]

    Factor out logic to fetch basic kfunc metadata based on struct bpf_insn.
    This is not exactly short or trivial code to just copy/paste and this
    information is sometimes necessary in other parts of the verifier logic.
    Subsequent patches will rely on this to determine if an instruction is
    a kfunc call to iterator next method.

    No functional changes intended, including that verbose() warning
    behavior when kfunc is not allowed for a particular program type.

    Signed-off-by: Andrii Nakryiko <andrii@kernel.org>
    Link: https://lore.kernel.org/r/20230308184121.1165081-2-andrii@kernel.org
    Signed-off-by: Alexei Starovoitov <ast@kernel.org>
    Stable-dep-of: f6a6a5a97628 ("bpf: Fix struct_meta lookup for bpf_obj_free_fields kfunc
call")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0354bce76ed1d775904acdb4cc0bf88c5b9b5b9f
Author: Armin Wolf <W_Armin@gmx.de>
Date:   Thu Mar 30 23:56:37 2023 +0200

    wifi: rt2x00: Fix memory leak when handling surveys

    [ Upstream commit cbef9a83c51dfcb07f77cfa6ac26f53a1ea86f49 ]

    When removing a rt2x00 device, its associated channel surveys
    are not freed, causing a memory leak observable with kmemleak:

    unreferenced object 0xffff9620f0881a00 (size 512):
      comm "systemd-udevd", pid 2290, jiffies 4294906974 (age 33.768s)
      hex dump (first 32 bytes):
        70 44 12 00 00 00 00 00 92 8a 00 00 00 00 00 00  pD..............
        00 00 00 00 00 00 00 00 ab 87 01 00 00 00 00 00  ................
      backtrace:
        [<ffffffffb0ed858b>] __kmalloc+0x4b/0x130
        [<ffffffffc1b0f29b>] rt2800_probe_hw+0xc2b/0x1380 [rt2800lib]
        [<ffffffffc1a9496e>] rt2800usb_probe_hw+0xe/0x60 [rt2800usb]
        [<ffffffffc1ae491a>] rt2x00lib_probe_dev+0x21a/0x7d0 [rt2x00lib]
        [<ffffffffc1b3b83e>] rt2x00usb_probe+0x1be/0x980 [rt2x00usb]
        [<ffffffffc05981e2>] usb_probe_interface+0xe2/0x310 [usbcore]
        [<ffffffffb13be2d5>] really_probe+0x1a5/0x410
        [<ffffffffb13be5c8>] __driver_probe_device+0x78/0x180
        [<ffffffffb13be6fe>] driver_probe_device+0x1e/0x90
        [<ffffffffb13be972>] __driver_attach+0xd2/0x1c0
        [<ffffffffb13bbc57>] bus_for_each_dev+0x77/0xd0
        [<ffffffffb13bd2a2>] bus_add_driver+0x112/0x210
        [<ffffffffb13bfc6c>] driver_register+0x5c/0x120
        [<ffffffffc0596ae8>] usb_register_driver+0x88/0x150 [usbcore]
        [<ffffffffb0c011c4>] do_one_initcall+0x44/0x220
        [<ffffffffb0d6134c>] do_init_module+0x4c/0x220

    Fix this by freeing the channel surveys on device removal.

    Tested with a RT3070 based USB wireless adapter.

    Fixes: 5447626910f5 ("rt2x00: save survey for every channel visited")
    Signed-off-by: Armin Wolf <W_Armin@gmx.de>
```

```
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Acked-by: Stanislaw Gruszka <stf_xl@wp.pl>
    Signed-off-by: Kalle Valo <kvalo@kernel.org>
    Link: https://lore.kernel.org/r/20230330215637.4332-1-W_Armin@gmx.de
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 15c8c81081ff59eae2d9852ac45e31fa353d912d
Author: Xingui Yang <yangxingui@huawei.com>
Date:   Mon Mar 20 11:34:23 2023 +0800

    scsi: hisi_sas: Handle NCQ error when IPTT is valid

    [ Upstream commit bb544224da77b96b2c11a13872bf91ede1e015be ]

    If an NCQ error occurs when the IPTT is valid and slot->abort flag is set
    in completion path, sas_task_abort() will be called to abort only one NCQ
    command now, and the host would be set to SHOST_RECOVERY state. But this
    may not kick-off EH Immediately until other outstanding QCs timeouts. As a
    result, the host may remain in the SHOST_RECOVERY state for up to 30
    seconds, such as follows:

    [7972317.645234] hisi_sas_v3_hw 0000:74:04.0: erroneous completion iptt=3264
task=00000000466116b8 dev id=2 sas_addr=0x5000000000000502 CQ hdr: 0x1883 0x20cc0 0x40000
0x20420000 Error info: 0x0 0x0 0x200000 0x0
    [7972341.508264] sas: Enter sas_scsi_recover_host busy: 32 failed: 32
    [7972341.984731] sas: --- Exit sas_scsi_recover_host: busy: 0 failed: 32 tries: 1

    All NCQ commands that are in the queue should be aborted when an NCQ error
    occurs in this scenario.

    Fixes: 05d91b557af9 ("scsi: hisi_sas: Directly trigger SCSI error handling for
completion errors")
    Signed-off-by: Xingui Yang <yangxingui@huawei.com>
    Signed-off-by: Xiang Chen <chenxiang66@hisilicon.com>
    Link: https://lore.kernel.org/r/1679283265-115066-3-git-send-email-
chenxiang66@hisilicon.com
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e8b61339ed62f5ef1560f11b851a5f249df53357
Author: Wei Chen <harperchen1110@gmail.com>
Date:   Sun Mar 26 05:42:17 2023 +0000

    wifi: rtlwifi: fix incorrect error codes in rtl_debugfs_set_write_reg()

    [ Upstream commit 5dbe1f8eb8c5ac69394400a5b86fd81775e96c43 ]

    If there is a failure during copy_from_user or user-provided data buffer is
    invalid, rtl_debugfs_set_write_reg should return negative error code instead
    of a positive value count.

    Fix this bug by returning correct error code. Moreover, the check of buffer
    against null is removed since it will be handled by copy_from_user.

    Fixes: 610247f46feb ("rtlwifi: Improve debugging by using debugfs")
    Signed-off-by: Wei Chen <harperchen1110@gmail.com>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Signed-off-by: Kalle Valo <kvalo@kernel.org>
    Link: https://lore.kernel.org/r/20230326054217.93492-1-harperchen1110@gmail.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f4614ac40d8cc60d91f2afcd49c8e1ae1247b7e3
Author: Wei Chen <harperchen1110@gmail.com>
Date:   Sun Mar 26 05:31:38 2023 +0000
```

```
    wifi: rtlwifi: fix incorrect error codes in rtl_debugfs_set_write_rfreg()

    [ Upstream commit 905a9241e4e8c15d2c084fee916280514848fe35 ]

    If there is a failure during copy_from_user or user-provided data buffer
    is invalid, rtl_debugfs_set_write_rfreg should return negative error code
    instead of a positive value count.

    Fix this bug by returning correct error code. Moreover, the check of buffer
    against null is removed since it will be handled by copy_from_user.

    Fixes: 610247f46feb ("rtlwifi: Improve debugging by using debugfs")
    Signed-off-by: Wei Chen <harperchen1110@gmail.com>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Signed-off-by: Kalle Valo <kvalo@kernel.org>
    Link: https://lore.kernel.org/r/20230326053138.91338-1-harperchen1110@gmail.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 772398573c6087adbf49c04079477928ea08b6b8
Author: Suman Anna <s-anna@ti.com>
Date:   Fri Mar 24 20:28:12 2023 +0530

    crypto: sa2ul - Select CRYPTO_DES

    [ Upstream commit 8832023efd20966e29944dac92118dfbf1fa1bc0 ]

    The SA2UL Crypto driver provides support for couple of
    DES3 algos "cbc(des3_ede)" and "ecb(des3_ede)", and enabling
    the crypto selftest throws the following errors (as seen on
    K3 J721E SoCs):
      saul-crypto 4e00000.crypto: Error allocating fallback algo cbc(des3_ede)
      alg: skcipher: failed to allocate transform for cbc-des3-sa2ul: -2
      saul-crypto 4e00000.crypto: Error allocating fallback algo ecb(des3_ede)
      alg: skcipher: failed to allocate transform for ecb-des3-sa2ul: -2

    Fix this by selecting CRYPTO_DES which was missed while
    adding base driver support.

    Fixes: 7694b6ca649f ("crypto: sa2ul - Add crypto driver")
    Signed-off-by: Suman Anna <s-anna@ti.com>
    Signed-off-by: Jayesh Choudhary <j-choudhary@ti.com>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e1b18b6a6cedbf41c56465e8c05994f83929c23c
Author: Christophe JAILLET <christophe.jaillet@wanadoo.fr>
Date:   Tue Mar 21 07:59:30 2023 +0100

    crypto: caam - Clear some memory in instantiate_rng

    [ Upstream commit 9c19fb86a8cb2ee82a832c95e139f29ea05c4d08 ]

    According to the comment at the end of the 'for' loop just a few lines
    below, it looks needed to clear 'desc'.

    So it should also be cleared for the first iteration.

    Move the memset() to the beginning of the loop to be safe.

    Fixes: 281922a1d4f5 ("crypto: caam - add support for SEC v5.x RNG4")
    Signed-off-by: Christophe JAILLET <christophe.jaillet@wanadoo.fr>
    Reviewed-by: Gaurav Jain <gaurav.jain@nxp.com>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit d2746c56dd2cc47f70cc3931977be556172c246d
Author: Jaegeuk Kim <jaegeuk@kernel.org>
Date:   Thu Mar 23 15:37:54 2023 -0700

    f2fs: fix scheduling while atomic in decompression path

    [ Upstream commit 1aa161e43106d46ca8e9a86f4aa28d420258134b ]

    [   16.945668][    C0] Call trace:
    [   16.945678][    C0]  dump_backtrace+0x110/0x204
    [   16.945706][    C0]  dump_stack_lvl+0x84/0xbc
    [   16.945735][    C0]  __schedule_bug+0xb8/0x1ac
    [   16.945756][    C0]  __schedule+0x724/0xbdc
    [   16.945778][    C0]  schedule+0x154/0x258
    [   16.945793][    C0]  bit_wait_io+0x48/0xa4
    [   16.945808][    C0]  out_of_line_wait_on_bit+0x114/0x198
    [   16.945824][    C0]  __sync_dirty_buffer+0x1f8/0x2e8
    [   16.945853][    C0]  __f2fs_commit_super+0x140/0x1f4
    [   16.945881][    C0]  f2fs_commit_super+0x110/0x28c
    [   16.945898][    C0]  f2fs_handle_error+0x1f4/0x2f4
    [   16.945917][    C0]  f2fs_decompress_cluster+0xc4/0x450
    [   16.945942][    C0]  f2fs_end_read_compressed_page+0xc0/0xfc
    [   16.945959][    C0]  f2fs_handle_step_decompress+0x118/0x1cc
    [   16.945978][    C0]  f2fs_read_end_io+0x168/0x2b0
    [   16.945993][    C0]  bio_endio+0x25c/0x2c8
    [   16.946015][    C0]  dm_io_dec_pending+0x3e8/0x57c
    [   16.946052][    C0]  clone_endio+0x134/0x254
    [   16.946069][    C0]  bio_endio+0x25c/0x2c8
    [   16.946084][    C0]  blk_update_request+0x1d4/0x478
    [   16.946103][    C0]  scsi_end_request+0x38/0x4cc
    [   16.946129][    C0]  scsi_io_completion+0x94/0x184
    [   16.946147][    C0]  scsi_finish_command+0xe8/0x154
    [   16.946164][    C0]  scsi_complete+0x90/0x1d8
    [   16.946181][    C0]  blk_done_softirq+0xa4/0x11c
    [   16.946198][    C0]  _stext+0x184/0x614
    [   16.946214][    C0]  __irq_exit_rcu+0x78/0x144
    [   16.946234][    C0]  handle_domain_irq+0xd4/0x154
    [   16.946260][    C0]  gic_handle_irq.33881+0x5c/0x27c
    [   16.946281][    C0]  call_on_irq_stack+0x40/0x70
    [   16.946298][    C0]  do_interrupt_handler+0x48/0xa4
    [   16.946313][    C0]  el1_interrupt+0x38/0x68
    [   16.946346][    C0]  el1h_64_irq_handler+0x20/0x30
    [   16.946362][    C0]  el1h_64_irq+0x78/0x7c
    [   16.946377][    C0]  finish_task_switch+0xc8/0x3d8
    [   16.946394][    C0]  __schedule+0x600/0xbdc
    [   16.946408][    C0]  preempt_schedule_common+0x34/0x5c
    [   16.946423][    C0]  preempt_schedule+0x44/0x48
    [   16.946438][    C0]  process_one_work+0x30c/0x550
    [   16.946456][    C0]  worker_thread+0x414/0x8bc
    [   16.946472][    C0]  kthread+0x16c/0x1e0
    [   16.946486][    C0]  ret_from_fork+0x10/0x20

    Fixes: bff139b49d9f ("f2fs: handle decompress only post processing in softirq")
    Fixes: 95fa90c9e5a7 ("f2fs: support recording errors into superblock")
    Reviewed-by: Chao Yu <chao@kernel.org>
    Signed-off-by: Jaegeuk Kim <jaegeuk@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ad31eed06c3b4d63b2d38322a271d4009aee4bb3
Author: Yangtao Li <frank.li@vivo.com>
Date:   Tue Mar 21 01:22:18 2023 +0800

    f2fs: compress: fix to call f2fs_wait_on_page_writeback() in f2fs_write_raw_pages()

    [ Upstream commit babedcbac164cec970872b8097401ca913a80e61 ]
```

```
BUG_ON() will be triggered when writing files concurrently,
because the same page is writtenback multiple times.

1597 void folio_end_writeback(struct folio *folio)
1598 {
                ......
1618     if (!__folio_end_writeback(folio))
1619         BUG();
                ......
1625 }

kernel BUG at mm/filemap.c:1619!
Call Trace:
 <TASK>
 f2fs_write_end_io+0x1a0/0x370
 blk_update_request+0x6c/0x410
 blk_mq_end_request+0x15/0x130
 blk_complete_reqs+0x3c/0x50
 __do_softirq+0xb8/0x29b
 ? sort_range+0x20/0x20
 run_ksoftirqd+0x19/0x20
 smpboot_thread_fn+0x10b/0x1d0
 kthread+0xde/0x110
 ? kthread_complete_and_exit+0x20/0x20
 ret_from_fork+0x22/0x30
 </TASK>

Below is the concurrency scenario:

[Process A]             [Process B]             [Process C]
f2fs_write_raw_pages()
  - redirty_page_for_writepage()
  - unlock page()
                        f2fs_do_write_data_page()
                          - lock_page()
                          - clear_page_dirty_for_io()
                          - set_page_writeback() [1st writeback]
                            .....
                            - unlock page()

                                                generic_perform_write()
                                                  - f2fs_write_begin()
                                                    - wait_for_stable_page()

                                                  - f2fs_write_end()
                                                    - set_page_dirty()

  - lock_page()
    - f2fs_do_write_data_page()
      - set_page_writeback() [2st writeback]

This problem was introduced by the previous commit 7377e853967b ("f2fs:
compress: fix potential deadlock of compress file"). All pagelocks were
released in f2fs_write_raw_pages(), but whether the page was
in the writeback state was ignored in the subsequent writing process.
Let's fix it by waiting for the page to writeback before writing.

Cc: Christoph Hellwig <hch@lst.de>
Fixes: 4c8ff7095bef ("f2fs: support data compression")
Fixes: 7377e853967b ("f2fs: compress: fix potential deadlock of compress file")
Signed-off-by: Qi Han <hanqi@vivo.com>
Signed-off-by: Yangtao Li <frank.li@vivo.com>
Reviewed-by: Chao Yu <chao@kernel.org>
Signed-off-by: Jaegeuk Kim <jaegeuk@kernel.org>
```

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

commit 26ae4405ce4d850832d67833d8137ad54bbc198f
Author: Jaegeuk Kim <jaegeuk@kernel.org>
Date:   Tue Mar 21 15:58:04 2023 -0700

    f2fs: apply zone capacity to all zone type

    [ Upstream commit 0b37ed21e3367539b79284e0b0af2246ffcf0dca ]

    If we manage the zone capacity per zone type, it'll break the GC assumption.
    And, the current logic complains valid block count mismatch.
    Let's apply zone capacity to all zone type, if specified.

    Fixes: de881df97768 ("f2fs: support zone capacity less than zone size")
    Reviewed-by: Chao Yu <chao@kernel.org>
    Signed-off-by: Jaegeuk Kim <jaegeuk@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7e9030d2d917ddcc4146905f5191d263863e9468
Author: Yonggil Song <yonggil.song@samsung.com>
Date:   Thu Feb 16 16:13:50 2023 +0900

    f2fs: fix uninitialized skipped_gc_rwsem

    [ Upstream commit c17caf0ba3aa3411b96c71b4ce24be1040b8f3e8 ]

    When f2fs skipped a gc round during victim migration, there was a bug which
    would skip all upcoming gc rounds unconditionally because skipped_gc_rwsem
    was not initialized. It fixes the bug by correctly initializing the
    skipped_gc_rwsem inside the gc loop.

    Fixes: 6f8d4455060d ("f2fs: avoid fi->i_gc_rwsem[WRITE] lock in f2fs_gc")
    Signed-off-by: Yonggil Song <yonggil.song@samsung.com>
    Reviewed-by: Chao Yu <chao@kernel.org>
    Signed-off-by: Jaegeuk Kim <jaegeuk@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4ba9f68a67ac7fc878ed77b82e4557586c7763e1
Author: Yangtao Li <frank.li@vivo.com>
Date:   Tue Feb 21 22:45:50 2023 +0800

    f2fs: handle dqget error in f2fs_transfer_project_quota()

    [ Upstream commit 8051692f5f23260215bfe9a72e712d93606acc5f ]

    We should set the error code when dqget() failed.

    Fixes: 2c1d03056991 ("f2fs: support F2FS_IOC_FS{GET,SET}XATTR")
    Signed-off-by: Yangtao Li <frank.li@vivo.com>
    Signed-off-by: Jaegeuk Kim <jaegeuk@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4a4b26794427b57762930382b9cd1f14ca4247fa
Author: Bobby Eshleman <bobby.eshleman@bytedance.com>
Date:   Mon Mar 27 22:16:06 2023 +0000

    testing/vsock: add vsock_perf to gitignore

    [ Upstream commit 24265c2c91ad6aae9446e18472566cd83e92b602 ]

    This adds the vsock_perf binary to the gitignore file.

    Fixes: 8abbffd27ced ("test/vsock: vsock_perf utility")
    Signed-off-by: Bobby Eshleman <bobby.eshleman@bytedance.com>
```

```
    Reviewed-by: Arseniy Krasnov <AVKrasnov@sberdevices.ru>
    Reviewed-by: Stefano Garzarella <sgarzare@redhat.com>
    Link: https://lore.kernel.org/r/20230327-vsock-add-vsock-perf-to-ignore-v1-1-
f28a84f3606b@bytedance.com
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 317371874c511a8e58abd68e0eafaf859e9342e9
Author: Sean Anderson <seanga2@gmail.com>
Date:   Fri Mar 24 13:51:27 2023 -0400

    net: sunhme: Fix uninitialized return code

    [ Upstream commit d61157414d0a591d10d27d0ce5873916614e5e31 ]

    Fix an uninitialized return code if we never found a qfe slot. It would be
    a bug if we ever got into this situation, but it's good to return something
    tracable.

    Fixes: acb3f35f920b ("sunhme: forward the error code from pci_enable_device()")
    Reported-by: kernel test robot <lkp@intel.com>
    Reported-by: Dan Carpenter <error27@gmail.com>
    Signed-off-by: Sean Anderson <seanga2@gmail.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f29d5a8e728d943e47482b15841b25b4b74df40f
Author: Danila Chernetsov <listdansp@mail.ru>
Date:   Fri Mar 17 17:51:09 2023 +0000

    scsi: megaraid: Fix mega_cmd_done() CMDID_INT_CMDS

    [ Upstream commit 75cb113cd43f06aaf4f1bda0069cfd5b98e909eb ]

    When cmdid == CMDID_INT_CMDS, the 'cmds' pointer is NULL but is
    dereferenced below.

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

    Fixes: 0f2bb84d2a68 ("[SCSI] megaraid: simplify internal command handling")
    Signed-off-by: Danila Chernetsov <listdansp@mail.ru>
    Link: https://lore.kernel.org/r/20230317175109.18585-1-listdansp@mail.ru
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2c8f1d4f90b9f6a781687bf1421196c43f03f064
Author: Mike Christie <michael.christie@oracle.com>
Date:   Sat Mar 18 20:56:19 2023 -0500

    scsi: target: iscsit: Fix TAS handling during conn cleanup

    [ Upstream commit cc79da306ebb2edb700c3816b90219223182ac3c ]

    Fix a bug added in commit f36199355c64 ("scsi: target: iscsi: Fix cmd abort
    fabric stop race").

    If CMD_T_TAS is set on the se_cmd we must call iscsit_free_cmd() to do the
    last put on the cmd and free it, because the connection is down and we will
    not up sending the response and doing the put from the normal I/O
    path.

    Add a check for CMD_T_TAS in iscsit_release_commands_from_conn() so we now
    detect this case and run iscsit_free_cmd().

    Fixes: f36199355c64 ("scsi: target: iscsi: Fix cmd abort fabric stop race")
```

```
    Signed-off-by: Mike Christie <michael.christie@oracle.com>
    Link: https://lore.kernel.org/r/20230319015620.96006-9-michael.christie@oracle.com
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2c43de56f9220dca3e28c774d1c5e2cab574223a
Author: Mike Christie <michael.christie@oracle.com>
Date:   Sat Mar 18 20:56:18 2023 -0500

    scsi: target: Fix multiple LUN_RESET handling

    [ Upstream commit 673db054d7a2b5a470d7a25baf65956d005ad729 ]

    This fixes a bug where an initiator thinks a LUN_RESET has cleaned up
    running commands when it hasn't. The bug was added in commit 51ec502a3266
    ("target: Delete tmr from list before processing").

    The problem occurs when:

     1. We have N I/O cmds running in the target layer spread over 2 sessions.

     2. The initiator sends a LUN_RESET for each session.

     3. session1's LUN_RESET loops over all the running commands from both
        sessions and moves them to its local drain_task_list.

     4. session2's LUN_RESET does not see the LUN_RESET from session1 because
        the commit above has it remove itself. session2 also does not see any
        commands since the other reset moved them off the state lists.

     5. sessions2's LUN_RESET will then complete with a successful response.

     6. sessions2's inititor believes the running commands on its session are
        now cleaned up due to the successful response and cleans up the running
        commands from its side. It then restarts them.

     7. The commands do eventually complete on the backend and the target
        starts to return aborted task statuses for them. The initiator will
        either throw a invalid ITT error or might accidentally lookup a new
        task if the ITT has been reallocated already.

    Fix the bug by reverting the patch, and serialize the execution of
    LUN_RESETs and Preempt and Aborts.

    Also prevent us from waiting on LUN_RESETs in core_tmr_drain_tmr_list,
    because it turns out the original patch fixed a bug that was not
    mentioned. For LUN_RESET1 core_tmr_drain_tmr_list can see a second
    LUN_RESET and wait on it. Then the second reset will run
    core_tmr_drain_tmr_list and see the first reset and wait on it resulting in
    a deadlock.

    Fixes: 51ec502a3266 ("target: Delete tmr from list before processing")
    Signed-off-by: Mike Christie <michael.christie@oracle.com>
    Link: https://lore.kernel.org/r/20230319015620.96006-8-michael.christie@oracle.com
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 80f7caf19a67cfc0091eda332e9fc51f7b56bdfb
Author: Mike Christie <michael.christie@oracle.com>
Date:   Sat Mar 18 20:56:16 2023 -0500

    scsi: target: iscsit: Stop/wait on cmds during conn close

    [ Upstream commit 395cee83d02de3073211b04fc85724f4abc663ad ]
```

```
    This fixes a bug added in commit f36199355c64 ("scsi: target: iscsi: Fix
    cmd abort fabric stop race").

    If we have multiple sessions to the same se_device we can hit a race where
    a LUN_RESET on one session cleans up the se_cmds from under another
    session which is being closed. This results in the closing session freeing
    its conn/session structs while they are still in use.

    The bug is:

     1. Session1 has IO se_cmd1.

     2. Session2 can also have se_cmds for I/O and optionally TMRs for ABORTS
        but then gets a LUN_RESET.

     3. The LUN_RESET on session2 sees the se_cmds on session1 and during the
        drain stages marks them all with CMD_T_ABORTED.

     4. session1 is now closed so iscsit_release_commands_from_conn() only sees
        se_cmds with the CMD_T_ABORTED bit set and returns immediately even
        though we have outstanding commands.

     5. session1's connection and session are freed.

     6. The backend request for se_cmd1 completes and it accesses the freed
        connection/session.

    This hooks the iscsit layer into the cmd counter code, so we can wait for
    all outstanding se_cmds before freeing the connection.

    Fixes: f36199355c64 ("scsi: target: iscsi: Fix cmd abort fabric stop race")
    Signed-off-by: Mike Christie <michael.christie@oracle.com>
    Link: https://lore.kernel.org/r/20230319015620.96006-6-michael.christie@oracle.com
    Reviewed-by: Maurizio Lombardi <mlombard@redhat.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 32d530c85cae8f0eb309b61573fee756f9949dad
Author: Mike Christie <michael.christie@oracle.com>
Date:   Sat Mar 18 20:56:15 2023 -0500

    scsi: target: iscsit: isert: Alloc per conn cmd counter

    [ Upstream commit 6d256bee602b131bd4fbc92863b6a1210bcf6325 ]

    This has iscsit allocate a per conn cmd counter and converts iscsit/isert
    to use it instead of the per session one.

    Signed-off-by: Mike Christie <michael.christie@oracle.com>
    Link: https://lore.kernel.org/r/20230319015620.96006-5-michael.christie@oracle.com
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Stable-dep-of: 395cee83d02d ("scsi: target: iscsit: Stop/wait on cmds during conn
close")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit fba08c6ee83d142552cd43d383ea6208df336fda
Author: Mike Christie <michael.christie@oracle.com>
Date:   Sat Mar 18 20:56:14 2023 -0500

    scsi: target: Pass in cmd counter to use during cmd setup

    [ Upstream commit 8e288be8606ad87c1726618eacfb8fbd3ab4b806 ]

    Allow target_get_sess_cmd() users to pass in the cmd counter they want to
    use. Right now we pass in the session's cmd counter but in a subsequent
```

```
    commit iSCSI will switch from per session to per conn.

    Signed-off-by: Mike Christie <michael.christie@oracle.com>
    Link: https://lore.kernel.org/r/20230319015620.96006-4-michael.christie@oracle.com
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Stable-dep-of: 395cee83d02d ("scsi: target: iscsit: Stop/wait on cmds during conn
close")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b51e30b5849785f9d36a92288fd2083e25fee276
Author: Mike Christie <michael.christie@oracle.com>
Date:   Sat Mar 18 20:56:13 2023 -0500

    scsi: target: Move cmd counter allocation

    [ Upstream commit 4edba7e4a8f39112398d3cda94128a8e13a7d527 ]

    iSCSI needs to allocate its cmd counter per connection for MCS support
    where we need to stop and wait on commands running on a connection instead
    of per session. This moves the cmd counter allocation to
    target_setup_session() which is used by drivers that need the stop+wait
    behavior per session.

    xcopy doesn't need stop+wait at all, so we will be OK moving the cmd
    counter allocation outside of transport_init_session().

    Signed-off-by: Mike Christie <michael.christie@oracle.com>
    Link: https://lore.kernel.org/r/20230319015620.96006-3-michael.christie@oracle.com
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Stable-dep-of: 395cee83d02d ("scsi: target: iscsit: Stop/wait on cmds during conn
close")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1eaaf1b828cdaa58abccc68962d24005fd5e8852
Author: Mike Christie <michael.christie@oracle.com>
Date:   Sat Mar 18 20:56:12 2023 -0500

    scsi: target: Move sess cmd counter to new struct

    [ Upstream commit becd9be6069e7b183c084f460f0eb363e43cc487 ]

    iSCSI needs to wait on outstanding commands like how SRP and the FC/FCoE
    drivers do. It can't use target_stop_session() because for MCS support we
    can't stop the entire session during recovery because if other connections
    are OK then we want to be able to continue to execute I/O on them.

    Move the per session cmd counters to a new struct so iSCSI can allocate
    them per connection. The xcopy code can also just not allocate in the
    future since it doesn't need to track commands.

    Signed-off-by: Mike Christie <michael.christie@oracle.com>
    Link: https://lore.kernel.org/r/20230319015620.96006-2-michael.christie@oracle.com
    Reviewed-by: Maurizio Lombardi <mlombard@redhat.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Stable-dep-of: 395cee83d02d ("scsi: target: iscsit: Stop/wait on cmds during conn
close")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 32057953f5ce73ebee1f9955abddf5e482e144cd
Author: Daniel Borkmann <daniel@iogearbox.net>
Date:   Wed Mar 22 22:30:55 2023 +0100

    bpf: Fix __reg_bound_offset 64->32 var_off subreg propagation

    [ Upstream commit 7be14c1c9030f73cc18b4ff23b78a0a081f16188 ]
```

```
    Xu reports that after commit 3f50f132d840 ("bpf: Verifier, do explicit ALU32
    bounds tracking"), the following BPF program is rejected by the verifier:

       0: (61) r2 = *(u32 *)(r1 +0)          ; R2_w=pkt(off=0,r=0,imm=0)
       1: (61) r3 = *(u32 *)(r1 +4)          ; R3_w=pkt_end(off=0,imm=0)
       2: (bf) r1 = r2
       3: (07) r1 += 1
       4: (2d) if r1 > r3 goto pc+8
       5: (71) r1 = *(u8 *)(r2 +0)           ; R1_w=scalar(umax=255,var_off=(0x0; 0xff))
       6: (18) r0 = 0x7fffffffffffff10
       8: (0f) r1 += r0                      ;
R1_w=scalar(umin=0x7fffffffffffff10,umax=0x800000000000000f)
       9: (18) r0 = 0x8000000000000000
      11: (07) r0 += 1
      12: (ad) if r0 < r1 goto pc-2
      13: (b7) r0 = 0
      14: (95) exit

    And the verifier log says:

      func#0 @0
      0: R1=ctx(off=0,imm=0) R10=fp0
      0: (61) r2 = *(u32 *)(r1 +0)          ; R1=ctx(off=0,imm=0) R2_w=pkt(off=0,r=0,imm=0)
      1: (61) r3 = *(u32 *)(r1 +4)          ; R1=ctx(off=0,imm=0) R3_w=pkt_end(off=0,imm=0)
      2: (bf) r1 = r2                       ; R1_w=pkt(off=0,r=0,imm=0)
R2_w=pkt(off=0,r=0,imm=0)
      3: (07) r1 += 1                       ; R1_w=pkt(off=1,r=0,imm=0)
      4: (2d) if r1 > r3 goto pc+8          ; R1_w=pkt(off=1,r=1,imm=0)
R3_w=pkt_end(off=0,imm=0)
      5: (71) r1 = *(u8 *)(r2 +0)           ; R1_w=scalar(umax=255,var_off=(0x0; 0xff))
R2_w=pkt(off=0,r=1,imm=0)
      6: (18) r0 = 0x7fffffffffffff10       ; R0_w=9223372036854775568
      8: (0f) r1 += r0                      ; R0_w=9223372036854775568
R1_w=scalar(umin=9223372036854775568,umax=9223372036854775823,s32_min=-240,s32_max=15)
      9: (18) r0 = 0x8000000000000000       ; R0_w=-9223372036854775808
      11: (07) r0 += 1                      ; R0_w=-9223372036854775807
      12: (ad) if r0 < r1 goto pc-2         ; R0_w=-9223372036854775807
R1_w=scalar(umin=9223372036854775568,umax=9223372036854775809)
      13: (b7) r0 = 0                       ; R0_w=0
      14: (95) exit

      from 12 to 11: R0_w=-9223372036854775807
R1_w=scalar(umin=9223372036854775810,umax=9223372036854775823,var_off=(0x8000000000000000;
0xffffffff)) R2_w=pkt(off=0,r=1,imm=0) R3_w=pkt_end(off=0,imm=0) R10=fp0
      11: (07) r0 += 1                      ; R0_w=-9223372036854775806
      12: (ad) if r0 < r1 goto pc-2         ; R0_w=-9223372036854775806
R1_w=scalar(umin=9223372036854775810,umax=9223372036854775810,var_off=(0x8000000000000000;
0xffffffff))
      13: safe

      [...]

      from 12 to 11: R0_w=-9223372036854775795
R1=scalar(umin=9223372036854775822,umax=9223372036854775823,var_off=(0x8000000000000000;
0xffffffff)) R2=pkt(off=0,r=1,imm=0) R3=pkt_end(off=0,imm=0) R10=fp0
      11: (07) r0 += 1                      ; R0_w=-9223372036854775794
      12: (ad) if r0 < r1 goto pc-2         ; R0_w=-9223372036854775794
R1=scalar(umin=9223372036854775822,umax=9223372036854775822,var_off=(0x8000000000000000;
0xffffffff))
      13: safe

      from 12 to 11: R0_w=-9223372036854775794
R1=scalar(umin=9223372036854775823,umax=9223372036854775823,var_off=(0x8000000000000000;
0xffffffff)) R2=pkt(off=0,r=1,imm=0) R3=pkt_end(off=0,imm=0) R10=fp0
```

```
      11: (07) r0 += 1                      ; R0_w=-9223372036854775793
      12: (ad) if r0 < r1 goto pc-2         ; R0_w=-9223372036854775793
 R1=scalar(umin=9223372036854775823,umax=9223372036854775823,var_off=(0x8000000000000000;
 0xffffffff))
      13: safe

      from 12 to 11: R0_w=-9223372036854775793
 R1=scalar(umin=9223372036854775824,umax=9223372036854775823,var_off=(0x8000000000000000;
 0xffffffff)) R2=pkt(off=0,r=1,imm=0) R3=pkt_end(off=0,imm=0) R10=fp0
      11: (07) r0 += 1                      ; R0_w=-9223372036854775792
      12: (ad) if r0 < r1 goto pc-2         ; R0_w=-9223372036854775792
 R1=scalar(umin=9223372036854775824,umax=9223372036854775823,var_off=(0x8000000000000000;
 0xffffffff))
      13: safe

      [...]
```

The 64bit umin=9223372036854775810 bound continuously bumps by +1 while umax=9223372036854775823 stays as-is until the verifier complexity limit is reached and the program gets finally rejected. During this simulation, the umin also eventually surpasses umax. Looking at the first 'from 12 to 11' output line from the loop, R1 has the following state:

```
  R1_w=scalar(umin=0x8000000000000002 (9223372036854775810),
              umax=0x800000000000000f (9223372036854775823),
          var_off=(0x8000000000000000;
                   0xffffffff))
```

The var_off has technically not an inconsistent state but it's very imprecise and far off surpassing 64bit umax bounds whereas the expected output with refined known bits in var_off should have been like:

```
  R1_w=scalar(umin=0x8000000000000002 (9223372036854775810),
              umax=0x800000000000000f (9223372036854775823),
          var_off=(0x8000000000000000;
                             0xf))
```

In the above log, var_off stays as var_off=(0x8000000000000000; 0xffffffff) and does not converge into a narrower mask where more bits become known, eventually transforming R1 into a constant upon umin=9223372036854775823, umax=9223372036854775823 case where the verifier would have terminated and let the program pass.

The __reg_combine_64_into_32() marks the subregister unknown and propagates 64bit {s,u}min/{s,u}max bounds to their 32bit equivalents iff they are within the 32bit universe. The question came up whether __reg_combine_64_into_32() should special case the situation that when 64bit {s,u}min bounds have the same value as 64bit {s,u}max bounds to then assign the latter as well to the 32bit reg->{s,u}32_{min,max}_value. As can be seen from the above example however, that is just /one/ special case and not a /generic/ solution given above example would still not be addressed this way and remain at an imprecise var_off=(0x8000000000000000; 0xffffffff).

The improvement is needed in __reg_bound_offset() to refine var32_off with the updated var64_off instead of the prior reg->var_off. The reg_bounds_sync() code first refines information about the register's min/max bounds via __update_reg_bounds() from the current var_off, then in __reg_deduce_bounds() from sign bit and with the potentially learned bits from bounds it'll update the var_off tnum in __reg_bound_offset(). For example, intersecting with the old var_off might have improved bounds slightly, e.g. if umax was 0x7f...f and var_off was (0; 0xf...fc), then new var_off will then result in (0; 0x7f...fc). The intersected var64_off holds then the universe which is a superset of var32_off. The point for the latter is not to broaden, but to further refine known bits based on the intersection of var_off with 32 bit bounds, so that we later construct the final var_off

```
    from upper and lower 32 bits. The final __update_reg_bounds() can then
    potentially still slightly refine bounds if more bits became known from the
    new var_off.

    After the improvement, we can see R1 converging successively:

      func#0 @0
      0: R1=ctx(off=0,imm=0) R10=fp0
      0: (61) r2 = *(u32 *)(r1 +0)          ; R1=ctx(off=0,imm=0) R2_w=pkt(off=0,r=0,imm=0)
      1: (61) r3 = *(u32 *)(r1 +4)          ; R1=ctx(off=0,imm=0) R3_w=pkt_end(off=0,imm=0)
      2: (bf) r1 = r2                       ; R1_w=pkt(off=0,r=0,imm=0)
R2_w=pkt(off=0,r=0,imm=0)
      3: (07) r1 += 1                       ; R1_w=pkt(off=1,r=0,imm=0)
      4: (2d) if r1 > r3 goto pc+8          ; R1_w=pkt(off=1,r=1,imm=0)
R3_w=pkt_end(off=0,imm=0)
      5: (71) r1 = *(u8 *)(r2 +0)           ; R1_w=scalar(umax=255,var_off=(0x0; 0xff))
R2_w=pkt(off=0,r=1,imm=0)
      6: (18) r0 = 0x7fffffffffffff10       ; R0_w=9223372036854775568
      8: (0f) r1 += r0                      ; R0_w=9223372036854775568
R1_w=scalar(umin=9223372036854775568,umax=9223372036854775823,s32_min=-240,s32_max=15)
      9: (18) r0 = 0x8000000000000000       ; R0_w=-9223372036854775808
      11: (07) r0 += 1                      ; R0_w=-9223372036854775807
      12: (ad) if r0 < r1 goto pc-2         ; R0_w=-9223372036854775807
R1_w=scalar(umin=9223372036854775568,umax=9223372036854775809)
      13: (b7) r0 = 0                       ; R0_w=0
      14: (95) exit

      from 12 to 11: R0_w=-9223372036854775807
R1_w=scalar(umin=9223372036854775810,umax=9223372036854775823,var_off=(0x8000000000000000;
0xf),s32_min=0,s32_max=15,u32_max=15) R2_w=pkt(off=0,r=1,imm=0) R3_w=pkt_end(off=0,imm=0)
R10=fp0
      11: (07) r0 += 1                      ; R0_w=-9223372036854775806
      12: (ad) if r0 < r1 goto pc-2         ; R0_w=-9223372036854775806
R1_w=-9223372036854775806
      13: safe

      from 12 to 11: R0_w=-9223372036854775806
R1_w=scalar(umin=9223372036854775811,umax=9223372036854775823,var_off=(0x8000000000000000;
0xf),s32_min=0,s32_max=15,u32_max=15) R2_w=pkt(off=0,r=1,imm=0) R3_w=pkt_end(off=0,imm=0)
R10=fp0
      11: (07) r0 += 1                      ; R0_w=-9223372036854775805
      12: (ad) if r0 < r1 goto pc-2         ; R0_w=-9223372036854775805
R1_w=-9223372036854775805
      13: safe

      [...]

      from 12 to 11: R0_w=-9223372036854775798
R1=scalar(umin=9223372036854775819,umax=9223372036854775823,var_off=(0x8000000000000008;
0x7),s32_min=8,s32_max=15,u32_min=8,u32_max=15) R2=pkt(off=0,r=1,imm=0)
R3=pkt_end(off=0,imm=0) R10=fp0
      11: (07) r0 += 1                      ; R0_w=-9223372036854775797
      12: (ad) if r0 < r1 goto pc-2         ; R0_w=-9223372036854775797
R1=-9223372036854775797
      13: safe

      from 12 to 11: R0_w=-9223372036854775797
R1=scalar(umin=9223372036854775820,umax=9223372036854775823,var_off=(0x800000000000000c;
0x3),s32_min=12,s32_max=15,u32_min=12,u32_max=15) R2=pkt(off=0,r=1,imm=0)
R3=pkt_end(off=0,imm=0) R10=fp0
      11: (07) r0 += 1                      ; R0_w=-9223372036854775796
      12: (ad) if r0 < r1 goto pc-2         ; R0_w=-9223372036854775796
R1=-9223372036854775796
      13: safe
```

```
      from 12 to 11: R0_w=-9223372036854775796
R1=scalar(umin=9223372036854775821,umax=9223372036854775823,var_off=(0x800000000000000c;
0x3),s32_min=12,s32_max=15,u32_min=12,u32_max=15) R2=pkt(off=0,r=1,imm=0)
R3=pkt_end(off=0,imm=0) R10=fp0
      11: (07) r0 += 1                      ; R0_w=-9223372036854775795
      12: (ad) if r0 < r1 goto pc-2         ; R0_w=-9223372036854775795
R1=-9223372036854775795
      13: safe

      from 12 to 11: R0_w=-9223372036854775795
R1=scalar(umin=9223372036854775822,umax=9223372036854775823,var_off=(0x800000000000000e;
0x1),s32_min=14,s32_max=15,u32_min=14,u32_max=15) R2=pkt(off=0,r=1,imm=0)
R3=pkt_end(off=0,imm=0) R10=fp0
      11: (07) r0 += 1                      ; R0_w=-9223372036854775794
      12: (ad) if r0 < r1 goto pc-2         ; R0_w=-9223372036854775794
R1=-9223372036854775794
      13: safe

      from 12 to 11: R0_w=-9223372036854775794 R1=-9223372036854775793
R2=pkt(off=0,r=1,imm=0) R3=pkt_end(off=0,imm=0) R10=fp0
      11: (07) r0 += 1                      ; R0_w=-9223372036854775793
      12: (ad) if r0 < r1 goto pc-2
      last_idx 12 first_idx 12
      parent didn't have regs=1 stack=0 marks: R0_rw=P-9223372036854775801
R1_r=scalar(umin=9223372036854775815,umax=9223372036854775823,var_off=(0x8000000000000000;
0xf),s32_min=0,s32_max=15,u32_max=15) R2=pkt(off=0,r=1,imm=0) R3=pkt_end(off=0,imm=0)
R10=fp0
      last_idx 11 first_idx 11
      regs=1 stack=0 before 11: (07) r0 += 1
      parent didn't have regs=1 stack=0 marks: R0_rw=P-9223372036854775805
R1_rw=scalar(umin=9223372036854775812,umax=9223372036854775823,var_off=(0x8000000000000000;
0xf),s32_min=0,s32_max=15,u32_max=15) R2_w=pkt(off=0,r=1,imm=0) R3_w=pkt_end(off=0,imm=0)
R10=fp0
      last_idx 12 first_idx 0
      regs=1 stack=0 before 12: (ad) if r0 < r1 goto pc-2
      regs=1 stack=0 before 11: (07) r0 += 1
      regs=1 stack=0 before 12: (ad) if r0 < r1 goto pc-2
      regs=1 stack=0 before 11: (07) r0 += 1
      regs=1 stack=0 before 12: (ad) if r0 < r1 goto pc-2
      regs=1 stack=0 before 11: (07) r0 += 1
      regs=1 stack=0 before 9: (18) r0 = 0x8000000000000000
      last_idx 12 first_idx 12
      parent didn't have regs=2 stack=0 marks: R0_rw=P-9223372036854775801
R1_r=Pscalar(umin=9223372036854775815,umax=9223372036854775823,var_off=(0x8000000000000000;
0xf),s32_min=0,s32_max=15,u32_max=15) R2=pkt(off=0,r=1,imm=0) R3=pkt_end(off=0,imm=0)
R10=fp0
      last_idx 11 first_idx 11
      regs=2 stack=0 before 11: (07) r0 += 1
      parent didn't have regs=2 stack=0 marks: R0_rw=P-9223372036854775805
R1_rw=Pscalar(umin=9223372036854775812,umax=9223372036854775823,var_off=(0x8000000000000000;
0xf),s32_min=0,s32_max=15,u32_max=15) R2_w=pkt(off=0,r=1,imm=0) R3_w=pkt_end(off=0,imm=0)
R10=fp0
      last_idx 12 first_idx 0
      regs=2 stack=0 before 12: (ad) if r0 < r1 goto pc-2
      regs=2 stack=0 before 11: (07) r0 += 1
      regs=2 stack=0 before 12: (ad) if r0 < r1 goto pc-2
      regs=2 stack=0 before 11: (07) r0 += 1
      regs=2 stack=0 before 12: (ad) if r0 < r1 goto pc-2
      regs=2 stack=0 before 11: (07) r0 += 1
      regs=2 stack=0 before 9: (18) r0 = 0x8000000000000000
      regs=2 stack=0 before 8: (0f) r1 += r0
      regs=3 stack=0 before 6: (18) r0 = 0x7fffffffffffff10
      regs=2 stack=0 before 5: (71) r1 = *(u8 *)(r2 +0)
      13: safe
```

```
       from 4 to 13: safe
       verification time 322 usec
       stack depth 0
       processed 56 insns (limit 1000000) max_states_per_insn 1 total_states 3 peak_states 3
  mark_read 1

      This also fixes up a test case along with this improvement where we match
      on the verifier log. The updated log now has a refined var_off, too.

      Fixes: 3f50f132d840 ("bpf: Verifier, do explicit ALU32 bounds tracking")
      Reported-by: Xu Kuohai <xukuohai@huaweicloud.com>
      Signed-off-by: Daniel Borkmann <daniel@iogearbox.net>
      Signed-off-by: Andrii Nakryiko <andrii@kernel.org>
      Reviewed-by: John Fastabend <john.fastabend@gmail.com>
      Link: https://lore.kernel.org/bpf/20230314203424.4015351-2-xukuohai@huaweicloud.com
      Link: https://lore.kernel.org/bpf/20230322213056.2470-1-daniel@iogearbox.net
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 03a9de691033378ad6e25d9c67314c96420cca4f
  Author: JP Kobryn <inwardvessel@gmail.com>
  Date:   Wed Mar 22 12:47:54 2023 -0700

      bpf: return long from bpf_map_ops funcs

      [ Upstream commit d7ba4cc900bf1eea2d8c807c6b1fc6bd61f41237 ]

      This patch changes the return types of bpf_map_ops functions to long, where
      previously int was returned. Using long allows for bpf programs to maintain
      the sign bit in the absence of sign extension during situations where
      inlined bpf helper funcs make calls to the bpf_map_ops funcs and a negative
      error is returned.

      The definitions of the helper funcs are generated from comments in the bpf
      uapi header at `include/uapi/linux/bpf.h`. The return type of these
      helpers was previously changed from int to long in commit bdb7b79b4ce8. For
      any case where one of the map helpers call the bpf_map_ops funcs that are
      still returning 32-bit int, a compiler might not include sign extension
      instructions to properly convert the 32-bit negative value a 64-bit
      negative value.

      For example:
      bpf assembly excerpt of an inlined helper calling a kernel function and
      checking for a specific error:

      ; err = bpf_map_update_elem(&mymap, &key, &val, BPF_NOEXIST);
        ...
        46:   call   0xffffffffe103291c       ; htab_map_update_elem
      ; if (err && err != -EEXIST) {
        4b:   cmp    $0xffffffffffffffef,%rax ; cmp -EEXIST,%rax

      kernel function assembly excerpt of return value from
      `htab_map_update_elem` returning 32-bit int:

      movl $0xffffffef, %r9d
      ...
      movl %r9d, %eax

      ...results in the comparison:
      cmp $0xffffffffffffffef, $0x00000000ffffffef

      Fixes: bdb7b79b4ce8 ("bpf: Switch most helper return values from 32-bit int to 64-bit
  long")
      Tested-by: Eduard Zingerman <eddyz87@gmail.com>
      Signed-off-by: JP Kobryn <inwardvessel@gmail.com>
      Link: https://lore.kernel.org/r/20230322194754.185781-3-inwardvessel@gmail.com
```

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

commit db64122a367cc8129dd416368a76c4434f393f02
Author: Madhu Koriginja <madhu.koriginja@nxp.com>
Date:   Tue Mar 21 21:28:44 2023 +0530

    netfilter: keep conntrack reference until IPsecv6 policy checks are done

    [ Upstream commit b0e214d212030fe497d4d150bb3474e50ad5d093 ]

    Keep the conntrack reference until policy checks have been performed for
    IPsec V6 NAT support, just like ipv4.

    The reference needs to be dropped before a packet is
    queued to avoid having the conntrack module unloadable.

    Fixes: 58a317f1061c ("netfilter: ipv6: add IPv6 NAT support")
    Signed-off-by: Madhu Koriginja <madhu.koriginja@nxp.com>
    Signed-off-by: Florian Westphal <fw@strlen.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 60c1ea3bc619a10271943f6518383426d2a53f60
Author: Russell King (Oracle) <rmk+kernel@armlinux.org.uk>
Date:   Sun Mar 19 12:33:29 2023 +0000

    net: dsa: qca8k: remove assignment of an_enabled in pcs_get_state()

    [ Upstream commit 9ef70d0130f282638b28cfce24222f71ada00c9c ]

    pcs_get_state() implementations are not supposed to alter an_enabled.
    Remove this assignment.

    Fixes: b3591c2a3661 ("net: dsa: qca8k: Switch to PHYLINK instead of PHYLIB")
    Signed-off-by: Russell King (Oracle) <rmk+kernel@armlinux.org.uk>
    Link: https://lore.kernel.org/r/E1pdsE5-00Dl2l-8F@rmk-PC.armlinux.org.uk
    Signed-off-by: Paolo Abeni <pabeni@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9f982cbf6c83ec40acb8561266cae65680b70205
Author: Alexei Starovoitov <ast@kernel.org>
Date:   Sun Mar 19 13:30:13 2023 -0700

    libbpf: Fix ld_imm64 copy logic for ksym in light skeleton.

    [ Upstream commit a506d6ce1dd184051037dc9d26c3eb187c9fe625 ]

    Unlike normal libbpf the light skeleton 'loader' program is doing
    btf_find_by_name_kind() call at run-time to find ksym in the kernel and
    populate its {btf_id, btf_obj_fd} pair in ld_imm64 insn. To avoid doing the
    search multiple times for the same ksym it remembers the first patched ld_imm64
    insn and copies {btf_id, btf_obj_fd} from it into subsequent ld_imm64 insn.
    Fix a bug in copying logic, since it may incorrectly clear BPF_PSEUDO_BTF_ID flag.

    Also replace always true if (btf_obj_fd >= 0) check with unconditional JMP_JA
    to clarify the code.

    Fixes: d995816b77eb ("libbpf: Avoid reload of imm for weak, unresolved, repeating ksym")
    Signed-off-by: Alexei Starovoitov <ast@kernel.org>
    Signed-off-by: Andrii Nakryiko <andrii@kernel.org>
    Link: https://lore.kernel.org/bpf/20230319203014.55866-1-alexei.starovoitov@gmail.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 30c202f852a5d5e1ea63d334f26ca27499a0f44d
Author: Eric Dumazet <edumazet@google.com>
```

```
Date:   Thu Mar 16 01:10:08 2023 +0000

    net/packet: convert po->auxdata to an atomic flag

    [ Upstream commit fd53c297aa7b077ae98a3d3d2d3aa278a1686ba6 ]

    po->auxdata can be read while another thread
    is changing its value, potentially raising KCSAN splat.

    Convert it to PACKET_SOCK_AUXDATA flag.

    Fixes: 8dc419447415 ("[PACKET]: Add optional checksum computation for recvmsg")
    Signed-off-by: Eric Dumazet <edumazet@google.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit aed937d530f67679efb1d563e334061d0dba36d8
Author: Eric Dumazet <edumazet@google.com>
Date:   Thu Mar 16 01:10:07 2023 +0000

    net/packet: convert po->origdev to an atomic flag

    [ Upstream commit ee5675ecdf7a4e713ed21d98a70c2871d6ebed01 ]

    syzbot/KCAN reported that po->origdev can be read
    while another thread is changing its value.

    We can avoid this splat by converting this field
    to an actual bit.

    Following patches will convert remaining 1bit fields.

    Fixes: 80feaacb8a64 ("[AF_PACKET]: Add option to return orig_dev to userspace.")
    Signed-off-by: Eric Dumazet <edumazet@google.com>
    Reported-by: syzbot <syzkaller@googlegroups.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0c90b089585a7e6be77eff177d80947f5d8b27e1
Author: Eric Dumazet <edumazet@google.com>
Date:   Thu Mar 16 01:10:06 2023 +0000

    net/packet: annotate accesses to po->xmit

    [ Upstream commit b9d83ab8a708f23a4001d60e9d8d0b3be3d9f607 ]

    po->xmit can be set from setsockopt(PACKET_QDISC_BYPASS),
    while read locklessly.

    Use READ_ONCE()/WRITE_ONCE() to avoid potential load/store
    tearing issues.

    Fixes: d346a3fae3ff ("packet: introduce PACKET_QDISC_BYPASS socket option")
    Signed-off-by: Eric Dumazet <edumazet@google.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0027ea82057e1870c6a55f96fbcba029fe11e0d8
Author: Vadim Fedorenko <vadim.fedorenko@linux.dev>
Date:   Wed Mar 15 08:33:02 2023 -0700

    vlan: partially enable SIOCSHWTSTAMP in container

    [ Upstream commit 731b73dba359e3ff00517c13aa0daa82b34ff466 ]
```

```
    Setting timestamp filter was explicitly disabled on vlan devices in
    containers because it might affect other processes on the host. But it's
    absolutely legit in case when real device is in the same namespace.

    Fixes: 873017af7784 ("vlan: disable SIOCSHWTSTAMP in container")
    Signed-off-by: Vadim Fedorenko <vadim.fedorenko@linux.dev>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1e9f7250ffe4d391829148a8ea184fdc36d259a2
Author: Russell King (Oracle) <rmk+kernel@armlinux.org.uk>
Date:   Wed Mar 15 14:46:43 2023 +0000

    net: pcs: xpcs: remove double-read of link state when using AN

    [ Upstream commit ef63461caf427a77a04620d74ba90035a712af9c ]

    Phylink does not want the current state of the link when reading the
    PCS link state - it wants the latched state. Don't double-read the
    MII status register. Phylink will re-read as necessary to capture
    transient link-down events as of dbae3388ea9c ("net: phylink: Force
    retrigger in case of latched link-fail indicator").

    The above referenced commit is a dependency for this change, and thus
    this change should not be backported to any kernel that does not
    contain the above referenced commit.

    Fixes: fcb26bd2b6ca ("net: phy: Add Synopsys DesignWare XPCS MDIO module")
    Signed-off-by: Russell King (Oracle) <rmk+kernel@armlinux.org.uk>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f6b05f8788735722f8289439a4db5779c814c91d
Author: Luis Gerhorst <gerhorst@cs.fau.de>
Date:   Wed Mar 15 17:54:00 2023 +0100

    bpf: Remove misleading spec_v1 check on var-offset stack read

    [ Upstream commit 082cdc69a4651dd2a77539d69416a359ed1214f5 ]

    For every BPF_ADD/SUB involving a pointer, adjust_ptr_min_max_vals()
    ensures that the resulting pointer has a constant offset if
    bypass_spec_v1 is false. This is ensured by calling sanitize_check_bounds()
    which in turn calls check_stack_access_for_ptr_arithmetic(). There,
    -EACCESS is returned if the register's offset is not constant, thereby
    rejecting the program.

    In summary, an unprivileged user must never be able to create stack
    pointers with a variable offset. That is also the case, because a
    respective check in check_stack_write() is missing. If they were able
    to create a variable-offset pointer, users could still use it in a
    stack-write operation to trigger unsafe speculative behavior [1].

    Because unprivileged users must already be prevented from creating
    variable-offset stack pointers, viable options are to either remove
    this check (replacing it with a clarifying comment), or to turn it
    into a "verifier BUG"-message, also adding a similar check in
    check_stack_write() (for consistency, as a second-level defense).
    This patch implements the first option to reduce verifier bloat.

    This check was introduced by commit 01f810ace9ed ("bpf: Allow
    variable-offset stack access") which correctly notes that
    "variable-offset reads and writes are disallowed (they were already
    disallowed for the indirect access case) because the speculative
    execution checking code doesn't support them". However, it does not
```

```
    further discuss why the check in check_stack_read() is necessary.
    The code which made this check obsolete was also introduced in this
    commit.

    I have compiled ~650 programs from the Linux selftests, Linux samples,
    Cilium, and libbpf/examples projects and confirmed that none of these
    trigger the check in check_stack_read() [2]. Instead, all of these
    programs are, as expected, already rejected when constructing the
    variable-offset pointers. Note that the check in
    check_stack_access_for_ptr_arithmetic() also prints "off=%d" while the
    code removed by this patch does not (the error removed does not appear
    in the "verification_error" values). For reproducibility, the
    repository linked includes the raw data and scripts used to create
    the plot.

      [1] https://arxiv.org/pdf/1807.03757.pdf
      [2] https://gitlab.cs.fau.de/un65esoq/bpf-
spectre/-/raw/53dc19fcf459c186613b1156a81504b39c8d49db/data/plots/23-02-26_23-
56_bpftool/bpftool/0004-errors.pdf?inline=false

    Fixes: 01f810ace9ed ("bpf: Allow variable-offset stack access")
    Signed-off-by: Luis Gerhorst <gerhorst@cs.fau.de>
    Signed-off-by: Daniel Borkmann <daniel@iogearbox.net>
    Acked-by: Daniel Borkmann <daniel@iogearbox.net>
    Link: https://lore.kernel.org/bpf/20230315165358.23701-1-gerhorst@cs.fau.de
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f559d874e04ebce95f1b521c5b086ff95de09b3b
Author: David Vernet <void@manifault.com>
Date:   Thu Mar 16 00:40:24 2023 -0500

    bpf: Free struct bpf_cpumask in call_rcu handler

    [ Upstream commit 77473d1a962f3d4f7ba48324502b6d27b8ef2591 ]

    The struct bpf_cpumask type uses the bpf_mem_cache_{alloc,free}() APIs
    to allocate and free its cpumasks. The bpf_mem allocator may currently
    immediately reuse some memory when its freed, without waiting for an RCU
    read cycle to elapse. We want to be able to treat struct bpf_cpumask
    objects as completely RCU safe.

    This is necessary for two reasons:

    1. bpf_cpumask_kptr_get() currently does an RCU-protected
       refcnt_inc_not_zero(). This of course assumes that the underlying
       memory is not reused, and is therefore unsafe in its current form.

    2. We want to be able to get rid of bpf_cpumask_kptr_get() entirely, and
       intead use the superior kptr RCU semantics now afforded by the
       verifier.

    This patch fixes (1), and enables (2), by making struct bpf_cpumask RCU
    safe. A subsequent patch will update the verifier to allow struct
    bpf_cpumask * pointers to be passed to KF_RCU kfuncs, and then a latter
    patch will remove bpf_cpumask_kptr_get().

    Fixes: 516f4d3397c9 ("bpf: Enable cpumasks to be queried and used as kptrs")
    Signed-off-by: David Vernet <void@manifault.com>
    Link: https://lore.kernel.org/r/20230316054028.88924-2-void@manifault.com
    Signed-off-by: Alexei Starovoitov <ast@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0f24e540e79b3de9112a01abc5dd78a42c792fa2
Author: Hou Tao <houtao1@huawei.com>
Date:   Thu Feb 16 10:48:21 2023 +0800
```

```
    bpf: Only allocate one bpf_mem_cache for bpf_cpumask_ma

    [ Upstream commit 5d5de3a431d87ac51d43da8d796891d014975ab7 ]

    The size of bpf_cpumask is fixed, so there is no need to allocate many
    bpf_mem_caches for bpf_cpumask_ma, just one bpf_mem_cache is enough.
    Also add comments for bpf_mem_alloc_init() in bpf_mem_alloc.h to prevent
    future miuse.

    Signed-off-by: Hou Tao <houtao1@huawei.com>
    Acked-by: Jiri Olsa <jolsa@kernel.org>
    Link: https://lore.kernel.org/r/20230216024821.2202916-1-houtao@huaweicloud.com
    Signed-off-by: Alexei Starovoitov <ast@kernel.org>
    Stable-dep-of: 77473d1a962f ("bpf: Free struct bpf_cpumask in call_rcu handler")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 83b65a35d25b749ad431a3ef5ae88c33cbec3049
Author: Martin KaFai Lau <martin.lau@kernel.org>
Date:   Wed Mar 15 17:07:26 2023 -0700

    selftests/bpf: Fix a fd leak in an error path in network_helpers.c

    [ Upstream commit 226efec2b0efad60d4a6c4b2c3a8710dafc4dc21 ]

    In __start_server, it leaks a fd when setsockopt(SO_REUSEPORT) fails.
    This patch fixes it.

    Fixes: eed92afdd14c ("bpf: selftest: Test batching and bpf_(get|set)sockopt in bpf tcp
iter")
    Reported-by: Andrii Nakryiko <andrii@kernel.org>
    Signed-off-by: Martin KaFai Lau <martin.lau@kernel.org>
    Signed-off-by: Daniel Borkmann <daniel@iogearbox.net>
    Acked-by: Yonghong Song <yhs@fb.com>
    Acked-by: John Fastabend <john.fastabend@gmail.com>
    Link: https://lore.kernel.org/bpf/20230316000726.1016773-2-martin.lau@linux.dev
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a1548363582a8066edd4986f839d785f13dda3aa
Author: Aditya Kumar Singh <quic_adisi@quicinc.com>
Date:   Thu Mar 9 15:23:08 2023 +0530

    wifi: ath11k: fix deinitialization of firmware resources

    [ Upstream commit 5a78ac33e3cb8822da64dd1af196e83664b332b0 ]

    Currently, in ath11k_ahb_fw_resources_init(), iommu domain
    mapping is done only for the chipsets having fixed firmware
    memory. Also, for such chipsets, mapping is done only if it
    does not have TrustZone support.

    During deinitialization, only if TrustZone support is not there,
    iommu is unmapped back. However, for non fixed firmware memory
    chipsets, TrustZone support is not there and this makes the
    condition check to true and it tries to unmap the memory which
    was not mapped during initialization.

    This leads to the following trace -

    [   83.198790] Unable to handle kernel NULL pointer dereference at virtual address
0000000000000008
    [   83.259537] Modules linked in: ath11k_ahb ath11k qmi_helpers
    .. snip ..
    [   83.280286] pstate: 20000005 (nzCv daif -PAN -UAO -TCO -DIT -SSBS BTYPE=--)
    [   83.287228] pc : __iommu_unmap+0x30/0x140
```

```
[   83.293907] lr : iommu_unmap+0x5c/0xa4
[   83.298072] sp : ffff80000b3abad0
.. snip ..
[   83.369175] Call trace:
[   83.376282]  __iommu_unmap+0x30/0x140
[   83.378541]  iommu_unmap+0x5c/0xa4
[   83.382360]  ath11k_ahb_fw_resource_deinit.part.12+0x2c/0xac [ath11k_ahb]
[   83.385666]  ath11k_ahb_free_resources+0x140/0x17c [ath11k_ahb]
[   83.392521]  ath11k_ahb_shutdown+0x34/0x40 [ath11k_ahb]
[   83.398248]  platform_shutdown+0x20/0x2c
[   83.403455]  device_shutdown+0x16c/0x1c4
[   83.407621]  kernel_restart_prepare+0x34/0x3c
[   83.411529]  kernel_restart+0x14/0x74
[   83.415781]  __do_sys_reboot+0x1c4/0x22c
[   83.419427]  __arm64_sys_reboot+0x1c/0x24
[   83.423420]  invoke_syscall+0x44/0xfc
[   83.427326]  el0_svc_common.constprop.3+0xac/0xe8
[   83.430974]  do_el0_svc+0xa0/0xa8
[   83.435659]  el0_svc+0x1c/0x44
[   83.438957]  el0t_64_sync_handler+0x60/0x144
[   83.441910]  el0t_64_sync+0x15c/0x160
[   83.446343] Code: aa0103f4 f9400001 f90027a1 d2800001 (f94006a0)
[   83.449903] ---[ end trace 0000000000000000 ]---
```

    This can be reproduced by probing an AHB chipset which is not
    having a fixed memory region. During reboot (or rmmod) trace
    can be seen.

    Fix this issue by adding a condition check on firmware fixed memory
    hw_param as done in the counter initialization function.

    Tested-on: IPQ8074 hw2.0 AHB WLAN.HK.2.7.0.1-01744-QCAHKSWPL_SILICONZ-1

    Fixes: f9eec4947add ("ath11k: Add support for targets without trustzone")
    Signed-off-by: Aditya Kumar Singh <quic_adisi@quicinc.com>
    Signed-off-by: Kalle Valo <quic_kvalo@quicinc.com>
    Link: https://lore.kernel.org/r/20230309095308.24937-1-quic_adisi@quicinc.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e47fd05744a94114da21362b1a3f89e877684274
Author: Harshit Mogalapalli <harshit.m.mogalapalli@oracle.com>
Date: Tue Mar 7 02:47:06 2023 -0800

    wifi: ath12k: Add missing unwind goto in ath12k_pci_probe()

    [Upstream commit 488d9a484f96eee4f0e8e108aed42a057a1c7295]

    Smatch Warns:
            drivers/net/wireless/ath/ath12k/pci.c:1198 ath12k_pci_probe()
            warn: missing unwind goto?

    Store the error value in ret and use correct label with a goto.

    Only Compile tested, found with Smatch.

    Fixes: d889913205cf ("wifi: ath12k: driver for Qualcomm Wi-Fi 7 devices")
    Reported-by: Dan Carpenter <error27@gmail.com>
    Link: https://lore.kernel.org/all/Y+426q6cfkEdb5Bv@kili/
    Suggested-by: Dan Carpenter <error27@gmail.com>
    Signed-off-by: Harshit Mogalapalli <harshit.m.mogalapalli@oracle.com>
    Reviewed-by: Simon Horman <simon.horman@corigine.com>
    Signed-off-by: Kalle Valo <quic_kvalo@quicinc.com>
    Link: https://lore.kernel.org/r/20230307104706.240119-1-harshit.m.mogalapalli@oracle.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

```
commit c16f30f802a4b572461b95c54bfbc359ef5a19fe
Author: Alexander Mikhalitsyn <alexander@mihalicyn.com>
Date:   Mon Mar 13 12:32:11 2023 +0100

    scm: fix MSG_CTRUNC setting condition for SO_PASSSEC

    [ Upstream commit a02d83f9947d8f71904eda4de046630c3eb6802c ]

    Currently, kernel would set MSG_CTRUNC flag if msg_control buffer
    wasn't provided and SO_PASSCRED was set or if there was pending SCM_RIGHTS.

    For some reason we have no corresponding check for SO_PASSSEC.

    In the recvmsg(2) doc we have:
           MSG_CTRUNC
                  indicates that some control data was discarded due to lack
                  of space in the buffer for ancillary data.

    So, we need to set MSG_CTRUNC flag for all types of SCM.

    This change can break applications those don't check MSG_CTRUNC flag.

    Cc: "David S. Miller" <davem@davemloft.net>
    Cc: Eric Dumazet <edumazet@google.com>
    Cc: Jakub Kicinski <kuba@kernel.org>
    Cc: Paolo Abeni <pabeni@redhat.com>
    Cc: Leon Romanovsky <leon@kernel.org>
    Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
    Signed-off-by: Alexander Mikhalitsyn <aleksandr.mikhalitsyn@canonical.com>

    v2:
    - commit message was rewritten according to Eric's suggestion
    Acked-by: Paul Moore <paul@paul-moore.com>

    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 114dfdda3cf405bbcf40f59063c079552f37be3d
Author: Shashank Gupta <shashank.gupta@intel.com>
Date:   Mon Feb 27 15:55:42 2023 -0500

    crypto: qat - fix concurrency issue when device state changes

    [ Upstream commit 1bdc85550a2b59bb7f62ead7173134e66dd2d60e ]

    The sysfs `state` attribute is not protected against race conditions.
    If multiple processes perform a device state transition on the same
    device in parallel, unexpected behaviors might occur.

    For transitioning the device state, adf_sysfs.c calls the functions
    adf_dev_init(), adf_dev_start(), adf_dev_stop() and adf_dev_shutdown()
    which are unprotected and interdependent on each other. To perform a
    state transition, these functions needs to be called in a specific
    order:
      * device up:   adf_dev_init() -> adf_dev_start()
      * device down: adf_dev_stop() -> adf_dev_shutdown()

    This change introduces the functions adf_dev_up() and adf_dev_down()
    which wrap the state machine functions and protect them with a
    per-device lock. These are then used in adf_sysfs.c instead of the
    individual state transition functions.

    Fixes: 5ee52118ac14 ("crypto: qat - expose device state through sysfs for 4xxx")
    Signed-off-by: Shashank Gupta <shashank.gupta@intel.com>
    Reviewed-by: Giovanni Cabiddu <giovanni.cabiddu@intel.com>
```

```
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9b784428bb8d7e06d3a4f098e33839521c413df8
Author: Andrii Nakryiko <andrii@kernel.org>
Date:   Mon Mar 13 11:40:17 2023 -0700

    bpf: fix precision propagation verbose logging

    [ Upstream commit 34f0677e7afd3a292bc1aadda7ce8e35faedb204 ]

    Fix wrong order of frame index vs register/slot index in precision
    propagation verbose (level 2) output. It's wrong and very confusing as is.

    Fixes: 529409ea92d5 ("bpf: propagate precision across all frames, not just the last
one")
    Signed-off-by: Andrii Nakryiko <andrii@kernel.org>
    Link: https://lore.kernel.org/r/20230313184017.4083374-1-andrii@kernel.org
    Signed-off-by: Alexei Starovoitov <ast@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7d8d66d93d5e6160a3525b2eb3068c0f5893bc41
Author: Andrii Nakryiko <andrii@kernel.org>
Date:   Thu Mar 9 14:41:31 2023 -0800

    bpf: take into account liveness when propagating precision

    [ Upstream commit 52c2b005a3c18c565fc70cfd0ca49375f301e952 ]

    When doing state comparison, if old state has register that is not
    marked as REG_LIVE_READ, then we just skip comparison, regardless what's
    the state of corresponing register in current state. This is because not
    REG_LIVE_READ register is irrelevant for further program execution and
    correctness. All good here.

    But when we get to precision propagation, after two states were declared
    equivalent, we don't take into account old register's liveness, and thus
    attempt to propagate precision for register in current state even if
    that register in old state was not REG_LIVE_READ anymore. This is bad,
    because register in current state could be anything at all and this
    could cause -EFAULT due to internal logic bugs.

    Fix by taking into account REG_LIVE_READ liveness mark to keep the logic
    in state comparison in sync with precision propagation.

    Fixes: a3ce685dd01a ("bpf: fix precision tracking")
    Signed-off-by: Andrii Nakryiko <andrii@kernel.org>
    Link: https://lore.kernel.org/r/20230309224131.57449-1-andrii@kernel.org
    Signed-off-by: Alexei Starovoitov <ast@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5556d92fb2952d86d6ec1d087f34659931adf0fa
Author: Martin KaFai Lau <martin.lau@kernel.org>
Date:   Wed Mar 8 22:02:44 2023 -0800

    selftests/bpf: Fix flaky fib_lookup test

    [ Upstream commit a6865576317f6249f3f83cf4c10ab56e627ee153 ]

    There is a report that fib_lookup test is flaky when running in parallel.
    A symptom of slowness or delay. An example:

    Testing IPv6 stale neigh
    set_lookup_params:PASS:inet_pton(IPV6_IFACE_ADDR) 0 nsec
    test_fib_lookup:PASS:bpf_prog_test_run_opts 0 nsec
```

```
    test_fib_lookup:FAIL:fib_lookup_ret unexpected fib_lookup_ret: actual 0 != expected 7
    test_fib_lookup:FAIL:dmac not match unexpected dmac not match: actual 1 != expected 0
    dmac expected 11:11:11:11:11:11 actual 00:00:00:00:00:00

    [ Note that the "fib_lookup_ret unexpected fib_lookup_ret actual 0 ..."
      is reversed in terms of expected and actual value. Fixing in this
      patch also. ]

    One possibility is the testing stale neigh entry was marked dead by the
    gc (in neigh_periodic_work). The default gc_stale_time sysctl is 60s.
    This patch increases it to 15 mins.

    It also:

    - fixes the reversed arg (actual vs expected) in one of the
      ASSERT_EQ test
    - removes the nodad command arg when adding v4 neigh entry which
      currently has a warning.

    Fixes: 168de0233586 ("selftests/bpf: Add bpf_fib_lookup test")
    Reported-by: Alexei Starovoitov <ast@kernel.org>
    Signed-off-by: Martin KaFai Lau <martin.lau@kernel.org>
    Signed-off-by: Daniel Borkmann <daniel@iogearbox.net>
    Link: https://lore.kernel.org/bpf/20230309060244.3242491-1-martin.lau@linux.dev
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 30f99c1447f92a36ebd20895ff8cc5b30dc000c7
Author: Hangbin Liu <liuhangbin@gmail.com>
Date:   Fri Feb 24 14:13:42 2023 +0800

    selftests/bpf: move SYS() macro into the test_progs.h

    [ Upstream commit b61987d37cbee3c44e80304598c60b163553926b ]

    A lot of tests defined SYS() macro to run system calls with goto label.
    Let's move this macro to test_progs.h and add configurable
    "goto_label" as the first arg.

    Suggested-by: Martin KaFai Lau <martin.lau@linux.dev>
    Signed-off-by: Hangbin Liu <liuhangbin@gmail.com>
    Link: https://lore.kernel.org/r/20230224061343.506571-2-liuhangbin@gmail.com
    Signed-off-by: Martin KaFai Lau <martin.lau@kernel.org>
    Stable-dep-of: a6865576317f ("selftests/bpf: Fix flaky fib_lookup test")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2157b8122ffe9d620a69a3e740dc3fb6fd4492c9
Author: Roberto Sassu <roberto.sassu@huawei.com>
Date:   Wed Mar 8 11:37:13 2023 +0100

    selftests/bpf: Fix IMA test

    [ Upstream commit 12fabae03ca6474fd571bf6ddb37d009533305d6 ]

    Commit 62622dab0a28 ("ima: return IMA digest value only when IMA_COLLECTED
    flag is set") caused bpf_ima_inode_hash() to refuse to give non-fresh
    digests. IMA test #3 assumed the old behavior, that bpf_ima_inode_hash()
    still returned also non-fresh digests.

    Correct the test by accepting both cases. If the samples returned are 1,
    assume that the commit above is applied and that the returned digest is
    fresh. If the samples returned are 2, assume that the commit above is not
    applied, and check both the non-fresh and fresh digest.

    Fixes: 62622dab0a28 ("ima: return IMA digest value only when IMA_COLLECTED flag is set")
    Reported-by: David Vernet <void@manifault.com>
```

```
    Signed-off-by: Roberto Sassu <roberto.sassu@huawei.com>
    Signed-off-by: Andrii Nakryiko <andrii@kernel.org>
    Reviewed-by: Matt Bobrowski <mattbobrowski@google.com>
    Link: https://lore.kernel.org/bpf/20230308103713.1681200-1-roberto.sassu@huaweicloud.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5252c7156e096710fb34f8505bb630b5905bcad0
Author: Martin Blumenstingl <martin.blumenstingl@googlemail.com>
Date:   Sun Feb 26 23:10:04 2023 +0100

    wifi: rtw88: mac: Return the original error from rtw_mac_power_switch()

    [ Upstream commit 15c8e267dfa62f207ee1db666c822324e3362b84 ]

    rtw_mac_power_switch() calls rtw_pwr_seq_parser() which can return
    -EINVAL, -EBUSY or 0. Propagate the original error code instead of
    unconditionally returning -EINVAL in case of an error.

    Fixes: e3037485c68e ("rtw88: new Realtek 802.11ac driver")
    Signed-off-by: Martin Blumenstingl <martin.blumenstingl@googlemail.com>
    Reviewed-by: Ping-Ke Shih <pkshih@realtek.com>
    Signed-off-by: Kalle Valo <kvalo@kernel.org>
    Link: https://lore.kernel.org/r/20230226221004.138331-3-
martin.blumenstingl@googlemail.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e61a4ded9a28d873c7a5b9914ba0fbc717c217b8
Author: Martin Blumenstingl <martin.blumenstingl@googlemail.com>
Date:   Sun Feb 26 23:10:03 2023 +0100

    wifi: rtw88: mac: Return the original error from rtw_pwr_seq_parser()

    [ Upstream commit b7ed9fa2cb76ca7a3c3cd4a6d35748fe1fbda9f6 ]

    rtw_pwr_seq_parser() calls rtw_sub_pwr_seq_parser() which can either
    return -EBUSY, -EINVAL or 0. Propagate the original error code instead
    of unconditionally returning -EBUSY in case of an error.

    Fixes: e3037485c68e ("rtw88: new Realtek 802.11ac driver")
    Signed-off-by: Martin Blumenstingl <martin.blumenstingl@googlemail.com>
    Reviewed-by: Ping-Ke Shih <pkshih@realtek.com>
    Signed-off-by: Kalle Valo <kvalo@kernel.org>
    Link: https://lore.kernel.org/r/20230226221004.138331-2-
martin.blumenstingl@googlemail.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2df287ef7bb53eb23a1ff52687e3600308d41e0a
Author: Yonghong Song <yhs@fb.com>
Date:   Mon Feb 27 14:49:43 2023 -0800

    libbpf: Fix bpf_xdp_query() in old kernels

    [ Upstream commit c8ee37bde4021a275d2e4f33bd48d54912bb00c4 ]

    Commit 04d58f1b26a4("libbpf: add API to get XDP/XSK supported features")
    added feature_flags to struct bpf_xdp_query_opts. If a user uses
    bpf_xdp_query_opts with feature_flags member, the bpf_xdp_query()
    will check whether 'netdev' family exists or not in the kernel.
    If it does not exist, the bpf_xdp_query() will return -ENOENT.

    But 'netdev' family does not exist in old kernels as it is
    introduced in the same patch set as Commit 04d58f1b26a4.
    So old kernel with newer libbpf won't work properly with
    bpf_xdp_query() api call.
```

```
    To fix this issue, if the return value of
    libbpf_netlink_resolve_genl_family_id() is -ENOENT, bpf_xdp_query()
    will just return 0, skipping the rest of xdp feature query.
    This preserves backward compatibility.

    Fixes: 04d58f1b26a4 ("libbpf: add API to get XDP/XSK supported features")
    Signed-off-by: Yonghong Song <yhs@fb.com>
    Signed-off-by: Andrii Nakryiko <andrii@kernel.org>
    Link: https://lore.kernel.org/bpf/20230227224943.1153459-1-yhs@fb.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e08209df801c47763aecab2221159a6bb1be4849
Author: Puranjay Mohan <puranjay12@gmail.com>
Date:   Thu Feb 23 09:53:46 2023 +0000

    libbpf: Fix arm syscall regs spec in bpf_tracing.h

    [ Upstream commit 06943ae675945c762bb8d5edc93d203f2b041d8d ]

    The syscall register definitions for ARM in bpf_tracing.h doesn't define
    the fifth parameter for the syscalls. Because of this some KPROBES based
    selftests fail to compile for ARM architecture.

    Define the fifth parameter that is passed in the R5 register (uregs[4]).

    Fixes: 3a95c42d65d5 ("libbpf: Define arm syscall regs spec in bpf_tracing.h")
    Signed-off-by: Puranjay Mohan <puranjay12@gmail.com>
    Signed-off-by: Andrii Nakryiko <andrii@kernel.org>
    Link: https://lore.kernel.org/bpf/20230223095346.10129-1-puranjay12@gmail.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ae968606102d7e4d1ea30a0aa277611bf665c63a
Author: Luis Gerhorst <gerhorst@cs.fau.de>
Date:   Mon Feb 27 16:08:54 2023 +0100

    tools: bpftool: Remove invalid \' json escape

    [ Upstream commit c679bbd611c08b0559ffae079330bc4e5574696a ]

    RFC8259 ("The JavaScript Object Notation (JSON) Data Interchange
    Format") only specifies \", \\, \/, \b, \f, \n, \r, and \r as valid
    two-character escape sequences. This does not include \', which is not
    required in JSON because it exclusively uses double quotes as string
    separators.

    Solidus (/) may be escaped, but does not have to. Only reverse
    solidus (\), double quotes ("), and the control characters have to be
    escaped. Therefore, with this fix, bpftool correctly supports all valid
    two-character escape sequences (but still does not support characters
    that require multi-character escape sequences).

    Witout this fix, attempting to load a JSON file generated by bpftool
    using Python 3.10.6's default json.load() may fail with the error
    "Invalid \escape" if the file contains the invalid escaped single
    quote (\').

    Fixes: b66e907cfee2 ("tools: bpftool: copy JSON writer from iproute2 repository")
    Signed-off-by: Luis Gerhorst <gerhorst@cs.fau.de>
    Signed-off-by: Andrii Nakryiko <andrii@kernel.org>
    Reviewed-by: Quentin Monnet <quentin@isovalent.com>
    Link: https://lore.kernel.org/bpf/20230227150853.16863-1-gerhorst@cs.fau.de
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cbec770521ebc455c9811a23222faf8911422d4a
Author: Fedor Pchelkin <pchelkin@ispras.ru>
```

```
Date:   Fri Feb 24 12:28:05 2023 +0200

    wifi: ath6kl: reduce WARN to dev_dbg() in callback

    [ Upstream commit 75c4a8154cb6c7239fb55d5550f481f6765fb83c ]

    The warn is triggered on a known race condition, documented in the code above
    the test, that is correctly handled.  Using WARN() hinders automated testing.
    Reducing severity.

    Fixes: de2070fc4aa7 ("ath6kl: Fix kernel panic on continuous driver load/unload")
    Reported-and-tested-by: syzbot+555908813b2ea35dae9a@syzkaller.appspotmail.com
    Signed-off-by: Oliver Neukum <oneukum@suse.com>
    Signed-off-by: Fedor Pchelkin <pchelkin@ispras.ru>
    Signed-off-by: Alexey Khoroshilov <khoroshilov@ispras.ru>
    Signed-off-by: Kalle Valo <quic_kvalo@quicinc.com>
    Link: https://lore.kernel.org/r/20230126182431.867984-1-pchelkin@ispras.ru
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 527e332daba9dd24140b270cf1b4e2643c6104ff
Author: John Keeping <john@metanate.com>
Date:   Tue Jan 24 10:42:48 2023 +0000

    wifi: brcmfmac: support CQM RSSI notification with older firmware

    [ Upstream commit ec52d77d077529f198fd874c550a26b9cc86a331 ]

    Using the BCM4339 firmware from linux-firmware (version "BCM4339/2 wl0:
    Sep  5 2019 11:05:52 version 6.37.39.113 (r722271 CY)" from
    cypress/cyfmac4339-sdio.bin) the RSSI respose is only 4 bytes, which
    results in an error being logged.

    It seems that older devices send only the RSSI field and neither SNR nor
    noise is included.  Handle this by accepting a 4 byte message and
    reading only the RSSI from it.

    Fixes: 7dd56ea45a66 ("brcmfmac: add support for CQM RSSI notifications")
    Signed-off-by: John Keeping <john@metanate.com>
    Signed-off-by: Kalle Valo <kvalo@kernel.org>
    Link: https://lore.kernel.org/r/20230124104248.2917465-1-john@metanate.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3f15d8e41ccda157b54e1d7914c8f152acf7a88e
Author: Bitterblue Smith <rtl8821cerfe2@gmail.com>
Date:   Fri Feb 17 20:49:12 2023 +0200

    wifi: rtl8xxxu: Remove always true condition in rtl8xxxu_print_chipinfo

    [ Upstream commit b9b1e4fe2957f361c86e288ecf373dc7895cf7c7 ]

    Fix a new smatch warning:
    drivers/net/wireless/realtek/rtl8xxxu/rtl8xxxu_core.c:1580 rtl8xxxu_print_chipinfo()
warn: always true condition '(priv->chip_cut <= 15) => (0-15 <= 15)'

    Reported-by: kernel test robot <lkp@intel.com>
    Link: https://lore.kernel.org/oe-kbuild-all/202302140753.71IgU77A-lkp@intel.com/
    Fixes: 7b0ac469e331 ("wifi: rtl8xxxu: Recognise all possible chip cuts")
    Signed-off-by: Bitterblue Smith <rtl8821cerfe2@gmail.com>
    Reviewed-by: Ping-Ke Shih <pkshih@realtek.com>
    Signed-off-by: Kalle Valo <kvalo@kernel.org>
    Link: https://lore.kernel.org/r/68eff98b-a022-5a00-f330-adf623a35772@gmail.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5d7aeb3e27e0f43f2d9cfc6008098f378edda1b2
Author: Christian Marangi <ansuelsmth@gmail.com>
```

```
Date:   Thu Feb 9 23:26:22 2023 +0100

    wifi: ath11k: fix SAC bug on peer addition with sta band migration

    [ Upstream commit 60b7d62ba8cdbd073997bff0f1cdae8d844002c0 ]

    Fix sleep in atomic context warning detected by Smatch static checker
    analyzer.

    Following the locking pattern for peer_rhash_add lock tbl_mtx_lock mutex
    always even if sta is not transitioning to another band.
    This is peer_add function and a more secure locking should not cause
    performance regression.

    Tested-on: IPQ8074 hw2.0 AHB WLAN.HK.2.5.0.1-01208-QCAHKSWPL_SILICONZ-1

    Fixes: d673cb6fe6c0 ("wifi: ath11k: fix peer addition/deletion error on sta band
migration")
    Reported-by: Dan Carpenter <error27@gmail.com>
    Signed-off-by: Christian Marangi <ansuelsmth@gmail.com>
    Signed-off-by: Kalle Valo <quic_kvalo@quicinc.com>
    Link: https://lore.kernel.org/r/20230209222622.1751-1-ansuelsmth@gmail.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f6e2ffbe4e6f90184f640384ca7a111a1ab553ce
Author: Dan Carpenter <error27@gmail.com>
Date:   Mon Feb 6 16:15:48 2023 +0300

    wifi: ath5k: fix an off by one check in ath5k_eeprom_read_freq_list()

    [ Upstream commit 4c856ee12df85aabd437c3836ed9f68d94268358 ]

    This loop checks that i < max at the start of loop but then it does
    i++ which could put it past the end of the array.  It's harmless to
    check again and prevent a potential out of bounds.

    Fixes: 1048643ea94d ("ath5k: Clean up eeprom parsing and add missing calibration data")
    Signed-off-by: Dan Carpenter <error27@gmail.com>
    Reviewed-by: Luis Chamberlain <mcgrof@kernel.org>
    Signed-off-by: Kalle Valo <quic_kvalo@quicinc.com>
    Link: https://lore.kernel.org/r/Y+D9hPQrHfWBJhXz@kili
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit bbe9043d8fd69863f00d941f0a445c5b96595252
Author: Douglas Anderson <dianders@chromium.org>
Date:   Wed Feb 1 08:54:43 2023 -0800

    wifi: ath5k: Use platform_get_irq() to get the interrupt

    [ Upstream commit 95c95251d0547b46d6571e4fbd51b42865c15a4a ]

    As of commit a1a2b7125e10 ("of/platform: Drop static setup of IRQ
    resource from DT core"), we need to use platform_get_irq() instead of
    platform_get_resource() to get our IRQs because
    platform_get_resource() simply won't get them anymore.

    This was already fixed in several other Atheros WiFi drivers,
    apparently in response to Zeal Robot reports. An example of another
    fix is commit 9503a1fc123d ("ath9k: Use platform_get_irq() to get the
    interrupt"). ath5k seems to have been missed in this effort, though.

    Fixes: a1a2b7125e10 ("of/platform: Drop static setup of IRQ resource from DT core")
    Signed-off-by: Douglas Anderson <dianders@chromium.org>
    Reviewed-by: Lad Prabhakar <prabhakar.mahadev-lad.rj@bp.renesas.com>
    Signed-off-by: Kalle Valo <quic_kvalo@quicinc.com>
```

```
    Link:
https://lore.kernel.org/r/20230201084131.v2.2.Ic4f8542b0588d7eb4bc6e322d4af3d2064e84ff0@chan
geid
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b67d234b6985e0c9799699cefd0c886c9e2a4578
Author: Douglas Anderson <dianders@chromium.org>
Date:   Wed Feb 1 08:54:42 2023 -0800

    wifi: ath11k: Use platform_get_irq() to get the interrupt

    [ Upstream commit f117276638b7600b981b3fe28550823cfbe1ef23 ]

    As of commit a1a2b7125e10 ("of/platform: Drop static setup of IRQ
    resource from DT core"), we need to use platform_get_irq() instead of
    platform_get_resource() to get our IRQs because
    platform_get_resource() simply won't get them anymore.

    This was already fixed in several other Atheros WiFi drivers,
    apparently in response to Zeal Robot reports. An example of another
    fix is commit 9503a1fc123d ("ath9k: Use platform_get_irq() to get the
    interrupt"). ath11k seems to have been missed in this effort, though.

    Without this change, WiFi wasn't coming up on my Qualcomm sc7280-based
    hardware. Specifically, "platform_get_resource(pdev, IORESOURCE_IRQ,
    i)" was failing even for i=0.

    Tested-on: WCN6750 hw1.0 AHB WLAN.MSL.1.0.1-00887-QCAMSLSWPLZ-1

    Fixes: a1a2b7125e10 ("of/platform: Drop static setup of IRQ resource from DT core")
    Fixes: 00402f49d26f ("ath11k: Add support for WCN6750 device")
    Signed-off-by: Douglas Anderson <dianders@chromium.org>
    Tested-by: Jun Yu <junyuu@chromium.org>
    Reviewed-by: Lad Prabhakar <prabhakar.mahadev-lad.rj@bp.renesas.com>
    Signed-off-by: Kalle Valo <quic_kvalo@quicinc.com>
    Link:
https://lore.kernel.org/r/20230201084131.v2.1.I69cf3d56c97098287fe3a70084ee515098390b70@chan
geid
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8f02d538878c9b1501f624595eb22ee4e5e0ff84
Author: Fedor Pchelkin <pchelkin@ispras.ru>
Date:   Thu Feb 16 22:23:01 2023 +0300

    wifi: ath9k: hif_usb: fix memory leak of remain_skbs

    [ Upstream commit 7654cc03eb699297130b693ec34e25f77b17c947 ]

    hif_dev->remain_skb is allocated and used exclusively in
    ath9k_hif_usb_rx_stream(). It is implied that an allocated remain_skb is
    processed and subsequently freed (in error paths) only during the next
    call of ath9k_hif_usb_rx_stream().

    So, if the urbs are deallocated between those two calls due to the device
    deinitialization or suspend, it is possible that ath9k_hif_usb_rx_stream()
    is not called next time and the allocated remain_skb is leaked. Our local
    Syzkaller instance was able to trigger that.

    remain_skb makes sense when receiving two consecutive urbs which are
    logically linked together, i.e. a specific data field from the first skb
    indicates a cached skb to be allocated, memcpy'd with some data and
    subsequently processed in the next call to ath9k_hif_usb_rx_stream(). Urbs
    deallocation supposedly makes that link irrelevant so we need to free the
    cached skb in those cases.
```

```
    Fix the leak by introducing a function to explicitly free remain_skb (if
    it is not NULL) when the rx urbs have been deallocated. remain_skb is NULL
    when it has not been allocated at all (hif_dev struct is kzalloced) or
    when it has been processed in next call to ath9k_hif_usb_rx_stream().

    Found by Linux Verification Center (linuxtesting.org) with Syzkaller.

    Fixes: fb9987d0f748 ("ath9k_htc: Support for AR9271 chipset.")
    Signed-off-by: Fedor Pchelkin <pchelkin@ispras.ru>
    Signed-off-by: Alexey Khoroshilov <khoroshilov@ispras.ru>
    Acked-by: Toke Høiland-Jørgensen <toke@toke.dk>
    Signed-off-by: Kalle Valo <quic_kvalo@quicinc.com>
    Link: https://lore.kernel.org/r/20230216192301.171225-1-pchelkin@ispras.ru
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8bb238cf4e90fb6f8b8c76ed165ed3095b651b2d
Author: Yang Yingliang <yangyingliang@huawei.com>
Date:   Fri Feb 17 11:00:31 2023 +0800

    wifi: ath11k: fix return value check in ath11k_ahb_probe()

    [ Upstream commit 342fcde9d91460f01f65707e16368a1571271a3a ]

    ioremap() returns NULL pointer not PTR_ERR() when it fails,
    so replace the IS_ERR() check with NULL pointer check.

    Fixes: b42b3678c91f ("wifi: ath11k: remap ce register space for IPQ5018")
    Signed-off-by: Yang Yingliang <yangyingliang@huawei.com>
    Signed-off-by: Kalle Valo <quic_kvalo@quicinc.com>
    Link: https://lore.kernel.org/r/20230217030031.4021289-1-yangyingliang@huawei.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b2101386bd9535e52eae64ea809bbf5abbc99b07
Author: Dan Carpenter <error27@gmail.com>
Date:   Thu Feb 16 15:16:14 2023 +0300

    wifi: ath12k: use kfree_skb() instead of kfree()

    [ Upstream commit 8c464d16809fa02982f6341ea598ec5d07457f19 ]

    Sk_buffs are supposed to be freed with kfree_skb().

    Fixes: d889913205cf ("wifi: ath12k: driver for Qualcomm Wi-Fi 7 devices")
    Signed-off-by: Dan Carpenter <error27@gmail.com>
    Signed-off-by: Kalle Valo <quic_kvalo@quicinc.com>
    Link: https://lore.kernel.org/r/Y+4ejiYakhEvEw7c@kili
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 32e68818b4f8ea96a74c49c48f90cbe62a1c8cc6
Author: Alexey V. Vissarionov <gremlin@altlinux.org>
Date:   Wed Feb 15 20:31:37 2023 +0200

    wifi: ath6kl: minor fix for allocation size

    [ Upstream commit 778f83f889e7fca37780d9640fcbd0229ae38eaa ]

    Although the "param" pointer occupies more or equal space compared
    to "*param", the allocation size should use the size of variable
    itself.

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

    Fixes: bdcd81707973cf8a ("Add ath6kl cleaned up driver")
    Signed-off-by: Alexey V. Vissarionov <gremlin@altlinux.org>
    Signed-off-by: Kalle Valo <quic_kvalo@quicinc.com>
```

```
    Link: https://lore.kernel.org/r/20230117110414.GC12547@altlinux.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 200bcba30219f12bf7672b7af898ccb649d6a89f
Author: Sakari Ailus <sakari.ailus@linux.intel.com>
Date:   Wed Mar 29 15:57:50 2023 +0100

    media: ov5670: Fix probe on ACPI

    [ Upstream commit 73b41dc51fbeffa4a216b20193274cfe92b5d95b ]

    devm_clk_get() will return either an error or NULL, which the driver
    handles, continuing to use the clock of reading the value of the
    clock-frequency property.

    However, the value of ov5670->xvclk is left as-is and the other clock
    framework functions aren't capable of handling error values.

    Use devm_clk_get_optional() to obtain NULL instead of -ENOENT.

    Fixes: 8004c91e2095 ("media: i2c: ov5670: Use common clock framework")
    Signed-off-by: Sakari Ailus <sakari.ailus@linux.intel.com>
    Reviewed-by: Jacopo Mondi <jacopo.mondi@ideasonboard.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a218b5ffaab5cb283682b262a9676d06607c1d63
Author: Liang He <windhl@126.com>
Date:   Wed Mar 22 12:16:57 2023 +0800

    platform/chrome: cros_typec_switch: Add missing fwnode_handle_put()

    [ Upstream commit dc70234c408c644505a24362b0f095f713e4697e ]

    In cros_typec_register_switches(), we should add fwnode_handle_put()
    when break out of the iteration device_for_each_child_node()
    as it will automatically increase and decrease the refcounter.

    Fixes: affc804c44c8 ("platform/chrome: cros_typec_switch: Add switch driver")
    Signed-off-by: Liang He <windhl@126.com>
    Link: https://lore.kernel.org/r/20230322041657.1857001-1-windhl@126.com
    Signed-off-by: Prashant Malani <pmalani@chromium.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 984f880040765c1de331fe4abc838d3011ecec43
Author: Tomáš Pecka <tomas.pecka@cesnet.cz>
Date:   Thu Apr 20 19:19:39 2023 +0200

    hwmon: (pmbus/fsp-3y) Fix functionality bitmask in FSP-3Y YM-2151E

    [ Upstream commit 93822f5161a2dc57a60b95b35b3cb8589f53413e ]

    The bit flags in pmbus_driver_info functionality for YM-2151E chip were
    joined with a comma operator instead of a bitwise OR. This means that
    the last constant PMBUS_HAVE_IIN was not OR-ed with the other
    PM_BUS_HAVE_* constants for this page but it initialized the next element
    of the func array (which was not accessed from anywhere because of the
    number of pages).

    However, there is no need for setting PMBUS_HAVE_IIN in the 5Vsb page
    because this command does not seem to be paged. Obviously, the device
    only has one IIN sensor, so it doesn't make sense to query it again from
    the second page.

    Fixes: 1734b4135a62 ("hwmon: Add driver for fsp-3y PSUs and PDUs")
```

```
    Signed-off-by: Jan Kundrát <jan.kundrat@cesnet.cz>
    Signed-off-by: Tomáš Pecka <tomas.pecka@cesnet.cz>
    Link: https://lore.kernel.org/r/20230420171939.212040-1-tomas.pecka@cesnet.cz
    Signed-off-by: Guenter Roeck <linux@roeck-us.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 07aa833d9c7a170d2bea386b898470b610ccafba
Author: Bjorn Andersson <quic_bjorande@quicinc.com>
Date:   Tue Apr 18 09:30:17 2023 -0700

    rpmsg: glink: Propagate TX failures in intentless mode as well

    [ Upstream commit 7a68f9fa97357a0f2073c9c31ed4101da4fce93e ]

    As support for splitting transmission over several messages using
    TX_DATA_CONT was introduced it does not immediately return the return
    value of qcom_glink_tx().

    The result is that in the intentless case (i.e. intent == NULL), the
    code will continue to send all additional chunks. This is wasteful, and
    it's possible that the send operation could incorrectly indicate
    success, if the last chunk fits in the TX fifo.

    Fix the condition.

    Fixes: 8956927faed3 ("rpmsg: glink: Add TX_DATA_CONT command while sending")
    Reviewed-by: Chris Lew <quic_clew@quicinc.com>
    Signed-off-by: Bjorn Andersson <quic_bjorande@quicinc.com>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230418163018.785524-2-quic_bjorande@quicinc.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7acdb2826712a29371b4c9c428acc93cc3c9cbb5
Author: Sanjay Chandrashekara <sanjayc@nvidia.com>
Date:   Tue Apr 18 17:04:54 2023 +0530

    cpufreq: use correct unit when verify cur freq

    [ Upstream commit 44295af5019f1997d038ad2611086a2d1e2af167 ]

    cpufreq_verify_current_freq checks() if the frequency returned by
    the hardware has a slight delta with the valid frequency value
    last set and returns "policy->cur" if the delta is within "1 MHz".
    In the comparison, "policy->cur" is in "kHz" but it's compared
    against HZ_PER_MHZ. So, the comparison range becomes "1 GHz".

    Fix this by comparing against KHZ_PER_MHZ instead of HZ_PER_MHZ.

    Fixes: f55ae08c8987 ("cpufreq: Avoid unnecessary frequency updates due to mismatch")
    Signed-off-by: Sanjay Chandrashekara <sanjayc@nvidia.com>
    [ sumit gupta: Commit message update ]
    Signed-off-by: Sumit Gupta <sumitg@nvidia.com>
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 56d9ac7436480b2e3965b610e6c30c13c0a5649d
Author: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
Date:   Fri Apr 14 16:00:48 2023 +0200

    ACPI: bus: Ensure that notify handlers are not running after removal

    [ Upstream commit faae443738c6f0dac9b0d3d11d108f6911a989a9 ]

    Currently, acpi_device_remove_notify_handler() may return while the
    notify handler being removed is still running which may allow the
```

```
    module holding that handler to be torn down prematurely.

    Address this issue by making acpi_device_remove_notify_handler() wait
    for the handling of all the ACPI events in progress to complete before
    returning.

    Fixes: 5894b0c46e49 ("ACPI / scan: Move bus operations and notification routines to
bus.c")
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 571c3b46c9b3c4176498ffb0705c0c546f3bf3e1
Author: Sebastian Andrzej Siewior <bigeasy@linutronix.de>
Date:   Tue Apr 18 14:26:39 2023 +0200

    tick/common: Align tick period with the HZ tick.

    [ Upstream commit e9523a0d81899361214d118ad60ef76f0e92f71d ]

    With HIGHRES enabled tick_sched_timer() is programmed every jiffy to
    expire the timer_list timers. This timer is programmed accurate in
    respect to CLOCK_MONOTONIC so that 0 seconds and nanoseconds is the
    first tick and the next one is 1000/CONFIG_HZ ms later. For HZ=250 it is
    every 4 ms and so based on the current time the next tick can be
    computed.

    This accuracy broke since the commit mentioned below because the jiffy
    based clocksource is initialized with higher accuracy in
    read_persistent_wall_and_boot_offset(). This higher accuracy is
    inherited during the setup in tick_setup_device(). The timer still fires
    every 4ms with HZ=250 but timer is no longer aligned with
    CLOCK_MONOTONIC with 0 as it origin but has an offset in the us/ns part
    of the timestamp. The offset differs with every boot and makes it
    impossible for user land to align with the tick.

    Align the tick period with CLOCK_MONOTONIC ensuring that it is always a
    multiple of 1000/CONFIG_HZ ms.

    Fixes: 857baa87b6422 ("sched/clock: Enable sched clock early")
    Reported-by: Gusenleitner Klaus <gus@keba.com>
    Signed-off-by: Sebastian Andrzej Siewior <bigeasy@linutronix.de>
    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Link: https://lore.kernel.org/20230406095735.0_14edn3@linutronix.de
    Link: https://lore.kernel.org/r/20230418122639.ikgfvu3f@linutronix.de
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 596d7308e189a3230bf33d667b64acc73846c2d0
Author: Cong Liu <liucong2@kylinos.cn>
Date:   Sat Apr 15 00:41:09 2023 +0200

    drm/i915: Fix memory leaks in i915 selftests

    [ Upstream commit 803033c148f754f32da1b93926c49c22731ec485 ]

    This patch fixes memory leaks on error escapes in function fake_get_pages

    Fixes: c3bfba9a2225 ("drm/i915: Check for integer truncation on scatterlist creation")
    Signed-off-by: Cong Liu <liucong2@kylinos.cn>
    Reviewed-by: Andrzej Hajda <andrzej.hajda@intel.com>
    Reviewed-by: Andi Shyti <andi.shyti@linux.intel.com>
    Signed-off-by: Andi Shyti <andi.shyti@linux.intel.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20230414224109.1051922-1-
andi.shyti@linux.intel.com
    (cherry picked from commit 8bfbdadce85c4c51689da10f39c805a7106d4567)
    Signed-off-by: Joonas Lahtinen <joonas.lahtinen@linux.intel.com>
```

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

commit fd8b0abecdf66379e9d25d7448b942b5be379cb2
Author: Ville Syrjälä <ville.syrjala@linux.intel.com>
Date:   Thu Apr 13 23:06:02 2023 +0300

    drm/i915: Make intel_get_crtc_new_encoder() less oopsy

    [ Upstream commit 631420b06597a33c72b6dcef78d1c2dea17f452d ]

    The point of the WARN was to print something, not oops
    straight up. Currently that is precisely what happens
    if we can't find the connector for the crtc in the atomic
    state. Get the dev pointer from the atomic state instead
    of the potentially NULL encoder to avoid that.

    Signed-off-by: Ville Syrjälä <ville.syrjala@linux.intel.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20230413200602.6037-2-
ville.syrjala@linux.intel.com
    Fixes: 3a47ae201e07 ("drm/i915/display: Make WARN* drm specific where encoder ptr is
available")
    Reviewed-by: Jani Nikula <jani.nikula@intel.com>
    (cherry picked from commit 3b6692357f70498f617ea1b31a0378070a0acf1c)
    Signed-off-by: Joonas Lahtinen <joonas.lahtinen@linux.intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 75dc252e705d2e38fc027b2f0b2970f88a3c21f0
Author: Thomas Gleixner <tglx@linutronix.de>
Date:   Wed Apr 12 09:54:39 2023 +0200

    debugobject: Prevent init race with static objects

    [ Upstream commit 63a759694eed61025713b3e14dd827c8548daadc ]

    Statically initialized objects are usually not initialized via the init()
    function of the subsystem. They are special cased and the subsystem
    provides a function to validate whether an object which is not yet tracked
    by debugobjects is statically initialized. This means the object is started
    to be tracked on first use, e.g. activation.

    This works perfectly fine, unless there are two concurrent operations on
    that object. Schspa decoded the problem:

    T0                                  T1

    debug_object_assert_init(addr)
      lock_hash_bucket()
      obj = lookup_object(addr);
      if (!obj) {
            unlock_hash_bucket();
            - > preemption
                                        lock_subsytem_object(addr);
                                          activate_object(addr)
                                          lock_hash_bucket();
                                          obj = lookup_object(addr);
                                          if (!obj) {
                                            unlock_hash_bucket();
                                            if (is_static_object(addr))
                                               init_and_track(addr);
                                          lock_hash_bucket();
                                          obj = lookup_object(addr);
                                          obj->state = ACTIVATED;
                                          unlock_hash_bucket();

                                        subsys function modifies content of addr,
```

```
                                        so static object detection does
                                        not longer work.

                                        unlock_subsytem_object(addr);

                if (is_static_object(addr)) <- Fails

                  debugobject emits a warning and invokes the fixup function which
                  reinitializes the already active object in the worst case.
```

This race exists forever, but was never observed until mod_timer() got a debug_object_assert_init() added which is outside of the timer base lock held section right at the beginning of the function to cover the lockless early exit points too.

Rework the code so that the lookup, the static object check and the tracking object association happens atomically under the hash bucket lock. This prevents the issue completely as all callers are serialized on the hash bucket lock and therefore cannot observe inconsistent state.

Fixes: 3ac7fe5a4aab ("infrastructure to debug (dynamic) objects")
Reported-by: syzbot+5093ba19745994288b53@syzkaller.appspotmail.com
Debugged-by: Schspa Shi <schspa@gmail.com>
Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
Reviewed-by: Stephen Boyd <swboyd@chromium.org>
Link: https://syzkaller.appspot.com/bug?id=22c8a5938eab640d1c6bcc0e3dc7be519d878462
Link: https://lore.kernel.org/lkml/20230303161906.831686-1-schspa@gmail.com
Link: https://lore.kernel.org/r/87zg7dzgao.ffs@tglx
Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c21717e575252334291893e9ef9ec2c6614d8f75
Author: Yunfei Dong <yunfei.dong@mediatek.com>
Date: Tue Mar 28 10:14:50 2023 +0100

    media: mediatek: vcodec: add remove function for decoder platform driver

    [Upstream commit e2a10b3801061d05d3e3415b9b824251451cfd6c]

    Need to disable decoder power when remove decoder hardware driver, adding
    remove callback function in the definition of platform driver.

    Fixes: c05bada35f01 ("media: mtk-vcodec: Add to support multi hardware decode")
    Signed-off-by: Yunfei Dong <yunfei.dong@mediatek.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 34fe290090ecfcf405cad9d0e0ddc8b8246ffaa2
Author: Yunfei Dong <yunfei.dong@mediatek.com>
Date: Tue Mar 28 10:14:49 2023 +0100

    media: mediatek: vcodec: fix decoder disable pm crash

    [Upstream commit 9d2f13fb47dcab6d094f34ecfd6a879a409722b3]

    Can't call pm_runtime_disable when the architecture support sub device for
    'dev->pm.dev' is NUll, or will get below crash log.

    [10.771551] pc : _raw_spin_lock_irq+0x4c/0xa0
    [10.771556] lr : __pm_runtime_disable+0x30/0x130
    [10.771558] sp : ffffffc01e4cb800
    [10.771559] x29: ffffffc01e4cb800 x28: ffffffdf082108a8
    [10.771563] x27: ffffffc01e4cbd70 x26: ffffff8605df55f0
    [10.771567] x25: 0000000000000002 x24: 0000000000000002
    [10.771570] x23: ffffff85c0dc9c00 x22: 0000000000000001
```
```

```
     [   10.771573] x21: 0000000000000001 x20: 0000000000000000
     [   10.771577] x19: 00000000000000f4 x18: ffffffdf2e9fbe18
     [   10.771580] x17: 0000000000000000 x16: ffffffdf2df13c74
     [   10.771583] x15: 00000000000002ea x14: 0000000000000058
     [   10.771587] x13: ffffffdf2de1b62c x12: ffffffdf2e9e30e4
     [   10.771590] x11: 0000000000000000 x10: 0000000000000001
     [   10.771593] x9 : 0000000000000000 x8 : 00000000000000f4
     [   10.771596] x7 : 6bff6264632c6264 x6 : 0000000000008000
     [   10.771600] x5 : 0080000000000000 x4 : 0000000000000001
     [   10.771603] x3 : 0000000000000008 x2 : 0000000000000001
     [   10.771608] x1 : 0000000000000000 x0 : 00000000000000f4
     [   10.771613] Call trace:
     [   10.771617]  _raw_spin_lock_irq+0x4c/0xa0
     [   10.771620]  __pm_runtime_disable+0x30/0x130
     [   10.771657]  mtk_vcodec_probe+0x69c/0x728 [mtk_vcodec_dec
 800cc929d6631f79f9b273254c8db94d0d3500dc]
     [   10.771662]  platform_drv_probe+0x9c/0xbc
     [   10.771665]  really_probe+0x13c/0x3a0
     [   10.771668]  driver_probe_device+0x84/0xc0
     [   10.771671]  device_driver_attach+0x54/0x78

     Fixes: ba31a5b39400 ("media: mtk-vcodec: Remove mtk_vcodec_release_dec_pm")
     Signed-off-by: Yunfei Dong <yunfei.dong@mediatek.com>
     Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
     Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit 0f7ad29380fdf0415fab08d63c0f236728f649a8
 Author: Robin Murphy <robin.murphy@arm.com>
 Date:   Wed Apr 12 11:29:40 2023 +0100

     perf/arm-cmn: Fix port detection for CMN-700

     [ Upstream commit 2ad91e44e6b0c7ef1ed151b3bb2242a2144e6085 ]

     When the "extra device ports" configuration was first added, the
     additional mxp_device_port_connect_info registers were added around the
     existing mxp_mesh_port_connect_info registers. What I missed about
     CMN-700 is that it shuffled them around to remove this discontinuity.
     As such, tweak the definitions and factor out a helper for reading these
     registers so we can deal with this discrepancy easily, which does at
     least allow nicely tidying up the callsites. With this we can then also
     do the nice thing and skip accesses completely rather than relying on
     RES0 behaviour where we know the extra registers aren't defined.

     Fixes: 23760a014417 ("perf/arm-cmn: Add CMN-700 support")
     Reported-by: Jing Zhang <renyu.zj@linux.alibaba.com>
     Signed-off-by: Robin Murphy <robin.murphy@arm.com>
     Link:
 https://lore.kernel.org/r/71d129241d4d7923cde72a0e5b4c8d2f6084525f.1681295193.git.robin.murp
 hy@arm.com
     Signed-off-by: Will Deacon <will@kernel.org>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit b75e93c06be771b192e01140f5cc60f96c7ad542
 Author: Sumit Garg <sumit.garg@linaro.org>
 Date:   Thu Feb 2 13:01:48 2023 +0530

     arm64: kgdb: Set PSTATE.SS to 1 to re-enable single-step

     [ Upstream commit af6c0bd59f4f3ad5daad2f7b777954b1954551d5 ]

     Currently only the first attempt to single-step has any effect. After
     that all further stepping remains "stuck" at the same program counter
     value.
```

```
    Refer to the ARM Architecture Reference Manual (ARM DDI 0487E.a) D2.12,
    PSTATE.SS=1 should be set at each step before transferring the PE to the
    'Active-not-pending' state. The problem here is PSTATE.SS=1 is not set
    since the second single-step.

    After the first single-step, the PE transferes to the 'Inactive' state,
    with PSTATE.SS=0 and MDSCR.SS=1, thus PSTATE.SS won't be set to 1 due to
    kernel_active_single_step()=true. Then the PE transferes to the
    'Active-pending' state when ERET and returns to the debugger by step
    exception.

    Before this patch:
    ==================
    Entering kdb (current=0xffff3376039f0000, pid 1) on processor 0 due to Keyboard Entry
    [0]kdb>

    [0]kdb>
    [0]kdb> bp write_sysrq_trigger
    Instruction(i) BP #0 at 0xffffa45c13d09290 (write_sysrq_trigger)
        is enabled   addr at ffffa45c13d09290, hardtype=0 installed=0

    [0]kdb> go
    $ echo h > /proc/sysrq-trigger

    Entering kdb (current=0xffff4f7e453f8000, pid 175) on processor 1 due to Breakpoint @
0xffffad651a309290
    [1]kdb> ss

    Entering kdb (current=0xffff4f7e453f8000, pid 175) on processor 1 due to SS trap @
0xffffad651a309294
    [1]kdb> ss

    Entering kdb (current=0xffff4f7e453f8000, pid 175) on processor 1 due to SS trap @
0xffffad651a309294
    [1]kdb>

    After this patch:
    =================
    Entering kdb (current=0xffff6851c39f0000, pid 1) on processor 0 due to Keyboard Entry
    [0]kdb> bp write_sysrq_trigger
    Instruction(i) BP #0 at 0xffffc02d2dd09290 (write_sysrq_trigger)
        is enabled   addr at ffffc02d2dd09290, hardtype=0 installed=0

    [0]kdb> go
    $ echo h > /proc/sysrq-trigger

    Entering kdb (current=0xffff6851c53c1840, pid 174) on processor 1 due to Breakpoint @
0xffffc02d2dd09290
    [1]kdb> ss

    Entering kdb (current=0xffff6851c53c1840, pid 174) on processor 1 due to SS trap @
0xffffc02d2dd09294
    [1]kdb> ss

    Entering kdb (current=0xffff6851c53c1840, pid 174) on processor 1 due to SS trap @
0xffffc02d2dd09298
    [1]kdb> ss

    Entering kdb (current=0xffff6851c53c1840, pid 174) on processor 1 due to SS trap @
0xffffc02d2dd0929c
    [1]kdb>

    Fixes: 44679a4f142b ("arm64: KGDB: Add step debugging support")
    Co-developed-by: Wei Li <liwei391@huawei.com>
```

```
    Signed-off-by: Wei Li <liwei391@huawei.com>
    Signed-off-by: Sumit Garg <sumit.garg@linaro.org>
    Tested-by: Douglas Anderson <dianders@chromium.org>
    Acked-by: Daniel Thompson <daniel.thompson@linaro.org>
    Tested-by: Daniel Thompson <daniel.thompson@linaro.org>
    Link: https://lore.kernel.org/r/20230202073148.657746-3-sumit.garg@linaro.org
    Signed-off-by: Will Deacon <will@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8788ef01061d96aefcded50288b74dbf9a835e96
Author: Saurabh Sengar <ssengar@linux.microsoft.com>
Date:   Tue Mar 28 00:30:04 2023 -0700

    x86/ioapic: Don't return 0 from arch_dynirq_lower_bound()

    [ Upstream commit 5af507bef93c09a94fb8f058213b489178f4cbe5 ]

    arch_dynirq_lower_bound() is invoked by the core interrupt code to
    retrieve the lowest possible Linux interrupt number for dynamically
    allocated interrupts like MSI.

    The x86 implementation uses this to exclude the IO/APIC GSI space.
    This works correctly as long as there is an IO/APIC registered, but
    returns 0 if not. This has been observed in VMs where the BIOS does
    not advertise an IO/APIC.

    0 is an invalid interrupt number except for the legacy timer interrupt
    on x86. The return value is unchecked in the core code, so it ends up
    to allocate interrupt number 0 which is subsequently considered to be
    invalid by the caller, e.g. the MSI allocation code.

    The function has already a check for 0 in the case that an IO/APIC is
    registered, as ioapic_dynirq_base is 0 in case of device tree setups.

    Consolidate this and zero check for both ioapic_dynirq_base and gsi_top,
    which is used in the case that no IO/APIC is registered.

    Fixes: 3e5bedc2c258 ("x86/apic: Fix arch_dynirq_lower_bound() bug for DT enabled
machines")
    Signed-off-by: Saurabh Sengar <ssengar@linux.microsoft.com>
    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Link: https://lore.kernel.org/r/1679988604-20308-1-git-send-email-
ssengar@linux.microsoft.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0ad07e02be0d3f0d554653382ffe53ae4879378d
Author: YAN SHI <m202071378@hust.edu.cn>
Date:   Wed Apr 12 11:35:29 2023 +0800

    regulator: stm32-pwr: fix of_iomap leak

    [ Upstream commit c4a413e56d16a2ae84e6d8992f215c4dcc7fac20 ]

    Smatch reports:
    drivers/regulator/stm32-pwr.c:166 stm32_pwr_regulator_probe() warn:
    'base' from of_iomap() not released on lines: 151,166.

    In stm32_pwr_regulator_probe(), base is not released
    when devm_kzalloc() fails to allocate memory or
    devm_regulator_register() fails to register a new regulator device,
    which may cause a leak.

    To fix this issue, replace of_iomap() with
    devm_platform_ioremap_resource(). devm_platform_ioremap_resource()
    is a specialized function for platform devices.
```

```
    It allows 'base' to be automatically released whether the probe
    function succeeds or fails.

    Besides, use IS_ERR(base) instead of !base
    as the return value of devm_platform_ioremap_resource()
    can either be a pointer to the remapped memory or
    an ERR_PTR() encoded error code if the operation fails.

    Fixes: dc62f951a6a8 ("regulator: stm32-pwr: Fix return value check in
stm32_pwr_regulator_probe()")
    Signed-off-by: YAN SHI <m202071378@hust.edu.cn>
    Reported-by: kernel test robot <lkp@intel.com>
    Link: https://lore.kernel.org/oe-kbuild-all/202304111750.o2643eJN-lkp@intel.com/
    Reviewed-by: Dongliang Mu <dzm91@hust.edu.cn>
    Link: https://lore.kernel.org/r/20230412033529.18890-1-m202071378@hust.edu.cn
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2a1459b55edfa6cd1a7374d7e4e1190a59c2a729
Author: Javier Martinez Canillas <javierm@redhat.com>
Date:   Fri Feb 10 09:18:35 2023 +0100

    media: venus: dec: Fix capture formats enumeration order

    [ Upstream commit a9d45ec74c8e68aaafe90191928eddbf79f4644f ]

    Commit 9593126dae3e ("media: venus: Add a handling of QC08C compressed
    format") and commit cef92b14e653 ("media: venus: Add a handling of QC10C
    compressed format") added support for the QC08C and QC10C compressed
    formats respectively.

    But these also caused a regression, because the new formats where added
    at the beginning of the vdec_formats[] array and the vdec_inst_init()
    function sets the default format output and capture using fixed indexes
    of that array:

    static void vdec_inst_init(struct venus_inst *inst)
    {
    ...
            inst->fmt_out = &vdec_formats[8];
            inst->fmt_cap = &vdec_formats[0];
    ...
    }

    Since now V4L2_PIX_FMT_NV12 is not the first entry in the array anymore,
    the default capture format is not set to that as it was done before.

    Both commits changed the first index to keep inst->fmt_out default format
    set to V4L2_PIX_FMT_H264, but did not update the latter to keep .fmt_out
    default format set to V4L2_PIX_FMT_NV12.

    Rather than updating the index to the current V4L2_PIX_FMT_NV12 position,
    let's reorder the entries so that this format is the first entry again.

    This would also make VIDIOC_ENUM_FMT report the V4L2_PIX_FMT_NV12 format
    with an index 0 as it did before the QC08C and QC10C formats were added.

    Fixes: 9593126dae3e ("media: venus: Add a handling of QC08C compressed format")
    Fixes: cef92b14e653 ("media: venus: Add a handling of QC10C compressed format")
    Signed-off-by: Javier Martinez Canillas <javierm@redhat.com>
    Signed-off-by: Stanimir Varbanov <stanimir.k.varbanov@gmail.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0d1cfb14e0f3f9076c3575c867c38d2e1449ccaa
```

```
Author: Michał Krawczyk <mk@semihalf.com>
Date:   Mon Jan 30 13:54:18 2023 +0000

    media: venus: dec: Fix handling of the start cmd

    [ Upstream commit 50248ad9f190d527cbd578190ca769729518b703 ]

    The decoder driver should clear the last_buffer_dequeued flag of the
    capture queue upon receiving V4L2_DEC_CMD_START.

    The last_buffer_dequeued flag is set upon receiving EOS (which always
    happens upon receiving V4L2_DEC_CMD_STOP).

    Without this patch, after issuing the V4L2_DEC_CMD_STOP and
    V4L2_DEC_CMD_START, the vb2_dqbuf() function will always fail, even if
    the buffers are completed by the hardware.

    Fixes: beac82904a87 ("media: venus: make decoder compliant with stateful codec API")

    Signed-off-by: Michał Krawczyk <mk@semihalf.com>
    Signed-off-by: Stanimir Varbanov <stanimir.k.varbanov@gmail.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit bad8617cd0f16a235cab7c705d7f54ffa1ff3788
Author: Florian Fainelli <f.fainelli@gmail.com>
Date:   Fri Mar 24 13:38:33 2023 -0700

    media: rc: gpio-ir-recv: Fix support for wake-up

    [ Upstream commit 9c592f8ab114875fdb3b2040f01818e53de44991 ]

    The driver was intended from the start to be a wake-up source for the
    system, however due to the absence of a suitable call to
    device_set_wakeup_capable(), the device_may_wakeup() call used to decide
    whether to enable the GPIO interrupt as a wake-up source would never
    happen. Lookup the DT standard "wakeup-source" property and call
    device_init_wakeup() to ensure the device is flagged as being wakeup
    capable.

    Reported-by: Matthew Lear <matthew.lear@broadcom.com>
    Fixes: fd0f6851eb46 ("[media] rc: Add support for GPIO based IR Receiver driver")
    Signed-off-by: Florian Fainelli <f.fainelli@gmail.com>
    Signed-off-by: Sean Young <sean@mess.org>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d4b749771fed3f99bbe8880eaab32a05ede0e5fa
Author: Igor Artemiev <Igor.A.Artemiev@mcst.ru>
Date:   Mon Apr 3 16:10:37 2023 +0300

    drm/amd/display: Fix potential null dereference

    [ Upstream commit 52f1783ff4146344342422c1cd94fcb4ce39b6fe ]

    The adev->dm.dc pointer can be NULL and dereferenced in amdgpu_dm_fini()
    without checking.

    Add a NULL pointer check before calling dc_dmub_srv_destroy().

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

    Fixes: 9a71c7d31734 ("drm/amd/display: Register DMUB service with DC")
    Signed-off-by: Igor Artemiev <Igor.A.Artemiev@mcst.ru>
    Signed-off-by: Hamza Mahfooz <hamza.mahfooz@amd.com>
```

```
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 07f0f15e5db60c5b0722049d3251ef4a46dc3b76
Author: Wei Chen <harperchen1110@gmail.com>
Date:   Mon Mar 27 11:58:09 2023 +0000

    media: hi846: Fix memleak in hi846_init_controls()

    [ Upstream commit 2649c1a20e8e399ee955d0e22192f9992662c3d2 ]

    hi846_init_controls doesn't clean the allocated ctrl_hdlr
    in case there is a failure, which causes memleak. Add
    v4l2_ctrl_handler_free to free the resource properly.

    Fixes: e8c0882685f9 ("media: i2c: add driver for the SK Hynix Hi-846 8M pixel camera")
    Signed-off-by: Wei Chen <harperchen1110@gmail.com>
    Reviewed-by: Martin Kepplinger <martin.kepplinger@puri.sm>
    Signed-off-by: Sakari Ailus <sakari.ailus@linux.intel.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f2b98b6f6acbba4ba84ad9d00142baaef984234f
Author: Sakari Ailus <sakari.ailus@linux.intel.com>
Date:   Tue Mar 7 15:49:44 2023 +0200

    media: v4l: async: Return async sub-devices to subnotifier list

    [ Upstream commit 5276c9df9c2ab9a43b534bfb56bdb10899cd3a22 ]

    When an async notifier is unregistered, the async sub-devices in the
    notifier's done list will disappear with the notifier. However this is
    currently also done to the sub-notifiers that remain registered. Their
    sub-devices only need to be unbound while the async sub-devices themselves
    need to be returned to the sub-notifier's waiting list. Do this now.

    Fixes: 2cab00bb076b ("media: v4l: async: Allow binding notifiers to sub-devices")
    Signed-off-by: Sakari Ailus <sakari.ailus@linux.intel.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 48765ca7c6b71bf73a4cc8475a4bad9e2633cf61
Author: Miaoqian Lin <linmq006@gmail.com>
Date:   Fri Jan 6 11:58:09 2023 +0400

    media: rcar_fdp1: Fix refcount leak in probe and remove function

    [ Upstream commit c766c90faf93897b77c9c5daa603cffab85ba907 ]

    rcar_fcp_get() take reference, which should be balanced with
    rcar_fcp_put(). Add missing rcar_fcp_put() in fdp1_remove and
    the error paths of fdp1_probe() to fix this.

    Fixes: 4710b752e029 ("[media] v4l: Add Renesas R-Car FDP1 Driver")
    Signed-off-by: Miaoqian Lin <linmq006@gmail.com>
    Reviewed-by: Laurent Pinchart <laurent.pinchart+renesas@ideasonboard.com>
    Signed-off-by: Laurent Pinchart <laurent.pinchart+renesas@ideasonboard.com>
    [hverkuil: resolve merge conflict, remove() is now void]
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e05b56a76df90b0d98699b1e4cec479c66a753b4
Author: Uwe Kleine-König <u.kleine-koenig@pengutronix.de>
Date:   Sun Mar 26 16:31:19 2023 +0200
```

```
    media: rcar_fdp1: Convert to platform remove callback returning void

    [ Upstream commit 0e82d3715fd208de567b8e4307fbf91ae5e57db4 ]

    The .remove() callback for a platform driver returns an int which makes
    many driver authors wrongly assume it's possible to do error handling by
    returning an error code. However the value returned is (mostly) ignored
    and this typically results in resource leaks. To improve here there is a
    quest to make the remove callback return void. In the first step of this
    quest all drivers are converted to .remove_new() which already returns
    void.

    Trivially convert this driver from always returning zero in the remove
    callback to the void returning variant.

    Signed-off-by: Uwe Kleine-König <u.kleine-koenig@pengutronix.de>
    Reviewed-by: Geert Uytterhoeven <geert+renesas@glider.be>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Stable-dep-of: c766c90faf93 ("media: rcar_fdp1: Fix refcount leak in probe and remove
function")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b54f74214adf4e77cba6badf488c564dd353b491
Author: Laurent Pinchart <laurent.pinchart+renesas@ideasonboard.com>
Date:   Fri Jan 20 22:20:31 2023 +0200

    media: vsp1: Replace vb2_is_streaming() with vb2_start_streaming_called()

    [ Upstream commit 52d8caca3d533cc499f1255be25576ffd936ec95 ]

    The vsp1 driver uses the vb2_is_streaming() function in its .buf_queue()
    handler to check if the .start_streaming() operation has been called,
    and decide whether to just add the buffer to an internal queue, or also
    trigger a hardware run. vb2_is_streaming() relies on the vb2_queue
    structure's streaming field, which used to be set only after calling the
    .start_streaming() operation.

    Commit a10b21532574 ("media: vb2: add (un)prepare_streaming queue ops")
    changed this, setting the .streaming field in vb2_core_streamon() before
    enqueuing buffers to the driver and calling .start_streaming(). This
    broke the vsp1 driver which now believes that .start_streaming() has
    been called when it hasn't, leading to a crash:

    [  881.058705] Unable to handle kernel NULL pointer dereference at virtual address
0000000000000020
    [  881.067495] Mem abort info:
    [  881.070290]   ESR = 0x0000000096000006
    [  881.074042]   EC = 0x25: DABT (current EL), IL = 32 bits
    [  881.079358]   SET = 0, FnV = 0
    [  881.082414]   EA = 0, S1PTW = 0
    [  881.085558]   FSC = 0x06: level 2 translation fault
    [  881.090439] Data abort info:
    [  881.093320]   ISV = 0, ISS = 0x00000006
    [  881.097157]   CM = 0, WnR = 0
    [  881.100126] user pgtable: 4k pages, 48-bit VAs, pgdp=000000004fa51000
    [  881.106573] [0000000000000020] pgd=080000004f36e003, p4d=080000004f36e003,
pud=080000004f7ec003, pmd=0000000000000000
    [  881.117217] Internal error: Oops: 0000000096000006 [#1] PREEMPT SMP
    [  881.123494] Modules linked in: rcar_fdp1 v4l2_mem2mem
    [  881.128572] CPU: 0 PID: 1271 Comm: yavta Tainted: G    B              6.2.0-rc1-
00023-g6c94e2e99343 #556
    [  881.138061] Hardware name: Renesas Salvator-X 2nd version board based on r8a77965
(DT)
    [  881.145981] pstate: 400000c5 (nZcv daIF -PAN -UAO -TCO -DIT -SSBS BTYPE=--)
    [  881.152951] pc : vsp1_dl_list_add_body+0xa8/0xe0
```

```
[  881.157580] lr : vsp1_dl_list_add_body+0x34/0xe0
[  881.162206] sp : ffff80000c267710
[  881.165522] x29: ffff80000c267710 x28: ffff000010938ae8 x27: ffff000013a8dd98
[  881.172683] x26: ffff000010938098 x25: ffff000013a8dc00 x24: ffff000010ed6ba8
[  881.179841] x23: ffff00000faa4000 x22: 0000000000000000 x21: 0000000000000020
[  881.186998] x20: ffff00000faa4000 x19: 0000000000000000 x18: 0000000000000000
[  881.194154] x17: 0000000000000000 x16: 0000000000000000 x15: 0000000000000000
[  881.201309] x14: 0000000000000000 x13: 746e696174206c65 x12: ffff70000157043d
[  881.208465] x11: 1ffff0000157043c x10: ffff70000157043c x9 : dfff800000000000
[  881.215622] x8 : ffff80000ab821e7 x7 : 00008ffffea8fbc4 x6 : 0000000000000001
[  881.222779] x5 : ffff80000ab821e0 x4 : ffff70000157043d x3 : 0000000000000020
[  881.229936] x2 : 0000000000000020 x1 : ffff00000e4f6400 x0 : 0000000000000000
[  881.237092] Call trace:
[  881.239542]  vsp1_dl_list_add_body+0xa8/0xe0
[  881.243822]  vsp1_video_pipeline_run+0x270/0x2a0
[  881.248449]  vsp1_video_buffer_queue+0x1c0/0x1d0
[  881.253076]  __enqueue_in_driver+0xbc/0x260
[  881.257269]  vb2_start_streaming+0x48/0x200
[  881.261461]  vb2_core_streamon+0x13c/0x280
[  881.265565]  vb2_streamon+0x3c/0x90
[  881.269064]  vsp1_video_streamon+0x2fc/0x3e0
[  881.273344]  v4l_streamon+0x50/0x70
[  881.276844]  __video_do_ioctl+0x2bc/0x5d0
[  881.280861]  video_usercopy+0x2a8/0xc80
[  881.284704]  video_ioctl2+0x20/0x40
[  881.288201]  v4l2_ioctl+0xa4/0xc0
[  881.291525]  __arm64_sys_ioctl+0xe8/0x110
[  881.295543]  invoke_syscall+0x68/0x190
[  881.299303]  el0_svc_common.constprop.0+0x88/0x170
[  881.304105]  do_el0_svc+0x4c/0xf0
[  881.307430]  el0_svc+0x4c/0xa0
[  881.310494]  el0t_64_sync_handler+0xbc/0x140
[  881.314773]  el0t_64_sync+0x190/0x194
[  881.318450] Code: d50323bf d65f03c0 91008263 f9800071 (885f7c60)
[  881.324551] ---[ end trace 0000000000000000 ]---
[  881.329173] note: yavta[1271] exited with preempt_count 1
```

A different regression report sent to the linux-media mailing list ([1])
was answered with a claim that the vb2_is_streaming() function has never
been meant for this purpose. The document of the function, as well as of
the struct vb2_queue streaming field, is sparse, so this claim may be
hard to verify.

The information needed by the vsp1 driver to decide how to process
queued buffers is also available from the vb2_start_streaming_called()
function. Use it instead of vb2_is_streaming() to fix the problem.

[1] https://lore.kernel.org/linux-media/545610e7-3446-2b82-60dc-7385fea3774f@redhat.com/

Fixes: a10b21532574 ("media: vb2: add (un)prepare_streaming queue ops")
Signed-off-by: Laurent Pinchart <laurent.pinchart+renesas@ideasonboard.com>
Reviewed-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
Tested-by: Duy Nguyen <duy.nguyen.rh@renesas.com>
Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2457058d9ac942df63a4705ba6c696ccbf27bd28
Author: Moudy Ho <moudy.ho@mediatek.com>
Date: Mon Mar 27 11:13:24 2023 +0800

media: platform: mtk-mdp3: fix potential frame size overflow in mdp_try_fmt_mplane()

[Upstream commit 4168720753ce6c14c5d3a35302fc2e1841383443]

Fix overflow risk when setting certain formats whose frame size exceeds

```
    a RGB24 with 7723x7723 resolution.

    For example, a 7723x7724 RGB24 frame:
        1. bpl (byte per line) = 7723 * 3.
        2. Overflow occurs when bpl * 7724 * depth.

    Fixes: 61890ccaefaf ("media: platform: mtk-mdp3: add MediaTek MDP3 driver")
    Signed-off-by: Moudy Ho <moudy.ho@mediatek.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3a60e51489a3ec61565f5bc53f726ac9ccc6083c
Author: Zheng Wang <zyytlz.wz@163.com>
Date:   Sat Mar 18 16:50:23 2023 +0800

    media: saa7134: fix use after free bug in saa7134_finidev due to race condition

    [ Upstream commit 30cf57da176cca80f11df0d9b7f71581fe601389 ]

    In saa7134_initdev, it will call saa7134_hwinit1. There are three
    function invoking here: saa7134_video_init1, saa7134_ts_init1
    and saa7134_vbi_init1.

    All of them will init a timer with same function. Take
    saa7134_video_init1 as an example. It'll bound &dev->video_q.timeout
    with saa7134_buffer_timeout.

    In buffer_activate, the timer funtcion is started.

    If we remove the module or device which will call saa7134_finidev
    to make cleanup, there may be a unfinished work. The
    possible sequence is as follows, which will cause a
    typical UAF bug.

    Fix it by canceling the timer works accordingly before cleanup in
    saa7134_finidev.

    CPU0                  CPU1

                         |saa7134_buffer_timeout
    saa7134_finidev      |
      kfree(dev);        |
                         |
                         | saa7134_buffer_next
                         | //use dev

    Fixes: 1e7126b4a86a ("media: saa7134: Convert timers to use timer_setup()")
    Signed-off-by: Zheng Wang <zyytlz.wz@163.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d730bc84064364cafdb20c6ee7fda2cd7416407a
Author: Zheng Wang <zyytlz.wz@163.com>
Date:   Sat Mar 18 16:15:06 2023 +0800

    media: dm1105: Fix use after free bug in dm1105_remove due to race condition

    [ Upstream commit 5abda7a16698d4d1f47af1168d8fa2c640116b4a ]

    In dm1105_probe, it called dm1105_ir_init and bound
    &dm1105->ir.work with dm1105_emit_key.
    When it handles IRQ request with dm1105_irq,
    it may call schedule_work to start the work.

    When we call dm1105_remove to remove the driver, there
```

```
    may be a sequence as follows:

    Fix it by finishing the work before cleanup in dm1105_remove

    CPU0                  CPU1

                         |dm1105_emit_key
    dm1105_remove      |
      dm1105_ir_exit       |
        rc_unregister_device |
        rc_free_device  |
        rc_dev_release  |
        kfree(dev);     |
                         |
                         | rc_keydown
                         |   //use

    Fixes: 34d2f9bf189c ("V4L/DVB: dm1105: use dm1105_dev & dev instead of dm1105dvb")
    Signed-off-by: Zheng Wang <zyytlz.wz@163.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 35422336f5c2f6eaa58b4afbc29588d5b6dbc34d
Author: Shyam Sundar S K <Shyam-sundar.S-k@amd.com>
Date:   Mon Apr 10 00:23:46 2023 +0530

    platform/x86/amd: pmc: Move out of BIOS SMN pair for STB init

    [ Upstream commit 8d99129eef8f42377b41c1bacee9f8ce806e9f44 ]

    The current SMN index used for the driver probe seems to be meant
    for the BIOS pair and there are potential concurrency problems that can
    occur with an inopportune SMI.

    It is been advised to use SMN_INDEX_0 instead of SMN_INDEX_6, which is
    what amd_nb.c provides and this function has protections to ensure that
    only one caller can use it at a time.

    Fixes: 426c0ff27b83 ("platform/x86: amd-pmc: Add support for AMD Smart Trace Buffer")
    Co-developed-by: Sanket Goswami <Sanket.Goswami@amd.com>
    Signed-off-by: Sanket Goswami <Sanket.Goswami@amd.com>
    Signed-off-by: Shyam Sundar S K <Shyam-sundar.S-k@amd.com>
    Link: https://lore.kernel.org/r/20230409185348.556161-7-Shyam-sundar.S-k@amd.com
    Reviewed-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f23eaf32dab68de5d3b54fa081a1f28119c334f2
Author: Shyam Sundar S K <Shyam-sundar.S-k@amd.com>
Date:   Mon Apr 10 00:23:45 2023 +0530

    platform/x86/amd: pmc: Utilize SMN index 0 for driver probe

    [ Upstream commit 310e782a99c7f16fb533a45d8f9c16defefa5aab ]

    The current SMN index used for the driver probe seems to be meant
    for the BIOS pair and there are potential concurrency problems that can
    occur with an inopportune SMI.

    It is been advised to use SMN_INDEX_0 instead of SMN_INDEX_2, which is
    what amd_nb.c provides and this function has protections to ensure that
    only one caller can use it at a time.

    Fixes: 156ec4731cb2 ("platform/x86: amd-pmc: Add AMD platform support for S2Idle")
    Co-developed-by: Sanket Goswami <Sanket.Goswami@amd.com>
```

```
    Signed-off-by: Sanket Goswami <Sanket.Goswami@amd.com>
    Signed-off-by: Shyam Sundar S K <Shyam-sundar.S-k@amd.com>
    Link: https://lore.kernel.org/r/20230409185348.556161-6-Shyam-sundar.S-k@amd.com
    Reviewed-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1b3b41f98f8ab3d8d9b4a6de5493f9d2960e137c
Author: Mario Limonciello <mario.limonciello@amd.com>
Date:   Mon Apr 10 00:23:44 2023 +0530

    platform/x86/amd: pmc: Move idlemask check into `amd_pmc_idlemask_read`

    [ Upstream commit 9217bd1d7699f34a01b26ba14ff38c1714ce1185 ]

    The version check requirement for idle mask support actually only
    applies to RN/CZN/BRC platforms.

    So far no issues have happened because the PMFW version string is
    bigger on other supported systems.  This can be reset for any new platform
    so move the check to only RN/CZN/BRC case.

    Fixes: f6045de1f532 ("platform/x86: amd-pmc: Export Idlemask values based on the APU")
    Signed-off-by: Mario Limonciello <mario.limonciello@amd.com>
    Link: https://lore.kernel.org/r/20230409185348.556161-5-Shyam-sundar.S-k@amd.com
    Reviewed-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 67aeb286afb46fa09a3311357db6b0bef0905a0f
Author: Mario Limonciello <mario.limonciello@amd.com>
Date:   Mon Apr 10 00:23:43 2023 +0530

    platform/x86/amd: pmc: Don't dump data after resume from s0i3 on picasso

    [ Upstream commit 7abc3618b65304d409d9489d77e4a8f047842fb7 ]

    This command isn't supported on Picasso, so guard against running it
    to avoid errors like `SMU cmd unknown. err: 0xfe` in the logs.

    Link: https://gitlab.freedesktop.org/drm/amd/-/issues/2449
    Fixes: 766205674962 ("platform/x86: amd-pmc: Add support for logging SMU metrics")
    Signed-off-by: Mario Limonciello <mario.limonciello@amd.com>
    Link: https://lore.kernel.org/r/20230409185348.556161-4-Shyam-sundar.S-k@amd.com
    Reviewed-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 36a31c63a7d831bdc4b4062c40678ec106d584b7
Author: Mario Limonciello <mario.limonciello@amd.com>
Date:   Mon Apr 10 00:23:42 2023 +0530

    platform/x86/amd: pmc: Hide SMU version and program attributes for Picasso

    [ Upstream commit 5ec9ee0d464750d72972d5685edf675824e259a1 ]

    As the command to get version isn't supported on Picasso, we shouldn't
    be exposing this into sysfs either.

    Link: https://gitlab.freedesktop.org/drm/amd/-/issues/2449
    Fixes: 7f1ea75d499a ("platform/x86/amd: pmc: Add sysfs files for SMU")
    Signed-off-by: Mario Limonciello <mario.limonciello@amd.com>
    Link: https://lore.kernel.org/r/20230409185348.556161-3-Shyam-sundar.S-k@amd.com
    Reviewed-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
```

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

commit 17ad757c606f4119c37ad56926e9eb5f953ee78a
Author: Mario Limonciello <mario.limonciello@amd.com>
Date:   Mon Apr 10 00:23:41 2023 +0530

    platform/x86/amd: pmc: Don't try to read SMU version on Picasso

    [ Upstream commit b845772677ea19b8e4c032bc07393ef32de4ee39 ]

    Picasso doesn't support the command in the uPEP mailbox to try to
    read the SMU version.

    Link: https://gitlab.freedesktop.org/drm/amd/-/issues/2449
    Fixes: f6045de1f532 ("platform/x86: amd-pmc: Export Idlemask values based on the APU")
    Signed-off-by: Mario Limonciello <mario.limonciello@amd.com>
    Link: https://lore.kernel.org/r/20230409185348.556161-2-Shyam-sundar.S-k@amd.com
    Reviewed-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c347d0af9b896a82c08fd7567c61208b536e5105
Author: Shyam Sundar S K <Shyam-sundar.S-k@amd.com>
Date:   Thu Apr 6 22:18:07 2023 +0530

    platform/x86/amd/pmf: Move out of BIOS SMN pair for driver probe

    [ Upstream commit aec8298c093f052fc8a86f9411b69b23953b0edb ]

    The current SMN index used for the driver probe seems to be meant
    for the BIOS pair and there are potential concurrency problems that can
    occur with an inopportune SMI.

    It is been advised to use SMN_INDEX_0 instead of SMN_INDEX_2, which is
    what amd_nb.c provides and this function has protections to ensure that
    only one caller can use it at a time.

    Fixes: da5ce22df5fe ("platform/x86/amd/pmf: Add support for PMF core layer")
    Co-developed-by: Patil Rajesh Reddy <Patil.Reddy@amd.com>
    Signed-off-by: Patil Rajesh Reddy <Patil.Reddy@amd.com>
    Signed-off-by: Shyam Sundar S K <Shyam-sundar.S-k@amd.com>
    Link: https://lore.kernel.org/r/20230406164807.50969-4-Shyam-sundar.S-k@amd.com
    Reviewed-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2115e94838adc9d1e7b75043c9f26abcc910f6fb
Author: Zheng Wang <zyytlz.wz@163.com>
Date:   Mon Mar 13 16:42:20 2023 +0000

    media: rkvdec: fix use after free bug in rkvdec_remove

    [ Upstream commit 3228cec23b8b29215e18090c6ba635840190993d ]

    In rkvdec_probe, rkvdec->watchdog_work is bound with
    rkvdec_watchdog_func. Then rkvdec_vp9_run may
    be called to start the work.

    If we remove the module which will call rkvdec_remove
     to make cleanup, there may be a unfinished work.
     The possible sequence is as follows, which will
     cause a typical UAF bug.

    Fix it by canceling the work before cleanup in rkvdec_remove.
```

```
     CPU0                  CPU1

                          |rkvdec_watchdog_func
     rkvdec_remove        |
      rkvdec_v4l2_cleanup|
       v4l2_m2m_release  |
         kfree(m2m_dev); |
                          |
                          | v4l2_m2m_get_curr_priv
                          |   m2m_dev->curr_ctx //use
```

```
    Fixes: cd33c830448b ("media: rkvdec: Add the rkvdec driver")
    Signed-off-by: Zheng Wang <zyytlz.wz@163.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 565c863bd982584aa4393f7bdb345dbccb3ad488
Author: Zheng Wang <zyytlz.wz@163.com>
Date:   Mon Mar 13 16:31:20 2023 +0000

    media: cedrus: fix use after free bug in cedrus_remove due to race condition

    [ Upstream commit 50d0a7aea4809cef87979d4669911276aa23b71f ]

    In cedrus_probe, dev->watchdog_work is bound with cedrus_watchdog function.
    In cedrus_device_run, it will started by schedule_delayed_work. If there is
    an unfinished work in cedrus_remove, there may be a race condition and
    trigger UAF bug.
```

```
    CPU0                  CPU1

                          |cedrus_watchdog
    cedrus_remove         |
      v4l2_m2m_release    |
      kfree(m2m_dev)      |
                          |
                          | v4l2_m2m_get_curr_priv
                          |   m2m_dev //use
```

```
    Fix it by canceling the worker in cedrus_remove.

    Fixes: 7c38a551bda1 ("media: cedrus: Add watchdog for job completion")
    Signed-off-by: Zheng Wang <zyytlz.wz@163.com>
    Acked-by: Jernej Skrabec <jernej.skrabec@gmail.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1f7ff801410dc8bbaeb9152563a0bba21e026996
Author: Yunfei Dong <yunfei.dong@mediatek.com>
Date:   Wed Feb 1 07:33:16 2023 +0000

    media: mediatek: vcodec: change lat thread decode error condition

    [ Upstream commit 960badda95f10fb0c60f6f64978b19eafa9507a7 ]

    If lat thread can't get lat buffer, it should be that current instance
    don't be schedulded, the driver can't free the src buffer directly.

    Fixes: 7b182b8d9c85 ("media: mediatek: vcodec: Refactor get and put capture buffer
flow")
    Signed-off-by: Yunfei Dong <yunfei.dong@mediatek.com>
    Reviewed-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
```

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

commit 13c22f8f822f680d767adb3d68f6d2a494797cb3
Author: Yunfei Dong <yunfei.dong@mediatek.com>
Date:   Wed Feb 1 07:33:15 2023 +0000

    media: mediatek: vcodec: making sure queue_work successfully

    [ Upstream commit 2e0ef56d81cb2569624d288b7e95a8a2734a7c74 ]

    Putting core work to work queue using queue_work maybe fail, call
    queue_work again when the count of core work in work queue is less
    than core_list_cnt, making sure all the buffer in core list can be
    scheduled.

    Fixes: 365e4ba01df4 ("media: mtk-vcodec: Add work queue for core hardware decode")
    Signed-off-by: Yunfei Dong <yunfei.dong@mediatek.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d3a8364c1edfdfa7905ec680f178f92d73daad45
Author: Yunfei Dong <yunfei.dong@mediatek.com>
Date:   Wed Feb 1 07:33:14 2023 +0000

    media: mediatek: vcodec: remove unused lat_buf

    [ Upstream commit af50b13dd3d7d5dbc1f08add1c462398e926a053 ]

    Remove unused lat_buf from core list, or leading to core list access
    NULL point.

    Fixes: 365e4ba01df4 ("media: mtk-vcodec: Add work queue for core hardware decode")
    Signed-off-by: Yunfei Dong <yunfei.dong@mediatek.com>
    Reviewed-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e422f765b92774c18856a1e4f3f47c970043514b
Author: Yunfei Dong <yunfei.dong@mediatek.com>
Date:   Wed Feb 1 07:33:13 2023 +0000

    media: mediatek: vcodec: add core decode done event

    [ Upstream commit d227af847ac2d7d90350124a1b2451e4fc1f050c ]

    Need to make sure core decode done before current instance is free.

    Fixes: 365e4ba01df4 ("media: mtk-vcodec: Add work queue for core hardware decode")
    Signed-off-by: Yunfei Dong <yunfei.dong@mediatek.com>
    Reviewed-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4a62d8f3d8621d2a08d122627084e9c2e7ef8d18
Author: Yunfei Dong <yunfei.dong@mediatek.com>
Date:   Wed Feb 1 07:33:12 2023 +0000

    media: mediatek: vcodec: move lat_buf to the top of core list

    [ Upstream commit 2cfca6c1bf8074175ea7a3b6b47f77ebdef8f701 ]
```

```
    Current instance will decode done when begin to wait lat buf full,
    move the lat_buf of current instance to the top of core list to make
    sure current instance's lat_buf will be used firstly.

    Fixes: 365e4ba01df4 ("media: mtk-vcodec: Add work queue for core hardware decode")
    Signed-off-by: Yunfei Dong <yunfei.dong@mediatek.com>
    Reviewed-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e1192a04aa91c43dd8945cb4d738e53d01ea04b0
Author: Yunfei Dong <yunfei.dong@mediatek.com>
Date:   Wed Feb 1 07:33:11 2023 +0000

    media: mediatek: vcodec: using each instance lat_buf count replace core ready list

    [ Upstream commit f7a3780cf96925670736582b9a623a2c9ffb4166 ]

    Core Hardware decoder depends on each instance lat_buf count,
    calling queue_work decode again when the lat_buf count of each instance
    isn't zero.

    Fixes: 365e4ba01df4 ("media: mtk-vcodec: Add work queue for core hardware decode")
    Signed-off-by: Yunfei Dong <yunfei.dong@mediatek.com>
    Reviewed-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8f81185d2b135d406226fd465b244de2ff7673fc
Author: Yunfei Dong <yunfei.dong@mediatek.com>
Date:   Wed Feb 1 07:33:10 2023 +0000

    media: mediatek: vcodec: add params to record lat and core lat_buf count

    [ Upstream commit 5bbb6e2ca67477ab41163b32e6b3444faea74a5e ]

    Using lat_buf to share decoder information between lat and core work
    queue, adding params to record the buf count.

    Fixes: 365e4ba01df4 ("media: mtk-vcodec: Add work queue for core hardware decode")
    Signed-off-by: Yunfei Dong <yunfei.dong@mediatek.com>
    Reviewed-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8a4fe2254616c10d73c7e1e515c715524a2c002e
Author: Yunfei Dong <yunfei.dong@mediatek.com>
Date:   Sat Mar 18 03:56:51 2023 +0000

    media: mediatek: vcodec: Force capture queue format to MM21

    [ Upstream commit 68c7df527657a9e962da7b5b9c0308557357d8dc ]

    While the decoder can produce frames in both MM21 and MT21C formats, only
    MM21 is currently supported by userspace tools (like gstreamer and libyuv).
    In order to ensure userspace keeps working after the SCP firmware is
    updated to support both MM21 and MT21C formats, force the MM21 format for
    the capture queue.

    This is meant as a stopgap solution while dynamic format switching between
    MM21 and MT21C isn't implemented in the driver.
```

```
    Fixes: 7501edef6b1f ("media: mediatek: vcodec: Different codec using different capture
format")
    Signed-off-by: Yunfei Dong <yunfei.dong@mediatek.com>
    Reviewed-by: Nicolas F. R. A. Prado <nfraprado@collabora.com>
    Tested-by: Nicolas F. R. A. Prado <nfraprado@collabora.com>
    Reviewed-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a393b039aec8bac1e6c06f9e5df1ba4053e99ac0
Author: Yunfei Dong <yunfei.dong@mediatek.com>
Date:   Sat Mar 18 03:56:50 2023 +0000

    media: mediatek: vcodec: Make MM21 the default capture format

    [ Upstream commit 6d020d81b91af80a977061e82de25cafa4456af5 ]

    Given that only the MM21 capture format is supported by userspace tools
    (like gstreamer and libyuv), make it the default capture format.

    This allows us to force the MM21 format even when a MM21 and MT21C capable
    firmware is available (which is needed while dynamic format switching isn't
    implemented in the driver), without causing the following regressions on
    v4l2-compliance:

            fail: v4l2-test-formats.cpp(478): pixelformat 3132544d (MT21) for buftype 9 not
reported by ENUM_FMT
        test VIDIOC_G_FMT: FAIL
            fail: v4l2-test-formats.cpp(478): pixelformat 3132544d (MT21) for buftype 9 not
reported by ENUM_FMT
        test VIDIOC_TRY_FMT: FAIL
            fail: v4l2-test-formats.cpp(478): pixelformat 3132544d (MT21) for buftype 9 not
reported by ENUM_FMT
        test VIDIOC_S_FMT: FAIL

    Fixes: 7501edef6b1f ("media: mediatek: vcodec: Different codec using different capture
format")
    Signed-off-by: Yunfei Dong <yunfei.dong@mediatek.com>
    Reviewed-by: Nicolas F. R. A. Prado <nfraprado@collabora.com>
    Tested-by: Nicolas F. R. A. Prado <nfraprado@collabora.com>
    Reviewed-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5da74e770fc7a4d7c9212a14725295f6586db367
Author: Pin-yen Lin <treapking@chromium.org>
Date:   Tue Mar 14 10:22:41 2023 +0000

    media: mediatek: vcodec: Use 4K frame size when supported by stateful decoder

    [ Upstream commit e25528e1dbe52784ac250071653104a8adc848e2 ]

    After commit b018be06f3c7 ("media: mediatek: vcodec: Read max resolution
    from dec_capability"), the stateful video decoder driver never really
    sets its output frame size to 4K.

    Parse the decoder capability reported by the firmware, and update the
    output frame size in mtk_init_vdec_params to enable 4K frame size when
    available.

    Fixes: b018be06f3c7 ("media: mediatek: vcodec: Read max resolution from dec_capability")
    Signed-off-by: Pin-yen Lin <treapking@chromium.org>
    Reviewed-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
```

```
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f58094cbfe7e5e119297f7490806ab5e1f8b7cc5
Author: kyrie wu <kyrie.wu@mediatek.com>
Date:   Fri Mar 10 06:23:51 2023 +0000

    media: mtk-jpeg: Fixes jpeg enc&dec worker sw flow

    [ Upstream commit 86379bd9d399e2c5fd638a869af223d4910725c3 ]

    1. Move removing buffer after sw setting and before hw setting
    in enc&dec worker to prevents the operation of removing
    the buffer twice if the sw setting fails.
    2. Remove the redundant operation of queue work in the
    jpegenc irq handler because the jpegenc worker has called
    v4l2_m2m_job_finish to do it.

    Fixes: 5fb1c2361e56 ("mtk-jpegenc: add jpeg encode worker interface")
    Fixes: dedc21500334 ("media: mtk-jpegdec: add jpeg decode worker interface")
    Signed-off-by: kyrie wu <kyrie.wu@mediatek.com>
    Signed-off-by: irui wang <irui.wang@mediatek.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8746b36caa970a05dca284119d48489ac1e2b66d
Author: kyrie wu <kyrie.wu@mediatek.com>
Date:   Fri Mar 10 06:23:49 2023 +0000

    media: mtk-jpeg: Fixes jpeghw multi-core judgement

    [ Upstream commit 75c38caf66a10983acc5a59069bfc9492c43d682 ]

    some chips have multi-hw, but others have only one,
    modify the condition of multi-hw judgement

    Fixes: 934e8bccac95 ("mtk-jpegenc: support jpegenc multi-hardware")
    Signed-off-by: kyrie wu <kyrie.wu@mediatek.com>
    Signed-off-by: irui wang <irui.wang@mediatek.com>
    Reviewed-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e90945175c99ec4aa89bc1fd71ae2960b12c97cd
Author: Douglas Anderson <dianders@chromium.org>
Date:   Thu Mar 23 10:30:07 2023 -0700

    arm64: dts: sdm845: Rename qspi data12 as data23

    [ Upstream commit 37f7349b56decc91c66f8039712e63740b1f25f9 ]

    There are 4 qspi data pins: data0, data1, data2, and data3. Currently
    we have a shared pin state for data0 and data1 (2 lane config) and a
    pin state for data2 and data3 (you'd enable both this and the 2 lane
    state for 4 lanes). The second state is obviously misnamed. Fix it.

    Fixes: e1ce853932b7 ("arm64: dts: qcom: sdm845: Add qspi (quad SPI) node")
    Signed-off-by: Douglas Anderson <dianders@chromium.org>
    Acked-by: Linus Walleij <linus.walleij@linaro.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link:
https://lore.kernel.org/r/20230323102605.3.I88528d037b7fda4e53a40f661be5ac61628691cd@changei
```

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

commit 414a5dc95e5b43d6eba2dc62bf7b477f54ac3959
Author: Douglas Anderson <dianders@chromium.org>
Date:   Thu Mar 23 10:30:06 2023 -0700

    arm64: dts: sc7280: Rename qspi data12 as data23

    [ Upstream commit 14acf21c0d3f7b7298ffcd2e5b5db4a476ec6202 ]

    There are 4 qspi data pins: data0, data1, data2, and data3. Currently
    we have a shared pin state for data0 and data1 (2 lane config) and a
    pin state for data2 and data3 (you'd enable both this and the 2 lane
    state for 4 lanes). The second state is obviously misnamed. Fix it.

    Fixes: 7720ea001b52 ("arm64: dts: qcom: sc7280: Add QSPI node")
    Signed-off-by: Douglas Anderson <dianders@chromium.org>
    Acked-by: Linus Walleij <linus.walleij@linaro.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link:
https://lore.kernel.org/r/20230323102605.2.I4043491bb24b1e92267c5033d76cdb0fe60934da@changei
d
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ca5a4def52db80553f1ea0f7ecc288354a03d169
Author: Douglas Anderson <dianders@chromium.org>
Date:   Thu Mar 23 10:30:05 2023 -0700

    arm64: dts: sc7180: Rename qspi data12 as data23

    [ Upstream commit d84f8f2687bdc67f20262e822b206419bcfd0038 ]

    There are 4 qspi data pins: data0, data1, data2, and data3. Currently
    we have a shared pin state for data0 and data1 (2 lane config) and a
    pin state for data2 and data3 (you'd enable both this and the 2 lane
    state for 4 lanes). The second state is obviously misnamed. Fix it.

    Fixes: ba3fc6496366 ("arm64: dts: sc7180: Add qupv3_0 and qupv3_1")
    Signed-off-by: Douglas Anderson <dianders@chromium.org>
    Acked-by: Linus Walleij <linus.walleij@linaro.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link:
https://lore.kernel.org/r/20230323102605.1.Ifc1b5be04653f4ab119698a5944bfecded2080d6@changei
d
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 846e9405d469de8f73856d006a8ab935e537a3c3
Author: Petr Vorel <pvorel@suse.cz>
Date:   Tue Jan 31 21:04:14 2023 +0100

    arm64: dts: qcom: msm8994-angler: removed clash with smem_region

    [ Upstream commit c85c8a992794dfcd7cea7a41871710c27c5592a6 ]

    This fixes memory overlap error:
    [    0.000000] reserved@6300000 (0x0000000006300000--0x0000000007000000) overlaps with
smem_region@6a00000 (0x0000000006a00000--0x0000000006c00000)

    smem_region is the same as in downstream (qcom,smem) [1], therefore
    split reserved memory into two sections on either side of smem_region.

    Not adding labels as it's not expected to be used.

    [1] https://android.googlesource.com/kernel/msm/+/refs/heads/android-msm-angler-3.10-
```

marshmallow-mr1/arch/arm/boot/dts/qcom/msm8994.dtsi#948

    Fixes: 380cd3a34b7f ("arm64: dts: msm8994-angler: fix the memory map")

    Signed-off-by: Petr Vorel <pvorel@suse.cz>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230131200414.24373-3-pvorel@suse.cz
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 460f1f6c884801e6f03a103b49b9d62d8878265d
Author: Petr Vorel <pvorel@suse.cz>
Date: Tue Jan 31 21:04:13 2023 +0100

    arm64: dts: qcom: msm8994-angler: Fix cont_splash_mem mapping

    [Upstream commit fe88480a6be92ecbf6f205ff3a7d7e5ded0562dd]

    Angler's cont_splash_mem mapping is shorter in downstream [1],
    therefore 380cd3a34b7f was wrong. Obviously also 0e5ded926f2a was wrong
    (workaround which fixed booting at the time).

    This fixes error:
    [0.000000] memory@3401000 (0x0000000003401000--0x0000000005601000) overlaps with
tzapp@4800000 (0x0000000004800000--0x0000000006100000)

    [1] https://android.googlesource.com/kernel/msm/+/refs/heads/android-msm-angler-3.10-
marshmallow-mr1/arch/arm64/boot/dts/huawei/huawei_msm8994_angler_row_vn1/huawei-
fingerprint.dtsi#16

    Fixes: 380cd3a34b7f ("arm64: dts: msm8994-angler: fix the memory map")
    Fixes: 0e5ded926f2a ("arm64: dts: qcom: msm8994-angler: Disable cont_splash_mem")

    Signed-off-by: Petr Vorel <pvorel@suse.cz>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230131200414.24373-2-pvorel@suse.cz
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dd01549e36cd7a487fa5d246c1e07771c5b6ec1e
Author: Tushar Nimkar <quic_tnimkar@quicinc.com>
Date: Thu Apr 6 17:27:32 2023 +0530

    soc: qcom: rpmh-rsc: Support RSC v3 minor versions

    [Upstream commit 88704a0cd71909c3107561261412a5d5beb23358]

    RSC v3 register offsets are same for all minor versions of v3. Fix a
    minor version check to pick correct offsets for all v3 minor versions.

    Fixes: 40482e4f7364 ("soc: qcom: rpmh-rsc: Add support for RSC v3 register offsets")
    Signed-off-by: Tushar Nimkar <quic_tnimkar@quicinc.com>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230406115732.9293-1-quic_tnimkar@quicinc.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 878acafd2566d9d67b7d6750c9ab8615bf0352f2
Author: Uros Bizjak <ubizjak@gmail.com>
Date: Mon Feb 27 17:09:17 2023 +0100

    x86/apic: Fix atomic update of offset in reserve_eilvt_offset()

    [Upstream commit f96fb2df3eb31ede1b34b0521560967310267750]

    The detection of atomic update failure in reserve_eilvt_offset() is
    not correct. The value returned by atomic_cmpxchg() should be compared
    to the old value from the location to be updated.

```
    If these two are the same, then atomic update succeeded and
    "eilvt_offsets[offset]" location is updated to "new" in an atomic way.

    Otherwise, the atomic update failed and it should be retried with the
    value from "eilvt_offsets[offset]" - exactly what atomic_try_cmpxchg()
    does in a correct and more optimal way.

    Fixes: a68c439b1966c ("apic, x86: Check if EILVT APIC registers are available (AMD
only)")
    Signed-off-by: Uros Bizjak <ubizjak@gmail.com>
    Signed-off-by: Borislav Petkov (AMD) <bp@alien8.de>
    Link: https://lore.kernel.org/r/20230227160917.107820-1-ubizjak@gmail.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d642a198b748647a6a4f797e525432cd324f3bf0
Author: Chen-Yu Tsai <wenst@chromium.org>
Date:   Tue Mar 28 11:10:17 2023 +0800

    thermal/drivers/mediatek/lvts_thermal: Fix sensor 1 interrupt status bitmask

    [ Upstream commit 13f03bcd02e4b0498c8ccb066b4eddf61dee6681 ]

    The binary representation for sensor 1 interrupt status was incorrectly
    assembled, when compared to the full table given in the same comment
    section. The conversion into hex was also incorrect, leading to
    incorrect interrupt status bitmask for sensor 1. This would cause the
    driver to incorrectly identify changes for sensor 1, when in fact it
    was sensor 0, or a sensor access time out.

    Fix the binary and hex representations in the comments, and the actual
    bitmask macro.

    Fixes: f5f633b18234 ("thermal/drivers/mediatek: Add the Low Voltage Thermal Sensor
driver")
    Signed-off-by: Chen-Yu Tsai <wenst@chromium.org>
    Signed-off-by: Daniel Lezcano <daniel.lezcano@linaro.org>
    Link: https://lore.kernel.org/r/20230328031017.1360976-1-wenst@chromium.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 998ea24f19bd79bc74aa983e3c8d8475fddfb557
Author: Geert Uytterhoeven <geert+renesas@glider.be>
Date:   Mon Mar 6 10:06:33 2023 +0100

    drm/msm/dpu: Fix bit-shifting UB in DPU_HW_VER() macro

    [ Upstream commit 4760be481dc075cd13f95f4650f5d5b53b4b336d ]

    With gcc-5 and CONFIG_UBSAN_SHIFT=y:

        drivers/gpu/drm/msm/msm_mdss.c: In function 'msm_mdss_enable':
        drivers/gpu/drm/msm/msm_mdss.c:296:2: error: case label does not reduce to an
integer constant
          case DPU_HW_VER_800:
          ^
        drivers/gpu/drm/msm/msm_mdss.c:299:2: error: case label does not reduce to an
integer constant
          case DPU_HW_VER_810:
          ^
        drivers/gpu/drm/msm/msm_mdss.c:300:2: error: case label does not reduce to an
integer constant
          case DPU_HW_VER_900:
          ^

    This happens because for major revisions 8 or greather, the non-sign bit
```

```
    of the major revision number is shifted into bit 31 of a signed integer,
    which is undefined behavior.

    Fix this by casting the major revision number to unsigned int.

    Fixes: efcd0107727c4f04 ("drm/msm/dpu: add support for SM8550")
    Fixes: 4a352c2fc15aec1e ("drm/msm/dpu: Introduce SC8280XP")
    Fixes: 100d7ef6995d1f86 ("drm/msm/dpu: add support for SM8450")
    Signed-off-by: Geert Uytterhoeven <geert+renesas@glider.be>
    Reviewed-by: Randy Dunlap <rdunlap@infradead.org>
    Reviewed-by: Rob Clark <robdclark@gmail.com>
    Patchwork: https://patchwork.freedesktop.org/patch/525152/
    Link: https://lore.kernel.org/r/20230306090633.65918-1-geert+renesas@glider.be
    Signed-off-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3a4e6dc2a6f659ac53a0d280832677b6ecb69944
Author: Johan Hovold <johan+linaro@kernel.org>
Date:   Mon Mar 6 11:07:14 2023 +0100

    Revert "drm/msm: Fix failure paths in msm_drm_init()"

    [ Upstream commit 652eadfde81031c7b3d8b21bdbcfdd6eb217dec8 ]

    This reverts commit 8636500300a01740d92b345c680b036b94555b1b.

    A recent commit tried to address a drm device leak in the early
    msm_drm_uninit() error paths but ended up making things worse.

    Specifically, it moved the drm device reference put in msm_drm_uninit()
    to msm_drm_init() which means that the drm would now be leaked on normal
    unbind.

    For reasons that were never spelled out, it also added kms NULL pointer
    checks to a couple of helper functions that had nothing to do with the
    paths modified by the patch.

    Instead of trying to salvage this incrementally, let's revert the bad
    commit so that clean and backportable fixes can be added in its place.

    Fixes: 8636500300a0 ("drm/msm: Fix failure paths in msm_drm_init()")
    Cc: Akhil P Oommen <quic_akhilpo@quicinc.com>
    Signed-off-by: Johan Hovold <johan+linaro@kernel.org>
    Reviewed-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Patchwork: https://patchwork.freedesktop.org/patch/525092/
    Link: https://lore.kernel.org/r/20230306100722.28485-3-johan+linaro@kernel.org
    Signed-off-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9078b434587722a6f2958dc1d536af6e39634db9
Author: Johan Hovold <johan+linaro@kernel.org>
Date:   Mon Mar 6 11:07:13 2023 +0100

    Revert "drm/msm: Add missing check and destroy for alloc_ordered_workqueue"

    [ Upstream commit dfa70344d1b5f5ff08525a8c872c8dd5e82fc5d9 ]

    This reverts commit 643b7d0869cc7f1f7a5ac7ca6bd25d88f54e31d0.

    A recent patch that tried to fix up the msm_drm_init() paths with
    respect to the workqueue but only ended up making things worse:

    First, the newly added calls to msm_drm_uninit() on early errors would
    trigger NULL-pointer dereferences, for example, as the kms pointer would
    not have been initialised. (Note that these paths were also modified by
```

```
    a second broken error handling patch which in effect cancelled out this
    part when merged.)

    Second, the newly added allocation sanity check would still leak the
    previously allocated drm device.

    Instead of trying to salvage what was badly broken (and clearly not
    tested), let's revert the bad commit so that clean and backportable
    fixes can be added in its place.

    Fixes: 643b7d0869cc ("drm/msm: Add missing check and destroy for
alloc_ordered_workqueue")
    Cc: Jiasheng Jiang <jiasheng@iscas.ac.cn>
    Signed-off-by: Johan Hovold <johan+linaro@kernel.org>
    Reviewed-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Patchwork: https://patchwork.freedesktop.org/patch/525107/
    Link: https://lore.kernel.org/r/20230306100722.28485-2-johan+linaro@kernel.org
    Signed-off-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cdc042430ea9e07f77ce05a9d29e227dbdecc733
Author: Douglas Anderson <dianders@chromium.org>
Date:   Wed Mar 29 14:33:54 2023 -0700

    regulator: core: Avoid lockdep reports when resolving supplies

    [ Upstream commit cba6cfdc7c3f1516f0d08ddfb24e689af0932573 ]

    An automated bot told me that there was a potential lockdep problem
    with regulators. This was on the chromeos-5.15 kernel, but I see
    nothing that would be different downstream compared to upstream. The
    bot said:
      ============================================
      WARNING: possible recursive locking detected
      5.15.104-lockdep-17461-gc1e499ed6604 #1 Not tainted
      --------------------------------------------
      kworker/u16:4/115 is trying to acquire lock:
      ffffff8083110170 (regulator_ww_class_mutex){+.+.}-{3:3}, at:
create_regulator+0x398/0x7ec

      but task is already holding lock:
      ffffff808378e170 (regulator_ww_class_mutex){+.+.}-{3:3}, at:
ww_mutex_trylock+0x3c/0x7b8

      other info that might help us debug this:
       Possible unsafe locking scenario:

             CPU0
             ----
        lock(regulator_ww_class_mutex);
        lock(regulator_ww_class_mutex);

       *** DEADLOCK ***

       May be due to missing lock nesting notation

      4 locks held by kworker/u16:4/115:
       #0: ffffff808006a948 ((wq_completion)events_unbound){+.+.}-{0:0}, at:
process_one_work+0x520/0x1348
       #1: ffffffc00e0a7cc0 ((work_completion)(&entry->work)){+.+.}-{0:0}, at:
process_one_work+0x55c/0x1348
       #2: ffffff80828a2260 (&dev->mutex){....}-{3:3}, at:
__device_attach_async_helper+0xd0/0x2a4
       #3: ffffff808378e170 (regulator_ww_class_mutex){+.+.}-{3:3}, at:
ww_mutex_trylock+0x3c/0x7b8
```

```
       stack backtrace:
       CPU: 2 PID: 115 Comm: kworker/u16:4 Not tainted 5.15.104-lockdep-17461-gc1e499ed6604
  #1 9292e52fa83c0e23762b2b3aa1bacf5787a4d5da
       Hardware name: Google Quackingstick (rev0+) (DT)
       Workqueue: events_unbound async_run_entry_fn
       Call trace:
        dump_backtrace+0x0/0x4ec
        show_stack+0x34/0x50
        dump_stack_lvl+0xdc/0x11c
        dump_stack+0x1c/0x48
        __lock_acquire+0x16d4/0x6c74
        lock_acquire+0x208/0x750
        __mutex_lock_common+0x11c/0x11f8
        ww_mutex_lock+0xc0/0x440
        create_regulator+0x398/0x7ec
        regulator_resolve_supply+0x654/0x7c4
        regulator_register_resolve_supply+0x30/0x120
        class_for_each_device+0x1b8/0x230
        regulator_register+0x17a4/0x1f40
        devm_regulator_register+0x60/0xd0
        reg_fixed_voltage_probe+0x728/0xaec
        platform_probe+0x150/0x1c8
        really_probe+0x274/0xa20
        __driver_probe_device+0x1dc/0x3f4
        driver_probe_device+0x78/0x1c0
        __device_attach_driver+0x1ac/0x2c8
        bus_for_each_drv+0x11c/0x190
        __device_attach_async_helper+0x1e4/0x2a4
        async_run_entry_fn+0xa0/0x3ac
        process_one_work+0x638/0x1348
        worker_thread+0x4a8/0x9c4
        kthread+0x2e4/0x3a0
        ret_from_fork+0x10/0x20

    The problem was first reported soon after we made many of the
    regulators probe asynchronously, though nothing I've seen implies that
    the problems couldn't have also happened even without that.

    I haven't personally been able to reproduce the lockdep issue, but the
    issue does look somewhat legitimate. Specifically, it looks like in
    regulator_resolve_supply() we are holding a "rdev" lock while calling
    set_supply() -> create_regulator() which grabs the lock of a
    _different_ "rdev" (the one for our supply). This is not necessarily
    safe from a lockdep perspective since there is no documented ordering
    between these two locks.

    In reality, we should always be locking a regulator before the
    supplying regulator, so I don't expect there to be any real deadlocks
    in practice. However, the regulator framework in general doesn't
    express this to lockdep.

    Let's fix the issue by simply grabbing the two locks involved in the
    same way we grab multiple locks elsewhere in the regulator framework:
    using the "wound/wait" mechanisms.

    Fixes: eaa7995c529b ("regulator: core: avoid regulator_resolve_supply() race condition")
    Signed-off-by: Douglas Anderson <dianders@chromium.org>
    Link:
  https://lore.kernel.org/r/20230329143317.RFC.v2.2.I30d8e1ca10cfbe5403884cdd192253a2e063eb9e@
  changeid
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit fc0e6b5e46738e74b3616e421b39f3d9d54a3ac9
```

```
Author: Douglas Anderson <dianders@chromium.org>
Date:   Wed Mar 29 14:33:53 2023 -0700

    regulator: core: Consistently set mutex_owner when using ww_mutex_lock_slow()

    [ Upstream commit b83a1772be854f87602de14726737d3e5b06e1f4 ]

    When a codepath locks a rdev using ww_mutex_lock_slow() directly then
    that codepath is responsible for incrementing the "ref_cnt" and also
    setting the "mutex_owner" to "current".

    The regulator core consistently got that right for "ref_cnt" but
    didn't always get it right for "mutex_owner". Let's fix this.

    It's unlikely that this truly matters because the "mutex_owner" is
    only needed if we're going to do subsequent locking of the same
    rdev. However, even though it's not truly needed it seems less
    surprising if we consistently set "mutex_owner" properly.

    Fixes: f8702f9e4aa7 ("regulator: core: Use ww_mutex for regulators locking")
    Signed-off-by: Douglas Anderson <dianders@chromium.org>
    Link:
https://lore.kernel.org/r/20230329143317.RFC.v2.1.I4e9d433ea26360c06dd1381d091c82bb1a4ce843@
changeid
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1980db029e4a55eded4c2a8db373a654a3214f95
Author: Thomas Hellström <thomas.hellstrom@linux.intel.com>
Date:   Tue Apr 4 22:06:48 2023 +0200

    drm/ttm/pool: Fix ttm_pool_alloc error path

    [ Upstream commit 379989e7cbdc7aa7496a00ee286ec146c7599cf0 ]

    When hitting an error, the error path forgot to unmap dma mappings and
    could call set_pages_wb() on already uncached pages.

    Fix this by introducing a common ttm_pool_free_range() function that
    does the right thing.

    v2:
    - Simplify that common function (Christian König)
    v3:
    - Rename that common function to ttm_pool_free_range() (Christian König)

    Fixes: d099fc8f540a ("drm/ttm: new TT backend allocation pool v3")
    Cc: Christian König <christian.koenig@amd.com>
    Cc: Dave Airlie <airlied@redhat.com>
    Cc: Christian Koenig <christian.koenig@amd.com>
    Cc: Huang Rui <ray.huang@amd.com>
    Cc: dri-devel@lists.freedesktop.org
    Signed-off-by: Thomas Hellström <thomas.hellstrom@linux.intel.com>
    Reviewed-by: Christian König <christian.koenig@amd.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20230404200650.11043-2-
thomas.hellstrom@linux.intel.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1626ab19a4b340c1af00f1a59291a166d74339b8
Author: Cristian Ciocaltea <cristian.ciocaltea@collabora.com>
Date:   Sun Apr 2 12:50:51 2023 +0300

    arm64: dts: rockchip: Assign PLL_PPLL clock rate to 1.1 GHz on rk3588s

    [ Upstream commit b46a22dea7530cf530a45c6b84c03300083b813d ]
```

```
    The clock rate for PLL_PPLL has been wrongly initialized to 100 MHz
    instead of 1.1 GHz. Fix it.

    Fixes: c9211fa2602b ("arm64: dts: rockchip: Add base DT for rk3588 SoC")
    Reported-by: Sebastian Reichel <sebastian.reichel@collabora.com>
    Signed-off-by: Cristian Ciocaltea <cristian.ciocaltea@collabora.com>
    Link: https://lore.kernel.org/r/20230402095054.384739-3-cristian.ciocaltea@collabora.com
    Signed-off-by: Heiko Stuebner <heiko@sntech.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 44d2f79975584de4e83130985bd3729b58af559c
Author: Alan Previn <alan.previn.teres.alexis@intel.com>
Date:   Thu Mar 23 11:41:56 2023 -0700

    drm/i915/pxp: limit drm-errors or warning on firmware API failures

    [ Upstream commit 69e6dd149212cdd681201352a79e6634665004e8 ]

    MESA driver is creating protected context on every driver handle
    creation to query caps bits for app. So when running CI tests,
    they are observing hundreds of drm_errors when enabling PXP
    in .config but using SOC fusing or BIOS configuration that cannot
    support PXP sessions.

    The fixes tag referenced below was to resolve a related issue
    where we wanted to silence error messages, but that case was due
    to outdated IFWI (firmware) that definitely needed an upgrade and
    was, at that point, considered a one-off case as opposed to today's
    realization that default CI was enabling PXP in kernel config for
    all testing.

    So with this patch, let's strike a balance between issues that is
    critical but are root-caused from HW/platform gaps (louder drm-warn
    but just ONCE) vs other cases where it could also come from session
    state machine (which cannot be a WARN_ONCE since it can be triggered
    due to runtime operation events).

    Let's use helpers for these so as more functions are added in future
    features / HW (or as FW designers continue to bless upstreaming of
    the error codes and meanings), we only need to update the helpers.

    NOTE: Don't completely remove FW errors (via drm_debug) or else cusomer
    apps that really needs to know that content protection failed won't
    be aware of it.

    v2: - Add fixes tag (Trvtko)
    v3: - Break multi-line drm_dbg strings into separate drm_dbg (Daniele)
        - Fix couple of typecasting nits (Daniele)
    v4: - Unsuccessful PXP FW cmd due to platform configuration shouldn't
          use drm_WARN_once (Tvrtko), Switched to use drm_info_once.
    v5: - Added "reported-and-tested" by Eero.

    Reported-and-tested-by: Eero Tamminen <eero.t.tamminen@intel.com>
    Fixes: b762787bf767 ("drm/i915/pxp: Use drm_dbg if arb session failed due to fw
version")
    Signed-off-by: Alan Previn <alan.previn.teres.alexis@intel.com>
    Reviewed-by: Daniele Ceraolo Spurio <daniele.ceraolospurio@intel.com>
    Signed-off-by: Daniele Ceraolo Spurio <daniele.ceraolospurio@intel.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20230323184156.4140659-1-
alan.previn.teres.alexis@intel.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 916c47d9669749c8006bd3523c434b9bb4fabcf3
Author: Alan Previn <alan.previn.teres.alexis@intel.com>
```

```
Date:   Wed Jan 25 00:26:35 2023 -0800

    drm/i915/pxp: Invalidate all PXP fw sessions during teardown

    [ Upstream commit d374c047b38e9f1130308aae207dc44045cd5cac ]

    A gap was recently discovered where if an application did not
    invalidate all of the stream keys (intentionally or not), and the
    driver did a full PXP global teardown on the GT subsystem, we
    find that future session creation would fail on the security
    firmware's side of the equation. i915 is the entity that needs
    ensure the sessions' state across both iGT and security firmware
    are at a known clean point when performing a full global teardown.

    Architecturally speaking, i915 should inspect all active sessions
    and submit the invalidate-stream-key PXP command to the security
    firmware for each of them. However, for the upstream i915 driver
    we only support the arbitration session that can be created
    so that will be the only session we will cleanup.

    Signed-off-by: Alan Previn <alan.previn.teres.alexis@intel.com>
    Reviewed-by: Juston Li <justonli@chromium.org>
    Acked-by: Rodrigo Vivi <rodrigo.vivi@intel.com>
    Signed-off-by: Rodrigo Vivi <rodrigo.vivi@intel.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20230125082637.118970-5-
alan.previn.teres.alexis@intel.com
    Stable-dep-of: 69e6dd149212 ("drm/i915/pxp: limit drm-errors or warning on firmware API
failures")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f3526102dde11e7ef43b1f3bff8d5ab8b6cbce19
Author: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
Date:   Mon Apr 3 15:23:28 2023 +0200

    arm64: dts: qcom: sc8280xp: correct Soundwire wakeup interrupt name

    [ Upstream commit 894e258b6a38922f9860a20ca07cf2f745e3b090 ]

    The bindings expect second Soundwire interrupt to be "wakeup" (Linux
    driver takes by index):

      sc8280xp-crd.dtb: soundwire-controller@3330000: interrupt-names:1: 'wakeup' was
expected

    Fixes: c18773d162a6 ("arm64: dts: qcom: sc8280xp: add SoundWire and LPASS")
    Signed-off-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230403132328.61414-1-krzysztof.kozlowski@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1800106277910606cdc04c531fff2dbd1605c9b8
Author: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
Date:   Sun Mar 12 19:36:22 2023 +0100

    arm64: dts: qcom: apq8096-db820c: drop unit address from PMI8994 regulator

    [ Upstream commit ec57cbce1a6d9384f8ac1ff966b204dc262f4927 ]

    The PMIC regulators are not supposed to have unit addresses.

    Fixes: 2317b87a2a6f ("arm64: dts: qcom: db820c: Add vdd_gfx and tie it into mmcc")
    Signed-off-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
```

```
    Link: https://lore.kernel.org/r/20230312183622.460488-8-krzysztof.kozlowski@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8ee8ae6139c38e100afc46e391a3607f17995dd0
Author: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
Date:   Sun Mar 12 19:36:21 2023 +0100

    arm64: dts: qcom: msm8994-msft-lumia-octagon: drop unit address from PMI8994 regulator

    [ Upstream commit 7a202df0f3eed006e4a9e7c06d62cf67be56c14c ]

    The PMIC regulators are not supposed to have unit addresses.

    Fixes: 60b214effb80 ("arm64: dts: qcom: msm8994-octagon: Configure regulators")
    Signed-off-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230312183622.460488-7-krzysztof.kozlowski@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6c90668f70d6b1c0f4d1427df5b61427eff8f272
Author: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
Date:   Sun Mar 12 19:36:20 2023 +0100

    arm64: dts: qcom: msm8994-kitakami: drop unit address from PMI8994 regulator

    [ Upstream commit 3555dd528ba9c08d6ccd56239c695dbeac3b63e3 ]

    The PMIC regulators are not supposed to have unit addresses.

    Fixes: e9783584c9b7 ("arm64: dts: qcom: msm8994-kitakami: Add VDD_GFX regulator")
    Signed-off-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230312183622.460488-6-krzysztof.kozlowski@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f6fc05571a3967370363e2307c8989d2ce51ea11
Author: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
Date:   Sun Mar 12 19:36:19 2023 +0100

    arm64: dts: qcom: msm8998-oneplus-cheeseburger: revert "fix backlight pin function"

    [ Upstream commit 97b4fdc6b82d6d5cfb92a9b164540278720fb700 ]

    This reverts commit 46546f28825cf3a5ef6873b9cf947cd85c8a7258 because it
    mistakenly took PMIC pinctrl/GPIO as TLMM.  The TLMM pinctrl uses "gpio"
    function, but PMIC uses "normal", so original code was correct:

      msm8998-oneplus-cheeseburger.dtb: pmic@2: gpio@c000:button-backlight-state: 'oneOf'
conditional failed, one must be fixed:
        'gpio' is not one of ['normal', 'paired', 'func1', 'func2', 'dtest1', 'dtest2',
'dtest3', 'dtest4', 'func3', 'func4']

    Fixes: 46546f28825c ("arm64: dts: qcom: msm8998-oneplus-cheeseburger: fix backlight pin
function")
    Signed-off-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230312183622.460488-5-krzysztof.kozlowski@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cada40fca805683f644a200f0e5bf404c543289e
Author: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
Date:   Sun Mar 12 19:36:18 2023 +0100
```

```
    arm64: dts: qcom: sc7180-trogdor-pazquel: correct trackpad supply

    [ Upstream commit 24f39eec6a70768e7c2eb2f3d8158f45050ff75a ]

    The hid-over-i2c takes VDD, not VCC supply.  Fix copy-pasta from other
    boards which use elan,ekth3000 with valid VCC:

      sc7180-trogdor-pazquel360-lte.dtb: trackpad@15: 'vcc-supply' does not match any of the
regexes: 'pinctrl-[0-9]+'

    Fixes: fb69f6adaf88 ("arm64: dts: qcom: sc7180: Add pazquel dts files")
    Signed-off-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Reviewed-by: Stephen Boyd <swboyd@chromium.org>
    Reviewed-by: Douglas Anderson <dianders@chromium.org>
    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230312183622.460488-4-krzysztof.kozlowski@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a3ac770779ac1c655256c129483972fb3b65c78a
Author: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
Date:   Sun Mar 12 19:36:17 2023 +0100

    arm64: dts: qcom: sc7180-trogdor-lazor: correct trackpad supply

    [ Upstream commit 52e2996f253d82520011340d40dbc1c76ea79208 ]

    The hid-over-i2c takes VDD, not VCC supply.  Fix copy-pasta from other
    boards which use elan,ekth3000 with valid VCC:

      sc7180-trogdor-lazor-limozeen-nots-r4.dtb: trackpad@2c: 'vcc-supply' does not match
any of the regexes: 'pinctrl-[0-9]+'

    Fixes: 2c26adb8dbab ("arm64: dts: qcom: Add sc7180-lazor-limozeen skus")
    Signed-off-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Reviewed-by: Stephen Boyd <swboyd@chromium.org>
    Reviewed-by: Douglas Anderson <dianders@chromium.org>
    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230312183622.460488-3-krzysztof.kozlowski@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0cfbea0ca3c1ca8da9c12cc9788c9b9626594a42
Author: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
Date:   Sun Mar 12 19:36:16 2023 +0100

    arm64: dts: qcom: sc7280-herobrine-villager: correct trackpad supply

    [ Upstream commit de88b1759b35086d5e63736fb604ea2d06486b1a ]

    The hid-over-i2c takes VDD, not VCC supply.  Fix copy-pasta from other
    Herobrine boards which use elan,ekth3000 with valid VCC:

      sc7280-herobrine-villager-r1-lte.dtb: trackpad@2c: 'vcc-supply' does not match any of
the regexes: 'pinctrl-[0-9]+'

    Fixes: ee2a62116015 ("arm64: dts: qcom: sc7280: Add device tree for herobrine villager")
    Signed-off-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Reviewed-by: Stephen Boyd <swboyd@chromium.org>
    Reviewed-by: Douglas Anderson <dianders@chromium.org>
    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230312183622.460488-2-krzysztof.kozlowski@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 3ab0f5ddb761270b11d8c90b8550a59666cfc9bb
Author: Yang Yingliang <yangyingliang@huawei.com>
Date:   Sat Nov 26 15:33:15 2022 +0800

    gpu: host1x: Fix memory leak of device names

    [ Upstream commit 55879dad0f3ae8468444b42f785ad79eac05fe5b ]

    The device names allocated by dev_set_name() need be freed
    before module unloading, but they can not be freed because
    the kobject's refcount which was set in device_initialize()
    has not be decreased to 0.

    As comment of device_add() says, if it fails, use only
    put_device() drop the refcount, then the name will be
    freed in kobejct_cleanup().

    device_del() and put_device() can be replaced with
    device_unregister(), so call it to unregister the added
    successfully devices, and just call put_device() to the
    not added device.

    Add a release() function to device to avoid null release()
    function WARNING in device_release(), it's empty, because
    the context devices are freed together in
    host1x_memory_context_list_free().

    Fixes: 8aa5bcb61612 ("gpu: host1x: Add context device management code")
    Signed-off-by: Yang Yingliang <yangyingliang@huawei.com>
    Reviewed-by: Mikko Perttunen <mperttunen@nvidia.com>
    Signed-off-by: Thierry Reding <treding@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d922c295710fe531c599f882baa28ae2cfd6a618
Author: Yang Yingliang <yangyingliang@huawei.com>
Date:   Sat Nov 26 15:33:14 2022 +0800

    gpu: host1x: Fix potential double free if IOMMU is disabled

    [ Upstream commit 8466ff24a37a9a18fb935e90dda64f049131ae28 ]

    If context device has no IOMMU, the 'cdl->devs' is freed in
    error path, but host1x_memory_context_list_init() doesn't
    return an error code, so the module can be loaded successfully,
    when it's unloading, the host1x_memory_context_list_free() is
    called in host1x_remove(), it will cause double free. Set the
    'cdl->devs' to NULL after freeing it to avoid double free.

    Fixes: 8aa5bcb61612 ("gpu: host1x: Add context device management code")
    Signed-off-by: Yang Yingliang <yangyingliang@huawei.com>
    Reviewed-by: Mikko Perttunen <mperttunen@nvidia.com>
    Signed-off-by: Thierry Reding <treding@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a5089a3d8ca6c04fbda3fd920510dbd6d03297b8
Author: Li Yang <lidaxian@hust.edu.cn>
Date:   Fri Mar 31 17:55:44 2023 +0800

    soc: renesas: renesas-soc: Release 'chipid' from ioremap()

    [ Upstream commit fc187a46a8e682f0f1167b230792b88de01ceaa0 ]

    Smatch reports:
```

```
    drivers/soc/renesas/renesas-soc.c:536 renesas_soc_init() warn:
    'chipid' from ioremap() not released on lines: 475.

    If soc_dev_atrr allocation is failed, function renesas_soc_init()
    will return without releasing 'chipid' from ioremap().

    Fix this by adding function iounmap().

    Fixes: cb5508e47e60 ("soc: renesas: Add support for reading product revision for RZ/G2L
family")
    Signed-off-by: Li Yang <lidaxian@hust.edu.cn>
    Reviewed-by: Dan Carpenter <error27@gmail.com>
    Reviewed-by: Geert Uytterhoeven <geert+renesas@glider.be>
    Link: https://lore.kernel.org/r/20230331095545.31823-1-lidaxian@hust.edu.cn
    Signed-off-by: Geert Uytterhoeven <geert+renesas@glider.be>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit fd9d29db9f7be23b7d33918cf221c0870ef536c0
Author: Zhaoyang Li <lizhaoyang04@hust.edu.cn>
Date:   Mon Mar 27 19:54:22 2023 +0800

    soc: bcm: brcmstb: biuctrl: fix of_iomap leak

    [ Upstream commit c3fbced9af885a6f217fd95509a613d6590916ce ]

    Smatch reports:

    drivers/soc/bcm/brcmstb/biuctrl.c:291 setup_hifcpubiuctrl_regs() warn:
    'cpubiuctrl_base' from of_iomap() not released on lines: 291.

    This is because in setup_hifcpubiuctrl_regs(),
    cpubiuctrl_base is not released when handle error, which may cause a leak.
    To fix this, iounmap is added when handle error.

    Fixes: 22f7a9116eba ("soc: brcmstb: Correct CPU_CREDIT_REG offset for Brahma-B53 CPUs")
    Signed-off-by: Zhaoyang Li <lizhaoyang04@hust.edu.cn>
    Reviewed-by: Dan Carpenter <error27@gmail.com>
    Reviewed-by: Dongliang Mu <dzm91@hust.edu.cn>
    Link: https://lore.kernel.org/r/20230327115422.1536615-1-lizhaoyang04@hust.edu.cn
    Signed-off-by: Florian Fainelli <f.fainelli@gmail.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 60be03859d544c187f91bf611e68d3b4b029a965
Author: Conor Dooley <conor.dooley@microchip.com>
Date:   Tue Mar 7 20:22:52 2023 +0000

    mailbox: mpfs: switch to txdone_poll

    [ Upstream commit b5984a9844fc45cd301a28fb56f3de95f7e20f3c ]

    The system controller on PolarFire SoC has no interrupt to signify that
    the TX has been completed. The interrupt instead signals that a service
    requested by the mailbox client has succeeded. If a service fails, there
    will be no interrupt delivered.

    Switch to polling the busy register to determine whether transmission
    has completed.

    Fixes: 83d7b1560810 ("mbox: add polarfire soc system controller mailbox")
    Acked-by: Jassi Brar <jaswinder.singh@linaro.org>
    Tested-by: Valentina Fernandez <valentina.fernandezalanis@microchip.com>
    Signed-off-by: Conor Dooley <conor.dooley@microchip.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a54b77727a3ea5755bbbe515d14f79df670777bb
```

```
Author: Xinlei Lee <xinlei.lee@mediatek.com>
Date:   Wed Mar 29 14:43:23 2023 +0800

    drm/mediatek: dp: Change the aux retries times when receiving AUX_DEFER

    [ Upstream commit 9243d70e05c5989f84f840612965f96b524da925 ]

    DP 1.4a Section 2.8.7.1.5.6.1:
    A DP Source device shall retry at least seven times upon receiving
    AUX_DEFER before giving up the AUX transaction.

    The drm_dp_i2c_do_msg() function in the drm_dp_helper.c file will
    judge the status of the msg->reply parameter passed to aux_transfer
    for different processing.

    Fixes: f70ac097a2cf ("drm/mediatek: Add MT8195 Embedded DisplayPort driver")
    Signed-off-by: Xinlei Lee <xinlei.lee@mediatek.com>
    Link: https://patchwork.kernel.org/project/linux-mediatek/patch/1680072203-10394-1-git-
send-email-xinlei.lee@mediatek.com/
    Signed-off-by: Chun-Kuang Hu <chunkuang.hu@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 399bf936546c1e70b98f240b1bcd9b32efa89652
Author: Harshit Mogalapalli <harshit.m.mogalapalli@oracle.com>
Date:   Mon Mar 13 22:27:11 2023 -0700

    drm/lima/lima_drv: Add missing unwind goto in lima_pdev_probe()

    [ Upstream commit c5647cae2704e58d1c4e5fedbf63f11bca6376c9 ]

    Smatch reports:
    drivers/gpu/drm/lima/lima_drv.c:396 lima_pdev_probe() warn:
            missing unwind goto?

    Store return value in err and goto 'err_out0' which has
    lima_sched_slab_fini() before returning.

    Fixes: a1d2a6339961 ("drm/lima: driver for ARM Mali4xx GPUs")
    Signed-off-by: Harshit Mogalapalli <harshit.m.mogalapalli@oracle.com>
    Signed-off-by: Qiang Yu <yuq825@gmail.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20230314052711.4061652-1-
harshit.m.mogalapalli@oracle.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7dc69785666d918a0d8fcecdaa21c7ee777e71bf
Author: Jean-Philippe Brucker <jean-philippe@linaro.org>
Date:   Mon Mar 20 18:05:29 2023 +0000

    ACPI: VIOT: Initialize the correct IOMMU fwspec

    [ Upstream commit 47d26684185d09e083669bbbd0c465ab3493a51f ]

    When setting up DMA for a PCI device, we need to initialize its
    iommu_fwspec with all possible alias RIDs (such as PCI bridges). To do
    this we use pci_for_each_dma_alias() which calls
    viot_pci_dev_iommu_init(). This function incorrectly initializes the
    fwspec of the bridge instead of the device being configured. Fix it by
    passing the original device as context to pci_for_each_dma_alias().

    Fixes: 3cf485540e7b ("ACPI: Add driver for the VIOT table")
    Link: https://lore.kernel.org/all/Y8qzOKm6kvhGWG1T@myrica
    Reported-by: Eric Auger <eric.auger@redhat.com>
    Signed-off-by: Jean-Philippe Brucker <jean-philippe@linaro.org>
    Reviewed-by: Eric Auger <eric.auger@redhat.com>
    Tested-by: Eric Auger <eric.auger@redhat.com>
```

```
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7a38ccfd71223e9b28fcac58c8ca15f611f81c37
Author: Dhruva Gole <d-gole@ti.com>
Date:   Tue Mar 28 15:43:14 2023 +0300

    arm64: dts: ti: k3-am625-sk: Add ti,vbus-divider property to usbss1

    [ Upstream commit 4b71618cb772f20dfeeba826e9d1713a04f9687f ]

    The property "ti,vbus-divider" is needed for both usbss0 and usbss1 as
    both USB0 and USB1 have the same external voltage divider circuit.

    Fixes: 2d94dfc43885 ("arm64: dts: ti: k3-am625-sk: Add support for USB")
    Signed-off-by: Dhruva Gole <d-gole@ti.com>
    Signed-off-by: Roger Quadros <rogerq@kernel.org>
    Link: https://lore.kernel.org/r/20230328124315.123778-2-rogerq@kernel.org
    Signed-off-by: Nishanth Menon <nm@ti.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b078285eaebefe73a2cfdc8d78aa426f19fd82ff
Author: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
Date:   Wed Mar 1 10:55:15 2023 +0100

    arm64: dts: mediatek: mt8192-asurada: Fix voltage constraint for Vgpu

    [ Upstream commit ff4c868ba8df9dcd144ab4943a50adca1cf33ba2 ]

    The MT8192 SoC specifies a maximum voltage for the GPU's digital supply
    of 0.88V and the GPU OPPs are declaring a maximum voltage of 0.80V.

    In order to keep the GPU voltage in the safe range, change the maximum
    voltage for mt6315@7's vbuck1 to 0.80V as sending, for any mistake,
    1.193V would be catastrophic.

    Fixes: 3183cb62b033 ("arm64: dts: mediatek: asurada: Add SPMI regulators")
    Signed-off-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Reviewed-by: Chen-yu Tsai <wenst@chromium.org>
    Tested-by: Chen-Yu Tsai <wenst@chromium.org>
    Link: https://lore.kernel.org/r/20230301095523.428461-12-
angelogioacchino.delregno@collabora.com
    Signed-off-by: Matthias Brugger <matthias.bgg@gmail.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9dbd11bd8f5879d63ca048b960e840eed7fb52a5
Author: Bjorn Andersson <quic_bjorande@quicinc.com>
Date:   Thu Mar 23 15:33:43 2023 -0700

    cpufreq: qcom-cpufreq-hw: Revert adding cpufreq qos

    [ Upstream commit e2b47e585931a988c856fd4ba31e1296f749aee3 ]

    The OSM/EPSS hardware controls the frequency of each CPU cluster based
    on requests from the OS and various throttling events in the system.
    While throttling is in effect the related dcvs interrupt will be kept
    high. The purpose of the code handling this interrupt is to
    continuously report the thermal pressure based on the throttled
    frequency.

    The reasoning for adding QoS control to this mechanism is not entirely
    clear, but the introduction of commit 'c4c0efb06f17 ("cpufreq:
    qcom-cpufreq-hw: Add cpufreq qos for LMh")' causes the
    scaling_max_frequncy to be set to the throttled frequency. On the next
    iteration of polling, the throttled frequency is above or equal to the
```

```
    newly requested frequency, so the polling is stopped.

    With cpufreq limiting the max frequency, the hardware no longer report a
    throttling state and no further updates to thermal pressure or qos
    state are made.

    The result of this is that scaling_max_frequency can only go down, and
    the system becomes slower and slower every time a thermal throttling
    event is reported by the hardware.

    Even if the logic could be improved, there is no reason for software to
    limit the max freqency in response to the hardware limiting the max
    frequency. At best software will follow the reported hardware state, but
    typically it will cause slower backoff of the throttling.

    This reverts commit c4c0efb06f17fa4a37ad99e7752b18a5405c76dc.

    Fixes: c4c0efb06f17 ("cpufreq: qcom-cpufreq-hw: Add cpufreq qos for LMh")
    Reported-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Signed-off-by: Bjorn Andersson <quic_bjorande@quicinc.com>
    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Signed-off-by: Viresh Kumar <viresh.kumar@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ebdcae2c1b28a96cc1c88b5df175c9952fb1a842
Author: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
Date:   Fri Mar 24 18:11:30 2023 +0800

    cpufreq: mediatek: Raise proc and sram max voltage for MT7622/7623

    [ Upstream commit 0883426fd07e39355362e3f2eb9aee1a154dcaf6 ]

    During the addition of SRAM voltage tracking for CCI scaling, this
    driver got some voltage limits set for the vtrack algorithm: these
    were moved to platform data first, then enforced in a later commit
    6a17b3876bc8 ("cpufreq: mediatek: Refine mtk_cpufreq_voltage_tracking()")
    using these as max values for the regulator_set_voltage() calls.

    In this case, the vsram/vproc constraints for MT7622 and MT7623
    were supposed to be the same as MT2701 (and a number of other SoCs),
    but that turned out to be a mistake because the aforementioned two
    SoCs' maximum voltage for both VPROC and VPROC_SRAM is 1.36V.

    Fix that by adding new platform data for MT7622/7623 declaring the
    right {proc,sram}_max_volt parameter.

    Fixes: ead858bd128d ("cpufreq: mediatek: Move voltage limits to platform data")
    Fixes: 6a17b3876bc8 ("cpufreq: mediatek: Refine mtk_cpufreq_voltage_tracking()")
    Signed-off-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Signed-off-by: Jia-Wei Chang <jia-wei.chang@mediatek.com>
    Signed-off-by: Viresh Kumar <viresh.kumar@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5a4ba8fcdf7bb6492463e501c6f3fd5b8c2c984a
Author: Jia-Wei Chang <jia-wei.chang@mediatek.com>
Date:   Fri Mar 24 18:11:29 2023 +0800

    cpufreq: mediatek: raise proc/sram max voltage for MT8516

    [ Upstream commit d3296bb4cafd4bad4a5cf2eeab9d19cc94f9e30e ]

    Since the upper boundary of proc/sram voltage of MT8516 is 1300 mV,
    which is greater than the value of MT2701 1150 mV, we fix it by adding
    the corresponding platform data and specify proc/sram_max_volt to
    support MT8516.
```

```
    Signed-off-by: Jia-Wei Chang <jia-wei.chang@mediatek.com>
    Fixes: ead858bd128d ("cpufreq: mediatek: Move voltage limits to platform data")
    Fixes: 6a17b3876bc8 ("cpufreq: mediatek: Refine mtk_cpufreq_voltage_tracking()")
    Reported-by: Nick Hainke <vincent@systemli.org>
    Reviewed-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Signed-off-by: Viresh Kumar <viresh.kumar@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6587b2918cdbd43ddcca7ff76ff8617bfb48c613
Author: Jia-Wei Chang <jia-wei.chang@mediatek.com>
Date:   Fri Mar 24 18:11:28 2023 +0800

    cpufreq: mediatek: fix KP caused by handler usage after regulator_put/clk_put

    [ Upstream commit d51e106240bc755cbe59634b70d567c192b045b2 ]

    Any kind of failure in mtk_cpu_dvfs_info_init() will lead to calling
    regulator_put() or clk_put() and the KP will occur since the regulator/clk
    handlers are used after released in mtk_cpu_dvfs_info_release().

    To prevent the usage after regulator_put()/clk_put(), the regulator/clk
    handlers are addressed in a way of "Free the Last Thing Style".

    Signed-off-by: Jia-Wei Chang <jia-wei.chang@mediatek.com>
    Fixes: 4b9ceb757bbb ("cpufreq: mediatek: Enable clocks and regulators")
    Suggested-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Suggested-by: Dan Carpenter <error27@gmail.com>
    Signed-off-by: Viresh Kumar <viresh.kumar@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 25c3970772cd4af63ebe76de2cc92a6572c97ea1
Author: Jia-Wei Chang <jia-wei.chang@mediatek.com>
Date:   Fri Mar 24 18:11:27 2023 +0800

    cpufreq: mediatek: fix passing zero to 'PTR_ERR'

    [ Upstream commit d51c63230994f167126d9d8381011b4cb2b0ad22 ]

    In order to prevent passing zero to 'PTR_ERR' in
    mtk_cpu_dvfs_info_init(), we fix the return value of of_get_cci() using
    error pointer by explicitly casting error number.

    Signed-off-by: Jia-Wei Chang <jia-wei.chang@mediatek.com>
    Fixes: 0daa47325bae ("cpufreq: mediatek: Link CCI device to CPU")
    Reported-by: Dan Carpenter <error27@gmail.com>
    Reviewed-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Signed-off-by: Viresh Kumar <viresh.kumar@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b28bbfb7f667d65e57f4f6a92533a553fbcfc5ce
Author: Janne Grunau <j@jannau.net>
Date:   Tue Feb 14 15:07:22 2023 +0100

    arm64: dts: apple: t8103: Disable unused PCIe ports

    [ Upstream commit a0189fdfb73dac856b8fa9b9f9581e5099c9391f ]

    The PCIe ports are unused (without devices) so disable them instead of
    removing them.

    Fixes: 7c77ab91b33d ("arm64: dts: apple: Add missing M1 (t8103) devices")
    Signed-off-by: Janne Grunau <j@jannau.net>
    Reviewed-by: Sven Peter <sven@svenpeter.dev>
    Signed-off-by: Hector Martin <marcan@marcan.st>
```

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

commit 2af9741680b9876fb2e9a976410804de513e3d33
Author: Alexandre Torgue <alexandre.torgue@foss.st.com>
Date:   Mon Mar 20 18:03:31 2023 +0100

    ARM: dts: stm32: fix spi1 pin assignment on stm32mp15

    [ Upstream commit 1b9f0ec81af0012aae30aa3b4c711ad71d42e246 ]

    Bank A and B IOs can't be handled by the pin controller 'Z'. This patch
    assign spi1 pin definition to the correct controller.

    Fixes: 9ad65d245b7b ("ARM: dts: stm32: stm32mp15-pinctrl: add spi1-1 pinmux group")

    Signed-off-by: Alexandre Torgue <alexandre.torgue@foss.st.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5e12bca9c06c204e4b5fc6b7369bd032412a818f
Author: Jiucheng Xu <jiucheng.xu@amlogic.com>
Date:   Thu Feb 9 19:54:01 2023 +0800

    perf/amlogic: Fix config1/config2 parsing issue

    [ Upstream commit c61e5720f23273269cc67ffb2908cf9831c8ca9d ]

    The 3th argument of for_each_set_bit is incorrect, fix them.

    Fixes: 2016e2113d35 ("perf/amlogic: Add support for Amlogic meson G12 SoC DDR PMU
driver")
    Signed-off-by: Jiucheng Xu <jiucheng.xu@amlogic.com>
    Link: https://lore.kernel.org/r/20230209115403.521868-1-jiucheng.xu@amlogic.com
    Signed-off-by: Will Deacon <will@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 38455aa5c64260b38fcd2fd1d139f35a109a1eb4
Author: Ilkka Koskinen <ilkka@os.amperecomputing.com>
Date:   Wed Mar 1 09:55:40 2023 -0800

    perf/arm-cmn: Move overlapping wp_combine field

    [ Upstream commit f87e9114b5e590c2c6658ca21d7b714ca240bdd0 ]

    As eventid field was expanded to support new mesh versions, it started to
    overlap with wp_combine field. Move wp_combine to fix the issue.

    Fixes: 23760a014417 ("perf/arm-cmn: Add CMN-700 support")
    Signed-off-by: Ilkka Koskinen <ilkka@os.amperecomputing.com>
    Link: https://lore.kernel.org/r/20230301175540.19891-1-ilkka@os.amperecomputing.com
    Signed-off-by: Will Deacon <will@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4d97e9f3fc758506f23d6bf0e1c76a8f8b3d660e
Author: Cristian Marussi <cristian.marussi@arm.com>
Date:   Sun Mar 26 21:34:49 2023 +0100

    firmware: arm_scmi: Fix xfers allocation on Rx channel

    [ Upstream commit b2ccba9e8cdc6fb3985cc227844e7c6af309ffb1 ]

    Two distinct pools of xfer descriptors are allocated at initialization
    time: one (Tx) used to provide xfers to track commands and their replies
    (or delayed replies) and another (Rx) to pick xfers from to be used for
    processing notifications.
```

```
    Such pools, though, are allocated globally to be used by the whole SCMI
    instance, they are not allocated per-channel and as such the allocation of
    notifications xfers cannot be simply skipped if no Rx channel was found for
    the base protocol common channel, because there could be defined more
    optional per-protocol dedicated channels that instead support Rx channels.

    Change the conditional check to skip allocation for the notification pool
    only if no Rx channel has been detected on any per-channel protocol at all.

    Fixes: 4ebd8f6dea81 ("firmware: arm_scmi: Add receive buffer support for notifications")
    Signed-off-by: Cristian Marussi <cristian.marussi@arm.com>
    Link: https://lore.kernel.org/r/20230326203449.3492948-1-cristian.marussi@arm.com
    Signed-off-by: Sudeep Holla <sudeep.holla@arm.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7ba924c9e49cb651b5dee1583fc2baf5abe03784
Author: H. Nikolaus Schaller <hns@goldelico.com>
Date:   Fri Jan 13 22:11:51 2023 +0100

    ARM: dts: gta04: fix excess dma channel usage

    [ Upstream commit a622310f7f0185da02e42cdb06475f533efaae60 ]

    OMAP processors support 32 channels but there is no check or
    inspect this except booting a device and looking at dmesg reports
    of not available channels.

    Recently some more subsystems with DMA (aes1+2) were added filling
    the list of dma channels beyond the limit of 32 (even if other
    parameters indicate 96 or 128 channels). This leads to random
    subsystem failures i(e.g. mcbsp for audio) after boot or boot
    messages that DMA can not be initialized.

    Another symptom is that

    /sys/kernel/debug/dmaengine/summary

    has 32 entries and does not show all required channels.

    Fix by disabling unused (on the GTA04 hardware) mcspi1...4.
    Each SPI channel allocates 4 DMA channels rapidly filling
    the available ones.

    Disabling unused SPI modules on the OMAP3 SoC may also save
    some energy (has not been checked).

    Fixes: c312f066314e ("ARM: dts: omap3: Migrate AES from hwmods to sysc-omap2")
    Signed-off-by: H. Nikolaus Schaller <hns@goldelico.com>
    [re-enabled aes2, improved commit subject line]
    Signed-off-by: Andreas Kemnade <andreas@kemnade.info>
    Message-Id: <20230113211151.2314874-1-andreas@kemnade.info>
    Signed-off-by: Tony Lindgren <tony@atomide.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 162e9294b3281840751e71eae0d90c935eb6b4c8
Author: Dan Carpenter <error27@gmail.com>
Date:   Mon Feb 27 13:06:59 2023 +0300

    drm: rcar-du: Fix a NULL vs IS_ERR() bug

    [ Upstream commit 40f43730f43699ce8557e4fe59622d4f4b69f44a ]

    The drmm_encoder_alloc() function returns error pointers.  It never
    returns NULL.  Fix the check accordingly.
```

```
    Fixes: 7a1adbd23990 ("drm: rcar-du: Use drmm_encoder_alloc() to manage encoder")
    Signed-off-by: Dan Carpenter <error27@gmail.com>
    Reviewed-by: Kieran Bingham <kieran.bingham+renesas@ideasonboard.com>
    Reviewed-by: Laurent Pinchart <laurent.pinchart+renesas@ideasonboard.com>
    Signed-off-by: Laurent Pinchart <laurent.pinchart+renesas@ideasonboard.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit efe4e2fcdc82e931d09c621eb724ba11ace5e49f
Author: Neil Armstrong <neil.armstrong@linaro.org>
Date:   Fri Mar 24 10:28:49 2023 +0100

    arm64: dts: qcom: sm8450: fix pcie1 gpios properties name

    [ Upstream commit e57430d2483506f046e39bf8c61159dde88aede2 ]

    Add the final "s" to the pgio properties and fix the invalid "enable"
    name to the correct "wake", checked against the HDK8450 schematics.

    Fixes: bc6588bc25fb ("arm64: dts: qcom: sm8450: add PCIe1 root device")
    Signed-off-by: Neil Armstrong <neil.armstrong@linaro.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230323-topic-sm8450-upstream-dt-bindings-fixes-v2-4-
0ca1bea1a843@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 02d96ebc500d99583578b6eae39858287b49db51
Author: Georgii Kruglov <georgy.kruglov@yandex.ru>
Date:   Tue Mar 21 23:37:15 2023 +0300

    mmc: sdhci-of-esdhc: fix quirk to ignore command inhibit for data

    [ Upstream commit 0dd8316037a2a6d85b2be208bef9991de7b42170 ]

    If spec_reg is equal to 'SDHCI_PRESENT_STATE', esdhc_readl_fixup()
    fixes up register value and returns it immediately. As a result, the
    further block
    (spec_reg == SDHCI_PRESENT_STATE)
        &&(esdhc->quirk_ignore_data_inhibit == true),
    is never executed.

    The patch merges the second block into the first one.

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

    Fixes: 1f1929f3f2fa ("mmc: sdhci-of-esdhc: add quirk to ignore command inhibit for
data")
    Signed-off-by: Georgii Kruglov <georgy.kruglov@yandex.ru>
    Acked-by: Adrian Hunter <adrian.hunter@intel.com>
    Link: https://lore.kernel.org/r/20230321203715.3975-1-georgy.kruglov@yandex.ru
    Signed-off-by: Ulf Hansson <ulf.hansson@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c41a2a175e4aa6c4bcf2a14e2f834ddc5badf115
Author: Roger Pau Monne <roger.pau@citrix.com>
Date:   Wed Mar 22 12:13:29 2023 +0100

    ACPI: processor: Fix evaluating _PDC method when running as Xen dom0

    [ Upstream commit 073828e954459b883f23e53999d31e4c55ab9654 ]

    In ACPI systems, the OS can direct power management, as opposed to the
    firmware.  This OS-directed Power Management is called OSPM.  Part of
    telling the firmware that the OS going to direct power management is
    making ACPI "_PDC" (Processor Driver Capabilities) calls.  These _PDC
    methods must be evaluated for every processor object.  If these _PDC
```

```
    calls are not completed for every processor it can lead to
    inconsistency and later failures in things like the CPU frequency
    driver.

    In a Xen system, the dom0 kernel is responsible for system-wide power
    management.  The dom0 kernel is in charge of OSPM.  However, the
    number of CPUs available to dom0 can be different than the number of
    CPUs physically present on the system.

    This leads to a problem: the dom0 kernel needs to evaluate _PDC for
    all the processors, but it can't always see them.

    In dom0 kernels, ignore the existing ACPI method for determining if a
    processor is physically present because it might not be accurate.
    Instead, ask the hypervisor for this information.

    Fix this by introducing a custom function to use when running as Xen
    dom0 in order to check whether a processor object matches a CPU that's
    online.  Such checking is done using the existing information fetched
    by the Xen pCPU subsystem, extending it to also store the ACPI ID.

    This ensures that _PDC method gets evaluated for all physically online
    CPUs, regardless of the number of CPUs made available to dom0.

    Fixes: 5d554a7bb064 ("ACPI: processor: add internal processor_physically_present()")
    Signed-off-by: Roger Pau Monné <roger.pau@citrix.com>
    Reviewed-by: Juergen Gross <jgross@suse.com>
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dbc6cb4f01307d3737ada2a3efcd7b4b839b8329
Author: Lee Jones <lee@kernel.org>
Date:   Fri Mar 17 08:17:16 2023 +0000

    drm/amd/display/dc/dce60/Makefile: Fix previous attempt to silence known override-init
warnings

    [ Upstream commit 4082b9f5ead4966797dddcfef0905d59e5a83873 ]

    Fixes the following W=1 kernel build warning(s):

     drivers/gpu/drm/amd/amdgpu/../display/dc/dce60/dce60_resource.c:157:21: note: in
expansion of macro ‘mmCRTC1_DCFE_MEM_LIGHT_SLEEP_CNTL’
     drivers/gpu/drm/amd/amdgpu/../display/dc/dce/dce_transform.h:170:9: note: in expansion
of macro ‘SRI’
     drivers/gpu/drm/amd/amdgpu/../display/dc/dce60/dce60_resource.c:183:17: note: in
expansion of macro ‘XFM_COMMON_REG_LIST_DCE60’
     drivers/gpu/drm/amd/amdgpu/../display/dc/dce60/dce60_resource.c:188:17: note: in
expansion of macro ‘transform_regs’
     drivers/gpu/drm/amd/amdgpu/../include/asic_reg/dce/dce_6_0_d.h:722:43: warning:
initialized field overwritten [-Woverride-init]
     drivers/gpu/drm/amd/amdgpu/../display/dc/dce60/dce60_resource.c:157:21: note: in
expansion of macro ‘mmCRTC2_DCFE_MEM_LIGHT_SLEEP_CNTL’
     drivers/gpu/drm/amd/amdgpu/../display/dc/dce/dce_transform.h:170:9: note: in expansion
of macro ‘SRI’
     drivers/gpu/drm/amd/amdgpu/../display/dc/dce60/dce60_resource.c:183:17: note: in
expansion of macro ‘XFM_COMMON_REG_LIST_DCE60’
     drivers/gpu/drm/amd/amdgpu/../display/dc/dce60/dce60_resource.c:189:17: note: in
expansion of macro ‘transform_regs’
     drivers/gpu/drm/amd/amdgpu/../include/asic_reg/dce/dce_6_0_d.h:722:43: note: (near
initialization for ‘xfm_regs[2].DCFE_MEM_LIGHT_SLEEP_CN

    [100 lines snipped for brevity]

    Fixes: ceb3cf476a441 ("drm/amd/display/dc/dce60/Makefile: Ignore -Woverride-init
```

```
 warning")
     Cc: Harry Wentland <harry.wentland@amd.com>
     Cc: Leo Li <sunpeng.li@amd.com>
     Cc: Rodrigo Siqueira <Rodrigo.Siqueira@amd.com>
     Cc: Alex Deucher <alexander.deucher@amd.com>
     Cc: "Christian König" <christian.koenig@amd.com>
     Cc: "Pan, Xinhui" <Xinhui.Pan@amd.com>
     Cc: David Airlie <airlied@gmail.com>
     Cc: Daniel Vetter <daniel@ffwll.ch>
     Cc: Mauro Rossi <issor.oruam@gmail.com>
     Cc: amd-gfx@lists.freedesktop.org
     Cc: dri-devel@lists.freedesktop.org
     Signed-off-by: Lee Jones <lee@kernel.org>
     Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit 66b97bf5a875cddec54ed888ba059cb9ee1df0c8
 Author: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
 Date:   Sat Mar 4 14:03:15 2023 +0100

     arm64: dts: qcom: sm8350-microsoft-surface: fix USB dual-role mode property

     [ Upstream commit 0beda02e530f8fc571877939645cb20ade113027 ]

     The "dr_mode" is a property of USB DWC3 node, not the Qualcomm wrapper
     one:
       sm8350-microsoft-surface-duo2.dtb: usb@a6f8800: 'dr_mode' does not match any of the
 regexes: '^usb@[0-9a-f]+$', 'pinctrl-[0-9]+'

     Fixes: c16160cfa565 ("arm64: dts: qcom: add minimal DTS for Microsoft Surface Duo 2")
     Signed-off-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
     Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
     Signed-off-by: Bjorn Andersson <andersson@kernel.org>
     Link: https://lore.kernel.org/r/20230304130315.51595-2-krzysztof.kozlowski@linaro.org
     Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit 08ec7a7165396cbc2523b4b4fef34ab0b2f6a3b7
 Author: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
 Date:   Sat Mar 4 14:03:14 2023 +0100

     arm64: dts: qcom: sm8250-xiaomi-elish: fix USB maximum speed property

     [ Upstream commit 613f14a3a9d7e12a832ff822f85a33ad0ebee952 ]

     Fix typo in USB DWC3 node maximum speed property.

     Fixes: a41b617530bf ("arm64: dts: qcom: sm8250: Add device tree for Xiaomi Mi Pad 5
 Pro")
     Signed-off-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
     Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
     Signed-off-by: Bjorn Andersson <andersson@kernel.org>
     Link: https://lore.kernel.org/r/20230304130315.51595-1-krzysztof.kozlowski@linaro.org
     Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit 4b69c63f716cfda38e1210e65b68f67f6cee2ddf
 Author: Dionna Glaze <dionnaglaze@google.com>
 Date:   Tue Mar 7 20:24:48 2023 +0100

     virt/coco/sev-guest: Double-buffer messages

     [ Upstream commit 965006103a14703cc42043bbf9b5e0cdf7a468ad ]

     The encryption algorithms read and write directly to shared unencrypted
     memory, which may leak information as well as permit the host to tamper
     with the message integrity. Instead, copy whole messages in or out as
```

```
    needed before doing any computation on them.

    Fixes: d5af44dde546 ("x86/sev: Provide support for SNP guest request NAEs")
    Signed-off-by: Dionna Glaze <dionnaglaze@google.com>
    Signed-off-by: Borislav Petkov (AMD) <bp@alien8.de>
    Link: https://lore.kernel.org/r/20230214164638.1189804-3-dionnaglaze@google.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6fed8810f8cc17d4833b71e56e141a7ed2eb497d
Author: Adam Skladowski <a39.skl@gmail.com>
Date:   Tue Mar 14 23:17:17 2023 +0100

    drm: msm: adreno: Disable preemption on Adreno 510

    [ Upstream commit 010c8bbad2cb8c33c47963e29f051f1e917e45a5 ]

    Downstream driver appears to not support preemption on A510 target,
    trying to use one make device slow and fill log with rings related errors.
    Set num_rings to 1 to disable preemption.

    Suggested-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Fixes: e20c9284c8f2 ("drm/msm/adreno: Add support for Adreno 510 GPU")
    Signed-off-by: Adam Skladowski <a39.skl@gmail.com>
    Reviewed-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Patchwork: https://patchwork.freedesktop.org/patch/526898/
    Link: https://lore.kernel.org/r/20230314221757.13096-1-a39.skl@gmail.com
    Signed-off-by: Rob Clark <robdclark@chromium.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 90f4dcccd5d0de20752adfb6a5288bb40958d2f9
Author: Johan Hovold <johan+linaro@kernel.org>
Date:   Fri Mar 3 17:48:06 2023 +0100

    drm/msm/adreno: drop bogus pm_runtime_set_active()

    [ Upstream commit db7662d076c973072d788bd0e8130e04430307a1 ]

    The runtime PM status can only be updated while runtime PM is disabled.

    Drop the bogus pm_runtime_set_active() call that was made after enabling
    runtime PM and which (incidentally but correctly) left the runtime PM
    status set to 'suspended'.

    Fixes: 2c087a336676 ("drm/msm/adreno: Load the firmware before bringing up the
hardware")
    Signed-off-by: Johan Hovold <johan+linaro@kernel.org>
    Patchwork: https://patchwork.freedesktop.org/patch/524972/
    Link: https://lore.kernel.org/r/20230303164807.13124-4-johan+linaro@kernel.org
    Signed-off-by: Rob Clark <robdclark@chromium.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b79d18038ca1c48dcfc5314df0f63a1394788af6
Author: Kiran K <kiran.k@intel.com>
Date:   Thu Mar 16 12:47:15 2023 +0530

    ACPI: utils: Fix acpi_evaluate_dsm_typed() redefinition error

    [ Upstream commit 2c5a06e5505a716387a4d86f1f39de506836435a ]

    acpi_evaluate_dsm_typed() needs to be gaurded with CONFIG_ACPI to avoid
    a redefintion error when the stub is also enabled.

    In file included from ../drivers/bluetooth/btintel.c:13:
    ../include/acpi/acpi_bus.h:57:1: error: redefinition of 'acpi_evaluate_dsm_typed'
       57 | acpi_evaluate_dsm_typed(acpi_handle handle, const guid_t *guid,..
```

```
          | ^~~~~~~~~~~~~~~~~~~~~~
    In file included from ../drivers/bluetooth/btintel.c:12:
    ../include/linux/acpi.h:967:34: note: previous definition of
    'acpi_evaluate_dsm_typed' with type 'union acpi_object *(void *,
    const guid_t *, u64,  u64,  union acpi_object *, acpi_object_type)'
    {aka 'union acpi_object *(void *, const guid_t *, long long unsigned int,
    long long unsigned int,  union acpi_object *, unsigned int)'}
      967 | static inline union acpi_object
    *acpi_evaluate_dsm_typed(acpi_handle handle,

    Fixes: 1b94ad7ccc21 ("ACPI: utils: Add acpi_evaluate_dsm_typed() and acpi_check_dsm()
stubs")
    Signed-off-by: Kiran K <kiran.k@intel.com>
    [ rjw: Subject and changelog edits ]
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c6e379b1208cbdf940090fbdbb14af639fafe6ed
Author: Vignesh Raghavendra <vigneshr@ti.com>
Date:   Mon Mar 20 10:19:35 2023 +0530

    arm64: dts: ti: k3-am62a7: Correct L2 cache size to 512KB

    [ Upstream commit 438b8dc949bf45979c32553e96086ff1c6e2504e ]

    Per AM62Ax SoC datasheet[0] L2 cache is 512KB.

    [0] https://www.ti.com/lit/gpn/am62a7 Page 1.

    Fixes: 5fc6b1b62639 ("arm64: dts: ti: Introduce AM62A7 family of SoCs")
    Signed-off-by: Vignesh Raghavendra <vigneshr@ti.com>
    Link: https://lore.kernel.org/r/20230320044935.2512288-2-vigneshr@ti.com
    Signed-off-by: Nishanth Menon <nm@ti.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1fe7cc2e9b6810b5eb4770a3e8a5a089911b7179
Author: Vignesh Raghavendra <vigneshr@ti.com>
Date:   Mon Mar 20 10:19:34 2023 +0530

    arm64: dts: ti: k3-am625: Correct L2 cache size to 512KB

    [ Upstream commit 6974371cab1c488a53960945cb139b20ebb5f16b ]

    Per AM62x SoC datasheet[0] L2 cache is 512KB.

    [0] https://www.ti.com/lit/gpn/am625 Page 1.

    Fixes: f1d17330a5be ("arm64: dts: ti: Introduce base support for AM62x SoC")
    Signed-off-by: Vignesh Raghavendra <vigneshr@ti.com>
    Link: https://lore.kernel.org/r/20230320044935.2512288-1-vigneshr@ti.com
    Signed-off-by: Nishanth Menon <nm@ti.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6e2fc85946085e486baeca69996822617ee85be7
Author: Bagas Sanjaya <bagasdotme@gmail.com>
Date:   Tue Mar 7 11:35:26 2023 +0700

    accel: Link to compute accelerator subsystem intro

    [ Upstream commit 6d179f84f274a87da51f24ac3e9427221bbaed51 ]

    Commit 2c204f3d53218d ("accel: add dedicated minor for accelerator
    devices") adds link to accelerator nodes section of DRM internals doc
    (Documentation/gpu/drm-internals.rst), but the target doesn't exist.
    Instead, there is only an introduction doc for computer accelerator
```

```
    subsytem.

    Link to that doc until there is documentation of accelerator internals.

    Fixes: 2c204f3d53218d ("accel: add dedicated minor for accelerator devices")
    Signed-off-by: Bagas Sanjaya <bagasdotme@gmail.com>
    Reviewed-by: Jeffrey Hugo <quic_jhugo@quicinc.com>
    Signed-off-by: Oded Gabbay <ogabbay@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 19f36204dbe28bf4ec0149e87e9996a56af4e654
Author: Laurent Pinchart <laurent.pinchart@ideasonboard.com>
Date:   Sat Jan 14 22:46:50 2023 +0100

    media: max9286: Free control handler

    [ Upstream commit bfce6a12e5ba1edde95126aa06778027f16115d4 ]

    The control handler is leaked in some probe-time error paths, as well as
    in the remove path. Fix it.

    Fixes: 66d8c9d2422d ("media: i2c: Add MAX9286 driver")
    Signed-off-by: Laurent Pinchart <laurent.pinchart@ideasonboard.com>
    Reviewed-by: Niklas Söderlund <niklas.soderlund+renesas@ragnatech.se>
    Reviewed-by: Jacopo Mondi <jacopo.mondi@ideasonboard.com>
    Signed-off-by: Sakari Ailus <sakari.ailus@linux.intel.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit eeda150ba82179342ee9e1168656bf488245f050
Author: Adam Ford <aford173@gmail.com>
Date:   Sun Mar 19 07:55:24 2023 -0500

    drm/bridge: adv7533: Fix adv7533_mode_valid for adv7533 and adv7535

    [ Upstream commit ee0285e13455fdbce5de315bdbe91b5f198a2a06 ]

    When dynamically switching lanes was removed, the intent of the code
    was to check to make sure that higher speed items used 4 lanes, but
    it had the unintended consequence of removing the slower speeds for
    4-lane users.

    This attempts to remedy this by doing a check to see that the
    max frequency doesn't exceed the chip limit, and a second
    check to make sure that the max bit-rate doesn't exceed the
    number of lanes * max bit rate / lane.

    Fixes: 9a0cdcd6649b ("drm/bridge: adv7533: remove dynamic lane switching from adv7533
bridge")
    Reviewed-by: Robert Foss <rfoss@kernel.org>
    Signed-off-by: Adam Ford <aford173@gmail.com>
    Signed-off-by: Robert Foss <rfoss@kernel.org>
    Link: https://patchwork.freedesktop.org/patch/msgid/20230319125524.58803-1-
aford173@gmail.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c63b5eac9ecfa924ad93acc4d3bf0ae4e8b8de76
Author: Mukesh Ojha <quic_mojha@quicinc.com>
Date:   Thu Mar 16 20:44:26 2023 +0530

    firmware: qcom_scm: Clear download bit during reboot

    [ Upstream commit 781d32d1c9709fd25655c4e3e3e15370ae4ae4db ]

    During normal restart of a system download bit should
```

```
    be cleared irrespective of whether download mode is
    set or not.

    Fixes: 8c1b7dc9ba22 ("firmware: qcom: scm: Expose download-mode control")
    Signed-off-by: Mukesh Ojha <quic_mojha@quicinc.com>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/1678979666-551-1-git-send-email-quic_mojha@quicinc.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7b93ab60fe9ed04be0ff155bc30ad39dea23e22b
Author: Dan Carpenter <error27@gmail.com>
Date:   Tue Mar 7 11:00:23 2023 +0100

    media: av7110: prevent underflow in write_ts_to_decoder()

    [ Upstream commit eed9496a0501357aa326ddd6b71408189ed872eb ]

    The buf[4] value comes from the user via ts_play().  It is a value in
    the u8 range.  The final length we pass to av7110_ipack_instant_repack()
    is "len - (buf[4] + 1) - 4" so add a check to ensure that the length is
    not negative.  It's not clear that passing a negative len value does
    anything bad necessarily, but it's not best practice.

    With the new bounds checking the "if (!len)" condition is no longer
    possible or required so remove that.

    Fixes: fd46d16d602a ("V4L/DVB (11759): dvb-ttpci: Add TS replay capability")
    Signed-off-by: Dan Carpenter <error27@gmail.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e412fa3ab4d14490f18bb8ff62ea789a62af039e
Author: Ming Qian <ming.qian@nxp.com>
Date:   Thu Mar 2 06:34:10 2023 +0100

    media: amphion: decoder implement display delay enable

    [ Upstream commit ffa331d9bf9407655fc4c4d57dcc92ed2868e326 ]

    amphion vpu support a low latency mode,
    when V4L2_CID_MPEG_VIDEO_DEC_DISPLAY_DELAY_ENABLE is enabled,
    decoder can display frame immediately after it's decoded.
    Only h264 is support yet.

    Fixes: 6de8d628df6e ("media: amphion: add v4l2 m2m vpu decoder stateful driver")
    Signed-off-by: Ming Qian <ming.qian@nxp.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 22b72cad501fb75500cc60af4d92de3066fb6fc2
Author: Jiasheng Jiang <jiasheng@iscas.ac.cn>
Date:   Thu Feb 9 14:52:45 2023 +0100

    media: platform: mtk-mdp3: Add missing check and free for ida_alloc

    [ Upstream commit d00f592250782538cda87745607695b0fe27dcd4 ]

    Add the check for the return value of the ida_alloc in order to avoid
    NULL pointer dereference.
    Moreover, free allocated "ctx->id" if mdp_m2m_open fails later in order
    to avoid memory leak.

    Fixes: 61890ccaefaf ("media: platform: mtk-mdp3: add MediaTek MDP3 driver")
```

```
    Signed-off-by: Jiasheng Jiang <jiasheng@iscas.ac.cn>
    Reviewed-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0d09ce05724cfb3f5c5136893bec95305c641875
Author: Jiasheng Jiang <jiasheng@iscas.ac.cn>
Date:   Wed Feb 8 08:14:42 2023 +0100

    media: bdisp: Add missing check for create_workqueue

    [ Upstream commit 2371adeab717d8fe32144a84f3491a03c5838cfb ]

    Add the check for the return value of the create_workqueue
    in order to avoid NULL pointer dereference.

    Fixes: 28ffeebbb7bd ("[media] bdisp: 2D blitter driver using v4l2 mem2mem framework")
    Signed-off-by: Jiasheng Jiang <jiasheng@iscas.ac.cn>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 67bb7521b6420d81dab7538c0686f18f7d6d09f4
Author: Muralidhara M K <muralimk@amd.com>
Date:   Fri Jan 27 15:16:01 2023 +0000

    x86/MCE/AMD: Use an u64 for bank_map

    [ Upstream commit 4c1cdec319b9aadb65737c3eb1f5cb74bd6aa156 ]

    Thee maximum number of MCA banks is 64 (MAX_NR_BANKS), see

      a0bc32b3cacf ("x86/mce: Increase maximum number of banks to 64").

    However, the bank_map which contains a bitfield of which banks to
    initialize is of type unsigned int and that overflows when those bit
    numbers are >= 32, leading to UBSAN complaining correctly:

      UBSAN: shift-out-of-bounds in arch/x86/kernel/cpu/mce/amd.c:1365:38
      shift exponent 32 is too large for 32-bit type 'int'

    Change the bank_map to a u64 and use the proper BIT_ULL() macro when
    modifying bits in there.

      [ bp: Rewrite commit message. ]

    Fixes: a0bc32b3cacf ("x86/mce: Increase maximum number of banks to 64")
    Signed-off-by: Muralidhara M K <muralimk@amd.com>
    Signed-off-by: Borislav Petkov (AMD) <bp@alien8.de>
    Link: https://lore.kernel.org/r/20230127151601.1068324-1-muralimk@amd.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 812624d1bdf49a7c5166eff627e3e6f2ec00bbd2
Author: Sakari Ailus <sakari.ailus@linux.intel.com>
Date:   Fri Feb 24 16:00:46 2023 +0100

    media: v4l: subdev: Make link validation safer

    [ Upstream commit 55f1ecb1199000932cf82e357841cc7498ac904f ]

    Link validation currently accesses invalid pointers if the link passed to
    it is not between two sub-devices. This is of course a driver bug.

    Ignore the error but print a warning message, as this is how it used to
```

```
    work previously.

    Fixes: a6b995ed03ff ("media: subdev: use streams in v4l2_subdev_link_validate()")
    Reported-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Sakari Ailus <sakari.ailus@linux.intel.com>
    Tested-by: Hans de Goede <hdegoede@redhat.com>
    Reviewed-by: Tomi Valkeinen <tomi.valkeinen@ideasonboard.com>
    Reviewed-by: Laurent Pinchart <laurent.pinchart+renesas@ideasonboard.com>
    Signed-off-by: Mauro Carvalho Chehab <mchehab@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d27b19896965e79ab01e8627c4c823bd68750eb7
Author: Johan Hovold <johan+linaro@kernel.org>
Date:   Thu Mar 16 15:12:52 2023 +0100

    arm64: dts: qcom: sc8280xp: fix external display power domain

    [ Upstream commit cf386126aef92256245adc8be686f2df4b837013 ]

    Fix the external display controller nodes which erroneously described
    the controllers as belonging to CX rather than MMCX.

    Fixes: 19d3bb90754f ("arm64: dts: qcom: sc8280xp: Add USB-C-related DP blocks")
    Signed-off-by: Johan Hovold <johan+linaro@kernel.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230316141252.2436-1-johan+linaro@kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e004739fec43aebe18a7776c9c6e21ce11ca858b
Author: Stephan Gerhold <stephan.gerhold@kernkonzept.com>
Date:   Wed Mar 8 13:36:17 2023 +0100

    arm64: dts: qcom: msm8916: Fix tsens_mode unit address

    [ Upstream commit 608465f798bb3eaf6773a49b893902860858e294 ]

    The reg address of the tsens_mode nvmem cell is correct but the unit
    address does not match (0xec vs 0xef). Fix it. No functional change.

    Fixes: 24aafd041fb2 ("arm64: dts: qcom: msm8916: specify per-sensor calibration cells")
    Signed-off-by: Stephan Gerhold <stephan.gerhold@kernkonzept.com>
    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230308123617.101211-1-stephan.gerhold@kernkonzept.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b9ccf0fe4069747ba8d8d954074e806619f38f41
Author: Neil Armstrong <neil.armstrong@linaro.org>
Date:   Wed Mar 8 09:32:54 2023 +0100

    arm64: dts: qcom: sm8550: misc style fixes

    [ Upstream commit f03908b23f84ecd49f12facf4acf34c3ad24f27a ]

    Miscellaneous DT fixes to remove spurious blank line and enhance readability.

    Fixes: ffc50b2d3828 ("arm64: dts: qcom: Add base SM8550 dtsi")
    Fixes: d7da51db5b81 ("arm64: dts: qcom: sm8550: add display hardware devices")
    Signed-off-by: Neil Armstrong <neil.armstrong@linaro.org>
    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230308-topic-sm8550-upstream-dt-fixups-v1-3-
595b02067672@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 6eff4066db05d8698e683cb2884fba836c8906f4
Author: Neil Armstrong <neil.armstrong@linaro.org>
Date:   Wed Mar 8 09:32:53 2023 +0100

    arm64: dts: qcom: sm8550: fix qup_spi0_cs node

    [ Upstream commit 7629c7a525d163f2a3a08e260a69ff25163ab357 ]

    The node is incomplete and doesn't need a subnode, add the missing
    properties and move everything to the root of qup-spi0-cs-state node.

    Fixes: ffc50b2d3828 ("arm64: dts: qcom: Add base SM8550 dtsi")
    Signed-off-by: Neil Armstrong <neil.armstrong@linaro.org>
    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230308-topic-sm8550-upstream-dt-fixups-v1-2-
595b02067672@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 344ba0381b2a9ca716c957007bb3d34466f1e30b
Author: Manivannan Sadhasivam <mani@kernel.org>
Date:   Wed Mar 8 13:54:16 2023 +0530

    ARM: dts: qcom: sdx55: Fix the unit address of PCIe EP node

    [ Upstream commit 3b76b736cd9933ff88764ffec01cbd859c1475e7 ]

    Unit address of PCIe EP node should be 0x1c00000 as it has to match the
    first address specified in the reg property.

    This also requires sorting the node in the ascending order.

    Fixes: e6b69813283f ("ARM: dts: qcom: sdx55: Add support for PCIe EP")
    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Signed-off-by: Manivannan Sadhasivam <manivannan.sadhasivam@linaro.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230308082424.140224-6-manivannan.sadhasivam@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 93c9488618831a93e81ae4cfe06ce1e955941b15
Author: Adam Skladowski <a39.skl@gmail.com>
Date:   Thu Mar 2 13:30:49 2023 +0100

    arm64: dts: qcom: msm8976: Add and provide xo clk to rpmcc

    [ Upstream commit 4a2c9b9e1215c557c17a48e3fabe9b1674c1d608 ]

    In order for consumers of RPMCC XO clock to probe successfully
    their parent needs to be feed with reference clock to obtain proper rate,
    add fixed xo-board clock and supply it to rpmcc to make consumers happy.
    Frequency setting is left per board basis just like on other recent trees.

    Fixes: 0484d3ce0902 ("arm64: dts: qcom: Add DTS for MSM8976 and MSM8956 SoCs")
    Fixes: ff7f6d34ca07 ("arm64: dts: qcom: Add support for SONY Xperia X/X Compact")
    Signed-off-by: Adam Skladowski <a39.skl@gmail.com>
    Reviewed-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    [bjorn: Squashed the two patches]
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230302123051.12440-1-a39.skl@gmail.com
    Link: https://lore.kernel.org/r/20230302123051.12440-2-a39.skl@gmail.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit caed24802c6ee49af5d2795d2a90306fa72fc57e
Author: Manivannan Sadhasivam <mani@kernel.org>
Date:   Tue Feb 28 22:17:52 2023 +0530
```

```
    ARM: dts: qcom: ipq8064: Fix the PCI I/O port range

    [ Upstream commit 0b16b34e491629016109e56747ad64588074194b ]

    For 64KiB of the I/O region, the I/O ports of the legacy PCI devices are
    located in the range of 0x0 to 0x10000. Hence, fix the bogus PCI addresses
    (0x0fe00000, 0x31e00000, 0x35e00000) specified in the ranges property for
    I/O region.

    While at it, let's use the missing 0x prefix for the addresses.

    Fixes: 93241840b664 ("ARM: dts: qcom: Add pcie nodes for ipq8064")
    Reported-by: Arnd Bergmann <arnd@arndb.de>
    Link: https://lore.kernel.org/linux-arm-msm/7c5dfa87-41df-4ba7-b0e4-
72c8386402a8@app.fastmail.com/
    Signed-off-by: Manivannan Sadhasivam <manivannan.sadhasivam@linaro.org>
    Reviewed-by: Arnd Bergmann <arnd@arndb.de>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230228164752.55682-17-manivannan.sadhasivam@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 05e0081ebe4118854f7d4ac234089adc841ff3b0
Author: Manivannan Sadhasivam <mani@kernel.org>
Date:   Tue Feb 28 22:17:51 2023 +0530

    ARM: dts: qcom: ipq4019: Fix the PCI I/O port range

    [ Upstream commit 2540279e9a9e74fc880d1e4c83754ecfcbe290a0 ]

    For 1MiB of the I/O region, the I/O ports of the legacy PCI devices are
    located in the range of 0x0 to 0x100000. Hence, fix the bogus PCI address
    (0x40200000) specified in the ranges property for I/O region.

    While at it, let's use the missing 0x prefix for the addresses.

    Fixes: 187519403273 ("ARM: dts: ipq4019: Add a few peripheral nodes")
    Reported-by: Arnd Bergmann <arnd@arndb.de>
    Link: https://lore.kernel.org/linux-arm-msm/7c5dfa87-41df-4ba7-b0e4-
72c8386402a8@app.fastmail.com/
    Signed-off-by: Manivannan Sadhasivam <manivannan.sadhasivam@linaro.org>
    Reviewed-by: Arnd Bergmann <arnd@arndb.de>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230228164752.55682-16-manivannan.sadhasivam@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5ffed322945014094fc52fc51cb07ac2538a4e36
Author: Manivannan Sadhasivam <mani@kernel.org>
Date:   Tue Feb 28 22:17:49 2023 +0530

    arm64: dts: qcom: sm8350: Fix the PCI I/O port range

    [ Upstream commit cf4e716e9a38b32a922a88ac3c5b701137a663dd ]

    For 1MiB of the I/O region, the I/O ports of the legacy PCI devices are
    located in the range of 0x0 to 0x100000. Hence, fix the bogus PCI addresses
    (0x60200000, 0x40200000) specified in the ranges property for I/O region.

    While at it, let's use the missing 0x prefix for the addresses.

    Fixes: 6daee40678a0 ("arm64: dts: qcom: sm8350: add PCIe devices")
    Reported-by: Arnd Bergmann <arnd@arndb.de>
    Link: https://lore.kernel.org/linux-arm-msm/7c5dfa87-41df-4ba7-b0e4-
72c8386402a8@app.fastmail.com/
    Signed-off-by: Manivannan Sadhasivam <manivannan.sadhasivam@linaro.org>
```

```
    Reviewed-by: Arnd Bergmann <arnd@arndb.de>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230228164752.55682-14-manivannan.sadhasivam@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c0153be6436840dbedf49ec5bac250fa68b6c174
Author: Manivannan Sadhasivam <mani@kernel.org>
Date:   Tue Feb 28 22:17:48 2023 +0530

    arm64: dts: qcom: sm8450: Fix the PCI I/O port range

    [ Upstream commit f57903c8f4c77938eb71fc67e4652264a9fa14f9 ]

    For 1MiB of the I/O region, the I/O ports of the legacy PCI devices are
    located in the range of 0x0 to 0x100000. Hence, fix the bogus PCI addresses
    (0x60200000, 0x40200000) specified in the ranges property for I/O region.

    While at it, let's use the missing 0x prefix for the addresses.

    Fixes: bc6588bc25fb ("arm64: dts: qcom: sm8450: add PCIe1 root device")
    Fixes: 7b09b1b47335 ("arm64: dts: qcom: sm8450: add PCIe0 RC device")
    Reported-by: Arnd Bergmann <arnd@arndb.de>
    Link: https://lore.kernel.org/linux-arm-msm/7c5dfa87-41df-4ba7-b0e4-
72c8386402a8@app.fastmail.com/
    Signed-off-by: Manivannan Sadhasivam <manivannan.sadhasivam@linaro.org>
    Reviewed-by: Arnd Bergmann <arnd@arndb.de>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230228164752.55682-13-manivannan.sadhasivam@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b6e7ce037c45ed914317c307505daf64c305e5cc
Author: Manivannan Sadhasivam <mani@kernel.org>
Date:   Tue Feb 28 22:17:47 2023 +0530

    arm64: dts: qcom: sm8150: Fix the PCI I/O port range

    [ Upstream commit 422b110b9b0af0afd4a4b19e8fc3ceab9e71d842 ]

    For 1MiB of the I/O region, the I/O ports of the legacy PCI devices are
    located in the range of 0x0 to 0x100000. Hence, fix the bogus PCI addresses
    (0x60200000, 0x40200000) specified in the ranges property for I/O region.

    While at it, let's use the missing 0x prefix for the addresses.

    Fixes: a1c86c680533 ("arm64: dts: qcom: sm8150: Add PCIe nodes")
    Reported-by: Arnd Bergmann <arnd@arndb.de>
    Link: https://lore.kernel.org/linux-arm-msm/7c5dfa87-41df-4ba7-b0e4-
72c8386402a8@app.fastmail.com/
    Signed-off-by: Manivannan Sadhasivam <manivannan.sadhasivam@linaro.org>
    Reviewed-by: Arnd Bergmann <arnd@arndb.de>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230228164752.55682-12-manivannan.sadhasivam@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ab029266b07fbc6ce2b213cfa4169d455bf10473
Author: Manivannan Sadhasivam <mani@kernel.org>
Date:   Tue Feb 28 22:17:46 2023 +0530

    arm64: dts: qcom: sc8280xp: Fix the PCI I/O port range

    [ Upstream commit 89fe81c01715f81c3a7d371e9e5f7d7ae5bc557c ]

    For 1MiB of the I/O region, the I/O ports of the legacy PCI devices are
    located in the range of 0x0 to 0x100000. Hence, fix the bogus PCI addresses
    (0x30200000, 0x32200000, 0x34200000, 0x38200000, 0x3c200000) specified in
```

```
    the ranges property for I/O region.

    Fixes: 813e83157001 ("arm64: dts: qcom: sc8280xp/sa8540p: add PCIe2-4 nodes")
    Reported-by: Arnd Bergmann <arnd@arndb.de>
    Link: https://lore.kernel.org/linux-arm-msm/7c5dfa87-41df-4ba7-b0e4-
72c8386402a8@app.fastmail.com/
    Signed-off-by: Manivannan Sadhasivam <manivannan.sadhasivam@linaro.org>
    Reviewed-by: Arnd Bergmann <arnd@arndb.de>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230228164752.55682-11-manivannan.sadhasivam@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 531c5c22c3fa163fe8cd9f00f8a3aeefdf37d83c
Author: Manivannan Sadhasivam <mani@kernel.org>
Date:   Tue Feb 28 22:17:44 2023 +0530

    arm64: dts: qcom: sm8250: Fix the PCI I/O port range

    [ Upstream commit e115a4495db687898b8d91d4f16c2cf55bbf167c ]

    For 1MiB of the I/O region, the I/O ports of the legacy PCI devices are
    located in the range of 0x0 to 0x100000. Hence, fix the bogus PCI addresses
    (0x60200000, 0x40200000, 0x64200000) specified in the ranges property for
    I/O region.

    While at it, let's use the missing 0x prefix for the addresses.

    Fixes: e53bdfc00977 ("arm64: dts: qcom: sm8250: Add PCIe support")
    Reported-by: Arnd Bergmann <arnd@arndb.de>
    Link: https://lore.kernel.org/linux-arm-msm/7c5dfa87-41df-4ba7-b0e4-
72c8386402a8@app.fastmail.com/
    Signed-off-by: Manivannan Sadhasivam <manivannan.sadhasivam@linaro.org>
    Reviewed-by: Arnd Bergmann <arnd@arndb.de>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230228164752.55682-9-manivannan.sadhasivam@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8e50719c74ad9d222bfb8967cc9c87d1614ef88a
Author: Manivannan Sadhasivam <mani@kernel.org>
Date:   Tue Feb 28 22:17:43 2023 +0530

    arm64: dts: qcom: msm8996: Fix the PCI I/O port range

    [ Upstream commit cf0ac10feb17661987d0018eb9475dc03e2a2253 ]

    For 1MiB of the I/O region, the I/O ports of the legacy PCI devices are
    located in the range of 0x0 to 0x100000. Hence, fix the bogus PCI addresses
    (0x0c200000, 0x0d200000, 0x0e200000) specified in the ranges property for
    I/O region.

    While at it, let's also align the entries.

    Fixes: ed965ef89227 ("arm64: dts: qcom: msm8996: add support to pcie")
    Reported-by: Arnd Bergmann <arnd@arndb.de>
    Link: https://lore.kernel.org/linux-arm-msm/7c5dfa87-41df-4ba7-b0e4-
72c8386402a8@app.fastmail.com/
    Signed-off-by: Manivannan Sadhasivam <manivannan.sadhasivam@linaro.org>
    Reviewed-by: Arnd Bergmann <arnd@arndb.de>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230228164752.55682-8-manivannan.sadhasivam@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 680695f6ecd9386ddc7acefe895a7adde0731249
Author: Manivannan Sadhasivam <mani@kernel.org>
Date:   Tue Feb 28 22:17:42 2023 +0530
```

```
    arm64: dts: qcom: ipq6018: Fix the PCI I/O port range

    [ Upstream commit 75a6e1fdb351189f55097741e8460ca3f9b2883f ]

    For 64KiB of the I/O region, the I/O ports of the legacy PCI devices are
    located in the range of 0x0 to 0x10000. Hence, fix the bogus PCI address
    (0x20200000) specified in the ranges property for I/O region.

    While at it, let's use the missing 0x prefix for the addresses.

    Fixes: 095bbdd9a5c3 ("arm64: dts: qcom: ipq6018: Add pcie support")
    Reported-by: Arnd Bergmann <arnd@arndb.de>
    Link: https://lore.kernel.org/linux-arm-msm/7c5dfa87-41df-4ba7-b0e4-
72c8386402a8@app.fastmail.com/
    Signed-off-by: Manivannan Sadhasivam <manivannan.sadhasivam@linaro.org>
    Reviewed-by: Arnd Bergmann <arnd@arndb.de>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230228164752.55682-7-manivannan.sadhasivam@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e89e7c6a5abcbc7dc2539104cda4e1c912da79dc
Author: Manivannan Sadhasivam <mani@kernel.org>
Date:   Tue Feb 28 22:17:41 2023 +0530

    arm64: dts: qcom: ipq8074: Fix the PCI I/O port range

    [ Upstream commit e49eafefe5ab325e38dd074f2005076ffc271e54 ]

    For 64KiB of the I/O region, the I/O ports of the legacy PCI devices are
    located in the range of 0x0 to 0x10000. Hence, fix the bogus PCI addresses
    (0x10200000, 0x20200000) specified in the ranges property for I/O region.

    While at it, let's use the missing 0x prefix for the addresses and align
    them in a single line.

    Fixes: 33057e1672fe ("ARM: dts: ipq8074: Add pcie nodes")
    Reported-by: Arnd Bergmann <arnd@arndb.de>
    Link: https://lore.kernel.org/linux-arm-msm/7c5dfa87-41df-4ba7-b0e4-
72c8386402a8@app.fastmail.com/
    Signed-off-by: Manivannan Sadhasivam <manivannan.sadhasivam@linaro.org>
    Reviewed-by: Arnd Bergmann <arnd@arndb.de>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230228164752.55682-6-manivannan.sadhasivam@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b3170b6edfcbf20c083c8cc6dae4325f37e2c581
Author: Manivannan Sadhasivam <mani@kernel.org>
Date:   Tue Feb 28 22:17:40 2023 +0530

    arm64: dts: qcom: sm8550: Fix the PCI I/O port range

    [ Upstream commit 565c633940312b4eea7785a0a58e1944258c47e9 ]

    For 1MiB of the I/O region, the I/O ports of the legacy PCI devices are
    located in the range of 0x0 to 0x100000. Hence, fix the bogus PCI addresses
    (0x60200000, 0x40200000) specified in the ranges property for I/O region.

    While at it, let's use the missing 0x prefix for the addresses.

    Fixes: 7d1158c984d3 ("arm64: dts: qcom: sm8550: Add PCIe PHYs and controllers nodes")
    Reported-by: Arnd Bergmann <arnd@arndb.de>
    Link: https://lore.kernel.org/linux-arm-msm/7c5dfa87-41df-4ba7-b0e4-
72c8386402a8@app.fastmail.com/
    Signed-off-by: Manivannan Sadhasivam <manivannan.sadhasivam@linaro.org>
```

```
    Reviewed-by: Arnd Bergmann <arnd@arndb.de>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230228164752.55682-5-manivannan.sadhasivam@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 373a678b0d0c892b99b96aefa81abb50d24c768b
Author: Manivannan Sadhasivam <mani@kernel.org>
Date:   Tue Feb 28 22:17:39 2023 +0530

    arm64: dts: qcom: sc7280: Fix the PCI I/O port range

    [ Upstream commit 1d4743d6312582978966d38908b69085621b7693 ]

    For 1MiB of the I/O region, the I/O ports of the legacy PCI devices are
    located in the range of 0x0 to 0x100000. Hence, fix the bogus PCI address
    (0x40200000) specified in the ranges property for I/O region.

    Fixes: 92e0ee9f83b3 ("arm64: dts: qcom: sc7280: Add PCIe and PHY related nodes")
    Reported-by: Arnd Bergmann <arnd@arndb.de>
    Link: https://lore.kernel.org/linux-arm-msm/7c5dfa87-41df-4ba7-b0e4-
72c8386402a8@app.fastmail.com/
    Signed-off-by: Manivannan Sadhasivam <manivannan.sadhasivam@linaro.org>
    Reviewed-by: Arnd Bergmann <arnd@arndb.de>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230228164752.55682-4-manivannan.sadhasivam@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4d43086e38685147b5cf25c367055a314e0e8169
Author: Manivannan Sadhasivam <mani@kernel.org>
Date:   Tue Feb 28 22:17:38 2023 +0530

    arm64: dts: qcom: msm8998: Fix the PCI I/O port range

    [ Upstream commit c30a27dcfe4545edbda1578b3a63ed6147519cdd ]

    For 1MiB of the I/O region, the I/O ports of the legacy PCI devices are
    located in the range of 0x0 to 0x100000. Hence, fix the bogus PCI address
    (0x1b200000) specified in the ranges property for I/O region.

    Fixes: b84dfd175c09 ("arm64: dts: qcom: msm8998: Add PCIe PHY and RC nodes")
    Reported-by: Arnd Bergmann <arnd@arndb.de>
    Link: https://lore.kernel.org/linux-arm-msm/7c5dfa87-41df-4ba7-b0e4-
72c8386402a8@app.fastmail.com/
    Signed-off-by: Manivannan Sadhasivam <manivannan.sadhasivam@linaro.org>
    Reviewed-by: Arnd Bergmann <arnd@arndb.de>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230228164752.55682-3-manivannan.sadhasivam@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0d94524419cc4532629350be977fbdf8b766490f
Author: Manivannan Sadhasivam <mani@kernel.org>
Date:   Tue Feb 28 22:17:37 2023 +0530

    arm64: dts: qcom: sdm845: Fix the PCI I/O port range

    [ Upstream commit 67aa109eee654c76dcc100554e637fa64d5aa099 ]

    For 1MiB of the I/O region, the I/O ports of the legacy PCI devices are
    located in the range of 0x0 to 0x100000. Hence, fix the bogus PCI addresses
    (0x60200000, 0x40200000) specified in the ranges property for I/O region.

    While at it, let's use the missing 0x prefix for the addresses.

    Fixes: 42ad231338c1 ("arm64: dts: qcom: sdm845: Add second PCIe PHY and controller")
    Fixes: 5c538e09cb19 ("arm64: dts: qcom: sdm845: Add first PCIe controller and PHY")
```

```
    Reported-by: Arnd Bergmann <arnd@arndb.de>
    Link: https://lore.kernel.org/linux-arm-msm/7c5dfa87-41df-4ba7-b0e4-
72c8386402a8@app.fastmail.com/
    Signed-off-by: Manivannan Sadhasivam <manivannan.sadhasivam@linaro.org>
    Reviewed-by: Arnd Bergmann <arnd@arndb.de>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230228164752.55682-2-manivannan.sadhasivam@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dbc83445a6678d4698e525d4324adc1fc1c5c6b2
Author: Vincent Guittot <vincent.guittot@linaro.org>
Date:   Fri Jan 6 17:46:18 2023 +0100

    arm64: dts: qcom: sdm845: correct dynamic power coefficients

    [ Upstream commit 44750f153699b6e4f851a399287e5c8df208d696 ]

    While stressing EAS on my dragonboard RB3, I have noticed that LITTLE cores
    where never selected as the most energy efficient CPU whatever the
    utilization level of waking task.

    energy model framework uses its cost field to estimate the energy with
    the formula:

      nrg = cost of the selected OPP * utilization / CPU's max capacity

    which ends up selecting the CPU with lowest cost / max capacity ration
    as long as the utilization fits in the OPP's capacity.

    If we compare the cost of a little OPP with similar capacity of a big OPP
    like :
           OPP(kHz)   OPP capacity    cost     max capacity   cost/max capacity
    LITTLE 1766400    407             351114   407            863
    big    1056000    408             520267   1024           508

    This can be interpreted as the LITTLE core consumes 70% more than big core
    for the same compute capacity.

    According to [1], LITTLE consumes 10% less than big core for Coremark
    benchmark at those OPPs. If we consider that everything else stays
    unchanged, the dynamic-power-coefficient of LITTLE core should be
    only 53% of the current value: 290 * 53% = 154

    Set the dynamic-power-coefficient of CPU0-3 to 154 to fix the energy model.

    [1] https://github.com/kdrag0n/freqbench/tree/master/results/sdm845/main

    Fixes: 0e0a8e35d725 ("arm64: dts: qcom: sdm845: correct dynamic power coefficients")
    Signed-off-by: Vincent Guittot <vincent.guittot@linaro.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230106164618.1845281-1-vincent.guittot@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 46172eaba7d48a85f2f22e474060e8edac846e6d
Author: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
Date:   Wed Mar 8 13:59:05 2023 +0100

    arm64: dts: qcom: sc7280: fix EUD port properties

    [ Upstream commit a369c74243ca4ad60b9de0ac5c2207fb4c4117b8 ]

    Nodes with unit addresses must have also 'reg' property:

      sc7280-herobrine-crd.dtb: eud@88e0000: ports:port@0: 'reg' is a required property
```

```
    Fixes: 0b059979090d ("arm64: dts: qcom: sc7280: Add EUD dt node and dwc3 connector")
    Signed-off-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Reviewed-by: Souradeep Chowdhury <quic_schowdhu@quicinc.com>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230308125906.236885-10-krzysztof.kozlowski@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d1ed5404302b6d4cd6e6fc077c1d2d54c881d29f
Author: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
Date:   Wed Mar 8 13:58:58 2023 +0100

    arm64: dts: qcom: qdu1000: drop incorrect serial properties

    [ Upstream commit 9b8bfc443349c0c220438c6ac8839774210c9ba2 ]

    The serial node does not use/allow address/size cells:

      qdu1000-idp.dtb: geniqup@9c0000: serial@99c000: Unevaluated properties are not allowed
('#address-cells', '#size-cells' were unexpected)

    Fixes: 6bd20c54b589 ("arm64: dts: qcom: Add base QDU1000/QRU1000 DTSIs")
    Signed-off-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230308125906.236885-3-krzysztof.kozlowski@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 38618d092443f8bcadcc2a6301b200f0354e8818
Author: Jesse Taube <mr.bossman075@gmail.com>
Date:   Tue Mar 14 17:10:30 2023 -0400

    soc: canaan: Make K210_SYSCTL depend on CLK_K210

    [ Upstream commit 49f965b6fbca63904d7397ce96066fa992f401a3 ]

    CLK_K210 is no longer a dependency of SOC_CANAAN,
    but K210_SYSCTL depends on CLK_K210. This patch makes K210_SYSCTL
    depend on CLK_K210. Also fix whitespace errors.

    Reported-by: Randy Dunlap <rdunlap@infradead.org>
    Link: https://lore.kernel.org/all/42446784-a88b-df09-41e9-5f685b4df6ee@infradead.org
    Fixes: 3af577f9826f ("RISC-V: stop directly selecting drivers for SOC_CANAAN")
    Signed-off-by: Jesse Taube <Mr.Bossman075@gmail.com>
    Reviewed-by: Damien Le Moal <damien.lemoal@opensource.wdc.com>
    Signed-off-by: Conor Dooley <conor.dooley@microchip.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1e5acf1476265cf0b3823d1a4a2565b330fabac7
Author: Konrad Dybcio <konrad.dybcio@linaro.org>
Date:   Mon Feb 13 22:03:31 2023 +0100

    arm64: dts: qcom: msm8998: Fix stm-stimulus-base reg name

    [ Upstream commit b5d08f08377218b1d2ab4026e427a7788b271c8e ]

    The name stm-data-base comes from ancient (msm-3.10 or older)
    downstream kernels. Upstream uses stm-stimulus-base instead. Fix it.

    Fixes: 783abfa2249a ("arm64: dts: qcom: msm8998: Add Coresight support")
    Signed-off-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230213210331.2106877-1-konrad.dybcio@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit 10e81105ae38cfb666b69e4d83a5d98fbe538c68
Author: Abel Vesa <abel.vesa@linaro.org>
Date:   Wed Feb 8 20:00:20 2023 +0200

    arm64: dts: qcom: sm8550: Fix PCIe PHYs and controllers nodes

    [ Upstream commit 32734bbd1802efbd60ea4f0c3c1d5500bd0b20fe ]

    First, move the pinctrl related propeties out from SoC dtsi and into the
    board dts and add blank lines before status properties in the PHY nodes
    to be consistent with the rest of the nodes. Then drop the pipe clock
    from the controller nodes. Rename the aggre0 and aggre1 clocks to more
    generic noc_aggr, and then the cnoc_pcie_sf_axi to cnoc_sf_axi. Add the
    cpu-pcie interconnects to both controller nodes. Rename the pcie1 second
    reset to link_down and drop the unnecessary enable-gpios. Switch the aux
    clock to GCC_PCIE_1_PHY_AUX_CLK for the pcie1 PHY and drop the aux_phy
    from clock-names. Also rename the nocsr reset to phy_nocsr. With this
    changes we are now in line with the SC8280XP bindings.

    Fixes: 7d1158c984d3 ("arm64: dts: qcom: sm8550: Add PCIe PHYs and controllers nodes")
    Signed-off-by: Abel Vesa <abel.vesa@linaro.org>
    Reviewed-by: Johan Hovold <johan+linaro@kernel.org>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230208180020.2761766-12-abel.vesa@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dfa194d1531b2cd04036ec53a8b8c337cb4fefa9
Author: Rafał Miłecki <rafal@milecki.pl>
Date:   Tue Feb 28 15:44:00 2023 +0100

    arm64: dts: broadcom: bcmbca: bcm4908: fix procmon nodename

    [ Upstream commit f16a8294dd7a02c7ad042cd2e3acc5ea06698dc1 ]

    This fixes:
    arch/arm64/boot/dts/broadcom/bcmbca/bcm94908.dtb: syscon@280000: $nodename:0:
'syscon@280000' does not match '^([a-z][a-z0-9\\-]+-bus|bus|localbus|soc|axi|ahb|apb)(@.+)?
$'
            From schema: schemas/simple-bus.yaml

    Signed-off-by: Rafał Miłecki <rafal@milecki.pl>
    Link: https://lore.kernel.org/all/20230228144400.21689-3-zajec5@gmail.com/
    Signed-off-by: Florian Fainelli <f.fainelli@gmail.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0a4c841414b7021c897a6dcb7c8f6e1baa9c704e
Author: Rafał Miłecki <rafal@milecki.pl>
Date:   Tue Feb 28 15:43:59 2023 +0100

    arm64: dts: broadcom: bcmbca: bcm4908: fix LED nodenames

    [ Upstream commit 23be9f68f933adee8163b8efc9c6bff71410cc7c ]

    This fixes:
    arch/arm64/boot/dts/broadcom/bcmbca/bcm4908-asus-gt-ac5300.dtb: leds@800: 'led-lan@19',
'led-power@11', 'led-wan-red@12', 'led-wan-white@15', 'led-wps@14' do not match any of the
regexes: '^led@[a-f0-9]+$', 'pinctrl-[0-9]+'
            From schema: Documentation/devicetree/bindings/leds/leds-bcm63138.yaml

    Signed-off-by: Rafał Miłecki <rafal@milecki.pl>
    Link: https://lore.kernel.org/all/20230228144400.21689-2-zajec5@gmail.com/
    Signed-off-by: Florian Fainelli <f.fainelli@gmail.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5a61cc244842e71d7815463ad425a46e0ad4b2a0
```

```
Author: Rafał Miłecki <rafal@milecki.pl>
Date:   Tue Feb 28 15:43:58 2023 +0100

    arm64: dts: broadcom: bcmbca: bcm4908: fix NAND interrupt name

    [ Upstream commit 5cca02449490e767289bda38db1577e2c375c084 ]

    This fixes:
    arch/arm64/boot/dts/broadcom/bcmbca/bcm94908.dtb: nand-controller@1800: interrupt-
names:0: 'nand_ctlrdy' was expected
            From schema: Documentation/devicetree/bindings/mtd/brcm,brcmnand.yaml
    arch/arm64/boot/dts/broadcom/bcmbca/bcm94908.dtb: nand-controller@1800: Unevaluated
properties are not allowed ('interrupt-names' was unexpected)
            From schema: Documentation/devicetree/bindings/mtd/brcm,brcmnand.yaml

    Signed-off-by: Rafał Miłecki <rafal@milecki.pl>
    Link: https://lore.kernel.org/all/20230228144400.21689-1-zajec5@gmail.com/
    Signed-off-by: Florian Fainelli <f.fainelli@gmail.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 746dc7428b2731aff214e7b04302adcc3046a083
Author: Jayesh Choudhary <j-choudhary@ti.com>
Date:   Tue Mar 14 20:56:10 2023 +0530

    arm64: dts: ti: k3-j784s4-*: Add 'ti,sci-dev-id' for NAVSS nodes

    [ Upstream commit 436b288687176bf4d2c1cd25b86173e5a1649a60 ]

    TISCI device ID for main_navss and mcu_navss nodes are missing in
    the device tree. Add them.

    Fixes: 4664ebd8346a ("arm64: dts: ti: Add initial support for J784S4 SoC")
    Signed-off-by: Jayesh Choudhary <j-choudhary@ti.com>
    Signed-off-by: Nishanth Menon <nm@ti.com>
    Reviewed-by: Kamlesh Gurudasani <kamlesh@ti.com>
    Link: https://lore.kernel.org/r/20230314152611.140969-2-j-choudhary@ti.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3d2eeedd9337004edde1e626a6883d223cf6e393
Author: Bhavya Kapoor <b-kapoor@ti.com>
Date:   Fri Feb 3 13:07:24 2023 +0530

    arm64: dts: ti: k3-j721e-main: Remove ti,strobe-sel property

    [ Upstream commit 4f4b30a777d3e61603119297965343a37be36435 ]

    According to latest errata of J721e [1], (i2024) 'MMCSD: Peripherals
    Do Not Support HS400' which applies to MMCSD0 subsystem. Speed modes
    supported has been already updated but missed dropping 'ti,strobe-sel'
    property which is only required by HS400 speed mode.

    Thus, drop 'ti,strobe-sel' property from kernel dtsi for J721e SoC.

    [1] https://www.ti.com/lit/er/sprz455/sprz455.pdf

    Fixes: eb8f6194e807 ("arm64: dts: ti: k3-j721e-main: Update the speed modes supported
and their itap delay values for MMCSD subsystems")
    Signed-off-by: Bhavya Kapoor <b-kapoor@ti.com>
    Signed-off-by: Nishanth Menon <nm@ti.com>
    Reviewed-by: Diwakar Dhyani <d-dhyani@ti.com>
    Reviewed-by: Nitin Yadav <n-yadav@ti.com>
    Link: https://lore.kernel.org/r/20230203073724.29529-1-b-kapoor@ti.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cef6c5eed08d6046e60af2a19af7462e8eb34beb
```

```
Author: Devarsh Thakkar <devarsht@ti.com>
Date:   Tue Mar 14 15:16:45 2023 +0530

    arm64: dts: ti: k3-am62a7-sk: Fix DDR size to full 4GB

    [ Upstream commit a1bc0d6084dba8a31831c65318a8a8e46f00906f ]

    All revisions of AM62A7-SK board have 4GB LPDDR4 Micron
    MT53E2G32D4DE-046 AUT:B memory. Commit 38c4a08c820c ("arm64: dts: ti:
    Add support for AM62A7-SK") enabled just 2GB due to a schematics error
    in early revision of the board. Fix it by enabling full 4GB available on
    the platform.

    Design docs: https://www.ti.com/lit/zip/sprr459

    Fixes: 38c4a08c820c ("arm64: dts: ti: Add support for AM62A7-SK")
    Signed-off-by: Devarsh Thakkar <devarsht@ti.com>
    Signed-off-by: Nishanth Menon <nm@ti.com>
    Reviewed-by: Bryan Brattlof <bb@ti.com>
    Link: https://lore.kernel.org/r/20230314094645.3411599-1-devarsht@ti.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5b89f5fb399f31fcd95b1d3a9356a4f91bc4a1a7
Author: Nitin Yadav <n-yadav@ti.com>
Date:   Thu Feb 2 14:29:17 2023 +0530

    arm64: dts: ti: k3-am62-main: Fix GPIO numbers in DT

    [ Upstream commit 28c8f2189d80c8b37068c367e9864b5aa530f208 ]

    Fix number of gpio pins in main_gpio0 & main_gpio1
    DT nodes according to AM62x SK datasheet. The Link
    of datasheet is in the following line:
    https://www.ti.com/lit/ds/symlink/am625.pdf?ts=1673852494660

    Section: 6.3.10 GPIO (Page No. 63-67)

    Fixes: f1d17330a5be ("arm64: dts: ti: Introduce base support for AM62x SoC")
    Signed-off-by: Nitin Yadav <n-yadav@ti.com>
    Signed-off-by: Nishanth Menon <nm@ti.com>
    Reviewed-by: Bryan Brattlof <bb@ti.com>
    Link: https://lore.kernel.org/r/20230202085917.3044567-1-n-yadav@ti.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 04e8c557cd6a6c9d512a25e7a1d4ba96f46a7d97
Author: Douglas Anderson <dianders@chromium.org>
Date:   Mon Mar 13 11:18:19 2023 -0700

    regulator: core: Shorten off-on-delay-us for always-on/boot-on by time since booted

    [ Upstream commit 691c1fcda5351ed98a44610b7dccc0e3ee920020 ]

    This is very close to a straight revert of commit 218320fec294
    ("regulator: core: Fix off-on-delay-us for always-on/boot-on
    regulators"). We've identified that patch as causing a boot speed
    regression on sc7180-trogdor boards. While boot speed certainly isn't
    more important than making sure that power sequencing is correct,
    looking closely at the original change it doesn't seem to have been
    fully justified. It mentions "cycling issues" without describing
    exactly what the issues were. That means it's possible that the
    cycling issues were really a problem that should be fixed in a
    different way.

    Let's take a careful look at how we should handle regulators that have
    an off-on-delay and that are boot-on or always-on. Linux currently
```

```
    doesn't have any way to identify whether a GPIO regulator was already
    on when the kernel booted. That means that when the kernel boots we
    probe a regulator, see that it wants boot-on / always-on we, and then
    turn the regulator on. We could be in one of two cases when we do
    this:

    a) The regulator might have been left on by the bootloader and we're
       ensuring that it stays on.
    b) The regulator might have been left off by the bootloader and we're
       just now turning it on.

    For case a) we definitely don't need any sort of delay. For case b) we
    _might_ need some delay in case the bootloader turned the regulator
    off _right_ before booting the kernel. To get the proper delay for
    case b) then we can just assume a `last_off` of 0, which is what it
    gets initialized to by default.

    As per above, we can't tell whether we're in case a) or case b) so
    we'll assume the longer delay (case b). This basically puts the code
    to how it was before commit 218320fec294 ("regulator: core: Fix
    off-on-delay-us for always-on/boot-on regulators"). However, we add
    one important change: we make sure that the delay is actually honored
    if `last_off` is 0. Though the original "cycling issues" cited were
    vague, I'm hopeful that this important extra change will be enough to
    fix the issues that the initial commit mentioned.

    With this fix, I've confined that on a sc7180-trogdor board the delay
    at boot goes down from 500 ms to ~250 ms. That's not as good as the 0
    ms that we had prior to commit 218320fec294 ("regulator: core: Fix
    off-on-delay-us for always-on/boot-on regulators"), but it's probably
    safer because we don't know if the bootloader turned the regulator off
    right before booting.

    One note is that it's possible that we could be in a state that's not
    a) or b) if there are other issues in the kernel. The only one I can
    think of is related to pinctrl. If the pinctrl driver being used on a
    board isn't careful about avoiding glitches when setting up a pin then
    it's possible that setting up a pin could cause the regulator to "turn
    off" briefly immediately before the regulator probes. If this is
    indeed causing problems then the pinctrl driver should be fixed,
    perhaps in a similar way to what was done in commit d21f4b7ffc22
    ("pinctrl: qcom: Avoid glitching lines when we first mux to output")

    Fixes: 218320fec294 ("regulator: core: Fix off-on-delay-us for always-on/boot-on
regulators")
    Cc: Christian Kohlschütter <christian@kohlschutter.com>
    Signed-off-by: Douglas Anderson <dianders@chromium.org>
    Link:
https://lore.kernel.org/r/20230313111806.1.I2eaad872be0932a805c239a7c7a102233fb0b03b@changei
d
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit bd2573ee0f91c0e6d2bee8599110453e2909060e
Author: Marek Vasut <marex@denx.de>
Date:   Mon Feb 27 16:54:22 2023 +0100

    arm64: dts: imx8mp: Drop simple-bus from fsl,imx8mp-media-blk-ctrl

    [ Upstream commit 5a51e1f2b083423f75145c512ee284862ab33854 ]

    This block should not be compatible with simple-bus and misuse it that way.
    Instead, the driver should scan its subnodes and bind drivers to them.

    Reviewed-by: Alexander Stein <alexander.stein@ew.tq-group.com>
```

```
    Reviewed-by: Liu Ying <victor.liu@nxp.com>
    Tested-by: Alexander Stein <alexander.stein@ew.tq-group.com>
    Fixes: 94e6197dadc9 ("arm64: dts: imx8mp: Add LCDIF2 & LDB nodes")
    Signed-off-by: Marek Vasut <marex@denx.de>
    Signed-off-by: Shawn Guo <shawnguo@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b3fa5607391e42e74afdd16212bf6d6edafa0e15
Author: Konrad Dybcio <konrad.dybcio@linaro.org>
Date:   Mon Feb 20 13:08:31 2023 +0100

    ARM: dts: qcom-apq8064: Fix opp table child name

    [ Upstream commit b9745c275246a7e43c34d1b3be5ff9a9f3cf9305 ]

    The opp-320000000 name is rather misleading with the opp-hz value
    of 450 MHz. Fix it!

    Fixes: 8db0b6c7b636 ("ARM: dts: qcom: apq8064: Convert adreno from legacy gpu-pwrlevels
to opp-v2")
    Signed-off-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Reviewed-by: David Heidelberg <david@ixit.cz>
    Signed-off-by: Bjorn Andersson <andersson@kernel.org>
    Link: https://lore.kernel.org/r/20230220120831.1591820-1-konrad.dybcio@linaro.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7012ed21cad599cc1074ee14753e310e5a2a0355
Author: Qiuxu Zhuo <qiuxu.zhuo@intel.com>
Date:   Sat Feb 11 09:17:28 2023 +0800

    EDAC/skx: Fix overflows on the DRAM row address mapping arrays

    [ Upstream commit 71b1e3ba3fed5a34c5fac6d3a15c2634b04c1eb7 ]

    The current DRAM row address mapping arrays skx_{open,close}_row[]
    only support ranks with sizes up to 16G. Decoding a rank address
    to a DRAM row address for a 32G rank by using either one of the
    above arrays by the skx_edac driver, will result in an overflow on
    the array.

    For a 32G rank, the most significant DRAM row address bit (the
    bit17) is mapped from the bit34 of the rank address. Add this new
    mapping item to both arrays to fix the overflow issue.

    Fixes: 4ec656bdf43a ("EDAC, skx_edac: Add EDAC driver for Skylake")
    Reported-by: Feng Xu <feng.f.xu@intel.com>
    Tested-by: Feng Xu <feng.f.xu@intel.com>
    Signed-off-by: Qiuxu Zhuo <qiuxu.zhuo@intel.com>
    Signed-off-by: Tony Luck <tony.luck@intel.com>
    Link: https://lore.kernel.org/all/20230211011728.71764-1-qiuxu.zhuo@intel.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4bc0b6a7cf50da6251ee754a58ca906fe7fa7b86
Author: Vinod Polimera <quic_vpolimer@quicinc.com>
Date:   Thu Mar 2 22:03:07 2023 +0530

    drm/msm/disp/dpu: check for crtc enable rather than crtc active to release shared
resources

    [ Upstream commit b6975693846b562c4d3e0e60cc884affc5bdac00 ]

    According to KMS documentation, The driver must not release any shared
    resources if active is set to false but enable still true.

    Fixes: ccc862b957c6 ("drm/msm/dpu: Fix reservation failures in modeset")
```

```
    Signed-off-by: Vinod Polimera <quic_vpolimer@quicinc.com>
    Reviewed-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Patchwork: https://patchwork.freedesktop.org/patch/524726/
    Link: https://lore.kernel.org/r/1677774797-31063-5-git-send-email-
quic_vpolimer@quicinc.com
    Signed-off-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d1c04e338016ae2517c641806a831b1f3eee2bed
Author: Chen-Yu Tsai <wenst@chromium.org>
Date:   Thu Feb 2 12:57:34 2023 +0800

    drm/mediatek: dp: Only trigger DRM HPD events if bridge is attached

    [ Upstream commit 36b617f7e4ae663fcadd202ea061ca695ca75539 ]

    The MediaTek DisplayPort interface bridge driver starts its interrupts
    as soon as its probed. However when the interrupts trigger the bridge
    might not have been attached to a DRM device. As drm_helper_hpd_irq_event()
    does not check whether the passed in drm_device is valid or not, a NULL
    pointer passed in results in a kernel NULL pointer dereference in it.

    Check whether the bridge is attached and only trigger an HPD event if
    it is.

    Fixes: f70ac097a2cf ("drm/mediatek: Add MT8195 Embedded DisplayPort driver")
    Signed-off-by: Chen-Yu Tsai <wenst@chromium.org>
    Reviewed-by: Guillaume Ranquet <granquet@baylibre.com>
    Reviewed-by: AngeloGioacchino Del Regno <angelogioacchino.delregno@collabora.com>
    Reviewed-by: Matthias Brugger <matthias.bgg@gmail.com>
    Link: https://patchwork.kernel.org/project/linux-mediatek/patch/20230202045734.2773503-
1-wenst@chromium.org/
    Signed-off-by: Chun-Kuang Hu <chunkuang.hu@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1e6d0a0cdf5a7e5207f4de87c086876426415a46
Author: Lad Prabhakar <prabhakar.mahadev-lad.rj@bp.renesas.com>
Date:   Fri Feb 17 18:52:25 2023 +0000

    arm64: dts: renesas: r9a07g043: Update IRQ numbers for SSI channels

    [ Upstream commit 2a5c9891392dca47f6865a2add1986876e40849c ]

    From R01UH0968EJ0100 Rev.1.00 HW manual the interrupt numbers for SSI
    channels have been updated,

    SPI 329 - SSIF0 is now marked as reserved
    SPI 333 - SSIF1 is now marked as reserved
    SPI 335 - SSIF2 is now marked as reserved
    SPI 336 - SSIF2 is now marked as reserved
    SPI 341 - SSIF3 is now marked as reserved

    This patch drops the above IRQs from SoC DTSI.

    Fixes: 559f2b0708c70 ("arm64: dts: renesas: r9a07g043: Add SSI{1,2,3} nodes and fillup
the SSI0 stub node")
    Signed-off-by: Lad Prabhakar <prabhakar.mahadev-lad.rj@bp.renesas.com>
    Reviewed-by: Biju Das <biju.das.jz@bp.renesas.com>
    Reviewed-by: Geert Uytterhoeven <geert+renesas@glider.be>
    Link: https://lore.kernel.org/r/20230217185225.43310-5-prabhakar.mahadev-
lad.rj@bp.renesas.com
    Signed-off-by: Geert Uytterhoeven <geert+renesas@glider.be>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5cb7c1ccd664c2a4c1c65fce709867ac27ad92ec
```

```
Author: Lad Prabhakar <prabhakar.mahadev-lad.rj@bp.renesas.com>
Date:   Fri Feb 17 18:52:24 2023 +0000

    arm64: dts: renesas: r9a07g054: Update IRQ numbers for SSI channels

    [ Upstream commit c2b0dc0e83ef4a74cbe381fd0c84cea16cf067f0 ]

    From R01UH0936EJ0120 Rev.1.20 HW manual the interrupt numbers for SSI
    channels have been updated,

    SPI 329 - SSIF0 is now marked as reserved
    SPI 333 - SSIF1 is now marked as reserved
    SPI 335 - SSIF2 is now marked as reserved
    SPI 336 - SSIF2 is now marked as reserved
    SPI 341 - SSIF3 is now marked as reserved

    This patch drops the above IRQs from SoC DTSI.

    Fixes: cd0339ec25895c0b ("arm64: dts: renesas: r9a07g054: Add SSI{1,2,3} nodes and
fillup the SSI0 stub node")
    Signed-off-by: Lad Prabhakar <prabhakar.mahadev-lad.rj@bp.renesas.com>
    Reviewed-by: Biju Das <biju.das.jz@bp.renesas.com>
    Reviewed-by: Geert Uytterhoeven <geert+renesas@glider.be>
    Link: https://lore.kernel.org/r/20230217185225.43310-4-prabhakar.mahadev-
lad.rj@bp.renesas.com
    Signed-off-by: Geert Uytterhoeven <geert+renesas@glider.be>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4d46516a93f1dca8caeba8313c2e7c8d76a59d14
Author: Lad Prabhakar <prabhakar.mahadev-lad.rj@bp.renesas.com>
Date:   Fri Feb 17 18:52:24 2023 +0000

    arm64: dts: renesas: r9a07g044: Update IRQ numbers for SSI channels

    [ Upstream commit 5da750ddd96454757a3b467e968e3fb70bb12bc8 ]

    From R01UH0914EJ0120 Rev.1.20 HW manual the interrupt numbers for SSI
    channels have been updated,

    SPI 329 - SSIF0 is now marked as reserved
    SPI 333 - SSIF1 is now marked as reserved
    SPI 335 - SSIF2 is now marked as reserved
    SPI 336 - SSIF2 is now marked as reserved
    SPI 341 - SSIF3 is now marked as reserved

    This patch drops the above IRQs from SoC DTSI.

    Fixes: 92a341315afc9 ("arm64: dts: renesas: r9a07g044: Add SSI support")
    Signed-off-by: Lad Prabhakar <prabhakar.mahadev-lad.rj@bp.renesas.com>
    Reviewed-by: Biju Das <biju.das.jz@bp.renesas.com>
    Reviewed-by: Geert Uytterhoeven <geert+renesas@glider.be>
    Link: https://lore.kernel.org/r/20230217185225.43310-4-prabhakar.mahadev-
lad.rj@bp.renesas.com
    Signed-off-by: Geert Uytterhoeven <geert+renesas@glider.be>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3af5a0d53ef5a341a4f3e41004ac4a410da62fae
Author: Geert Uytterhoeven <geert+renesas@glider.be>
Date:   Thu Feb 16 16:30:32 2023 +0100

    arm64: dts: renesas: r8a774c0: Remove bogus voltages from OPP table

    [ Upstream commit 554edc3e9239bb81e61be9f0f5dbbeb528a69e72 ]

    According to the RZ/G Series, 2nd Generation Hardware User’s Manual
```

```
    Rev. 1.11, the System CPU cores on RZ/G2E do not have their own power
    supply, but use the common internal power supply (typical 1.03V).

    Hence remove the "opp-microvolt" properties from the Operating
    Performance Points table.  They are optional, and unused, when none of
    the CPU nodes is tied to a regulator using the "cpu-supply" property.

    Fixes: 231d8908a66fa98f ("arm64: dts: renesas: r8a774c0: Add OPPs table for cpu
devices")
    Signed-off-by: Geert Uytterhoeven <geert+renesas@glider.be>
    Link:
https://lore.kernel.org/r/8348e18a011ded94e35919cd8e17c0be1f9acf2f.1676560856.git.geert+rene
sas@glider.be
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9490f831c1a0841ddb8e06ad748a1ccbb2a65268
Author: Geert Uytterhoeven <geert+renesas@glider.be>
Date:   Thu Feb 16 16:30:31 2023 +0100

    arm64: dts: renesas: r8a77990: Remove bogus voltages from OPP table

    [ Upstream commit fb76b0fae3ca880363214e1dcd6513ab8bd529e7 ]

    According to the R-Car Series, 3rd Generation Hardware User's Manual
    Rev. 2.30, the System CPU cores on R-Car E3 do not have their own power
    supply, but use the common internal power supply (typical 1.03V).

    Hence remove the "opp-microvolt" properties from the Operating
    Performance Points table.  They are optional, and unused, when none of
    the CPU nodes is tied to a regulator using the "cpu-supply" property.

    Fixes: dd7188eb4ed128dc ("arm64: dts: renesas: r8a77990: Add OPPs table for cpu
devices")
    Signed-off-by: Geert Uytterhoeven <geert+renesas@glider.be>
    Link:
https://lore.kernel.org/r/9232578d9d395d529f64db3333a371e31327f459.1676560856.git.geert+rene
sas@glider.be
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 65305e8c0009a1933679dad5c8196060a10f3c8b
Author: Miaoqian Lin <linmq006@gmail.com>
Date:   Fri Jan 6 09:40:22 2023 +0400

    soc: ti: pm33xx: Fix refcount leak in am33xx_pm_probe

    [ Upstream commit 8f3c307b580a4a6425896007325bddefc36e8d91 ]

    wkup_m3_ipc_get() takes refcount, which should be freed by
    wkup_m3_ipc_put(). Add missing refcount release in the error paths.

    Fixes: 5a99ae0092fe ("soc: ti: pm33xx: AM437X: Add rtc_only with ddr in self-refresh
support")
    Signed-off-by: Miaoqian Lin <linmq006@gmail.com>
    Link: https://lore.kernel.org/r/20230106054022.947529-1-linmq006@gmail.com
    Signed-off-by: Nishanth Menon <nm@ti.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b47954341476c6593a5d1b1814e85d9ebf6df78a
Author: Nicolas Frayer <nfrayer@baylibre.com>
Date:   Fri Dec 30 01:14:04 2022 +0100

    soc: ti: k3-ringacc: Add try_module_get() to k3_dmaring_request_dual_ring()

    [ Upstream commit 5f1732a8683c1da8faaa90d6ffc3bd6d33013a58 ]
```

```
    When the k3 ring accelerator driver has been modified to add module build
    support, try_module_get() and module_put() have been added to update the
    module refcnt. One code path has not been updated and it has introduced
    an issue where the refcnt is decremented by module_put() in
    k3_ringacc_ring_free() without being incremented previously.
    Adding try_module_get() to k3_dmaring_request_dual_ring() ensures the
    refcnt is kept up to date.

    Fixes: c07f216a8b72 ("soc: ti: k3-ringacc: Allow the driver to be built as module")
    Signed-off-by: Nicolas Frayer <nfrayer@baylibre.com>
    Reviewed-by: Peter Ujfalusi <peter.ujfalusi@gmail.com>
    Link: https://lore.kernel.org/r/20221230001404.10902-1-nfrayer@baylibre.com
    Signed-off-by: Nishanth Menon <nm@ti.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a7eaf4147daf4aec0d0100a10708dff20b358b34
Author: Terry Bowman <terry.bowman@amd.com>
Date:   Mon Feb 6 08:18:30 2023 -0600

    tools/x86/kcpuid: Fix avx512bw and avx512lvl fields in Fn00000007

    [ Upstream commit 4e347bdf44c1fd4296a7b9657a2c0e1bd900fa50 ]

    Leaf Fn00000007 contains avx512bw at bit 26 and avx512vl at bit 28. This
    is incorrect per the SDM. Correct avx512bw to be bit 30 and avx512lvl to
    be bit 31.

    Fixes: c6b2f240bf8d ("tools/x86: Add a kcpuid tool to show raw CPU features")
    Signed-off-by: Terry Bowman <terry.bowman@amd.com>
    Signed-off-by: Borislav Petkov (AMD) <bp@alien8.de>
    Reviewed-by: Feng Tang <feng.tang@intel.com>
    Link: https://lore.kernel.org/r/20230206141832.4162264-2-terry.bowman@amd.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dd54ebf1a4e1b0e0f45424347fedb3c179482b5d
Author: Orlando Chamberlain <orlandoch.dev@gmail.com>
Date:   Fri Mar 3 22:34:25 2023 +1100

    drm/amdgpu: register a vga_switcheroo client for MacBooks with apple-gmux

    [ Upstream commit d37a3929ca0363ed1dce02b2772cd5bc547ca66d ]

    Commit 3840c5bcc245 ("drm/amdgpu: disentangle runtime pm and
    vga_switcheroo") made amdgpu only register a vga_switcheroo client for
    GPU's with PX, however AMD GPUs in dual gpu Apple Macbooks do need to
    register, but don't have PX. Instead of AMD's PX, they use apple-gmux.

    Use apple_gmux_detect() to identify these gpus, and
    pci_is_thunderbolt_attached() to ensure eGPUs connected to Dual GPU
    Macbooks don't register with vga_switcheroo.

    Fixes: 3840c5bcc245 ("drm/amdgpu: disentangle runtime pm and vga_switcheroo")
    Link: https://lore.kernel.org/amd-gfx/20230210044826.9834-10-orlandoch.dev@gmail.com/
    Signed-off-by: Orlando Chamberlain <orlandoch.dev@gmail.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9142b682272df5b4853d66dcdacec93bda61fd9a
Author: Neil Armstrong <neil.armstrong@linaro.org>
Date:   Tue Feb 7 16:07:57 2023 +0100

    arm64: dts: amlogic: meson-g12b-radxa-zero2: fix pwm clock names

    [ Upstream commit db217e84d0a3f4183ea5b6d5929e55b73128fcb2 ]
```

```
    Fixes the following bindings check error:
     - pwm@2000: clock-names: 'oneOf' conditional failed, one must be fixed:
           ['clkin4'] is too short
           'clkin4' is not one of ['clkin0', 'clkin1']
           'clkin0' was expected
     - pwm@7000: clock-names: 'oneOf' conditional failed, one must be fixed:
           ['clkin3'] is too short
           'clkin3' is not one of ['clkin0', 'clkin1']
           'clkin0' was expected
     - pwm@19000: clock-names: 'oneOf' conditional failed, one must be fixed:
           ['clkin2'] is too short
           'clkin2' is not one of ['clkin0', 'clkin1']
           'clkin0' was expected

    Fixes: d747e7f76a5f ("arm64: dts: meson: add support for Radxa Zero2")
    Reviewed-by: Martin Blumenstingl <martin.blumenstingl@googlemail.com>
    Link: https://lore.kernel.org/r/20230207-b4-amlogic-bindings-fixups-v2-v1-4-
93b7e50286e7@linaro.org
    Signed-off-by: Neil Armstrong <neil.armstrong@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9e787a1320af42765aabac369b7cc2e962d72e89
Author: Dom Cobley <popcornmix@gmail.com>
Date:   Fri Jan 27 16:40:52 2023 +0100

    drm/probe-helper: Cancel previous job before starting new one

    [ Upstream commit a8e47884f1906cd7440fafa056adc8817568e73e ]

    Currently we schedule a call to output_poll_execute from
    drm_kms_helper_poll_enable for 10s in future. Later we try to replace
    that in drm_helper_probe_single_connector_modes with a 0s schedule with
    delayed_event set.

    But as there is already a job in the queue this fails, and the immediate
    job we wanted with delayed_event set doesn't occur until 10s later.

    And that call acts as if connector state has changed, reprobing modes.
    This has a side effect of waking up a display that has been blanked.

    Make sure we cancel the old job before submitting the immediate one.

    Fixes: 162b6a57ac50 ("drm/probe-helper: don't lose hotplug event")
    Acked-by: Daniel Vetter <daniel@ffwll.ch>
    Signed-off-by: Dom Cobley <popcornmix@gmail.com>
    [Maxime: Switched to mod_delayed_work]
    Signed-off-by: Maxime Ripard <maxime@cerno.tech>
    Link: https://patchwork.freedesktop.org/patch/msgid/20230127154052.452524-1-
maxime@cerno.tech
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a62bfdbacf8fe47e5703570bbf59aca3d85ef0e7
Author: Maíra Canal <mcanal@igalia.com>
Date:   Thu Feb 2 09:55:17 2023 -0300

    drm/vgem: add missing mutex_destroy

    [ Upstream commit 7c18189b14b33c1fbf76480b1bd217877c086e67 ]

    vgem_fence_open() instantiates a mutex for a particular fence
    instance, but never destroys it by calling mutex_destroy() in
    vgem_fence_close().

    So, add the missing mutex_destroy() to guarantee proper resource
    destruction.
```

```
    Fixes: 407779848445 ("drm/vgem: Attach sw fences to exported vGEM dma-buf (ioctl)")
    Signed-off-by: Maíra Canal <mcanal@igalia.com>
    Reviewed-by: Stanislaw Gruszka <stanislaw.gruszka@linux.intel.com>
    Signed-off-by: Maíra Canal <mairacanal@riseup.net>
    Link: https://patchwork.freedesktop.org/patch/msgid/20230202125517.427976-1-
mcanal@igalia.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5ffe1167627be22c4b5fdf9474bf75eb2ddb8200
Author: Matt Roper <matthew.d.roper@intel.com>
Date:   Wed Feb 8 12:09:05 2023 -0800

    drm/i915/dg2: Drop one PCI ID

    [ Upstream commit 3a38be31ec82920a871963c086393bc0ba26a655 ]

    The bspec was recently updated to remove PCI ID 0x5698; this ID is
    actually reserved for future use and should not be treated as DG2-G11.

    Bspec: 44477
    Fixes: 8618b8489ba6 ("drm/i915: DG2 and ATS-M device ID updates")
    Signed-off-by: Matt Roper <matthew.d.roper@intel.com>
    Reviewed-by: Gustavo Sousa <gustavo.sousa@intel.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20230208200905.680865-1-
matthew.d.roper@intel.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1396802fb040ed8dfebdde2ba1b38f8238efa119
Author: Rob Clark <robdclark@chromium.org>
Date:   Thu Jan 19 15:17:34 2023 -0800

    drm/rockchip: Drop unbalanced obj unref

    [ Upstream commit 8ee3b0e85f6ccd9e6c527bc50eaba774c3bb18d0 ]

    In the error path, rockchip_drm_gem_object_mmap() is dropping an obj
    reference that it doesn't own.

    Fixes: 41315b793e13 ("drm/rockchip: use drm_gem_mmap helpers")
    Signed-off-by: Rob Clark <robdclark@chromium.org>
    Signed-off-by: Heiko Stuebner <heiko@sntech.de>
    Link: https://patchwork.freedesktop.org/patch/msgid/20230119231734.2884543-1-
robdclark@gmail.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8e7dd16741078989b9e5107587e9d4a5a446ce56
Author: Arnd Bergmann <arnd@arndb.de>
Date:   Thu Jan 26 17:37:54 2023 +0100

    accel/ivpu: PM: remove broken ivpu_dbg() statements

    [ Upstream commit 17ab1ea679be48d905559d968a7622f5f212de6e ]

    When CONFIG_PM is disabled, the driver fails to build:

    drivers/accel/ivpu/ivpu_pm.c: In function 'ivpu_rpm_get':
    drivers/accel/ivpu/ivpu_pm.c:240:84: error: 'struct dev_pm_info' has no member named
'usage_count'
      240 |         ivpu_dbg(vdev, RPM, "rpm_get count %d\n", atomic_read(&vdev->drm.dev-
>power.usage_count));
          |
^
    include/linux/dynamic_debug.h:223:29: note: in definition of macro
'__dynamic_func_call_cls'
```

```
      223 |                 func(&id, ##__VA_ARGS__);                       \
          |                                  ^~~~~~~~~~
    include/linux/dynamic_debug.h:249:9: note: in expansion of macro
  '_dynamic_func_call_cls'
      249 |         _dynamic_func_call_cls(_DPRINTK_CLASS_DFLT, fmt, func, ##__VA_ARGS__)
          |         ^~~~~~~~~~~~~~~~~~~~~
    include/linux/dynamic_debug.h:272:9: note: in expansion of macro '_dynamic_func_call'
      272 |         _dynamic_func_call(fmt, __dynamic_dev_dbg,              \
          |         ^~~~~~~~~~~~~~~~~~
    include/linux/dev_printk.h:155:9: note: in expansion of macro 'dynamic_dev_dbg'
      155 |         dynamic_dev_dbg(dev, dev_fmt(fmt), ##__VA_ARGS__)
          |         ^~~~~~~~~~~~~~~
    drivers/accel/ivpu/ivpu_drv.h:65:17: note: in expansion of macro 'dev_dbg'
       65 |                 dev_dbg((vdev)->drm.dev, "[%s] " fmt, #type, ##args);          \
          |                 ^~~~~~
    drivers/accel/ivpu/ivpu_pm.c:240:9: note: in expansion of macro 'ivpu_dbg'
      240 |         ivpu_dbg(vdev, RPM, "rpm_get count %d\n", atomic_read(&vdev->drm.dev-
  >power.usage_count));
          |         ^~~~~~~
```

    It would be possible to rework these statements to only conditionally print
    the reference counter, or to make the driver depend on CONFIG_PM, but my
    impression is that these are not actually needed at all if the driver generally
    works, or they could be put back when required. Just remove all four of these
    to make the driver build in all configurations.

    Fixes: 852be13f3bd3 ("accel/ivpu: Add PM support")
    Signed-off-by: Arnd Bergmann <arnd@arndb.de>
    Reviewed-by: Stanislaw Gruszka <stanislaw.gruszka@linux.intel.com>
    Signed-off-by: Jacek Lawrynowicz <jacek.lawrynowicz@linux.intel.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20230126163804.3648051-1-
  arnd@kernel.org
    Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 1bf559e9dc82e3843749dd574684633e0f9b8b9e
  Author: Jingbo Xu <jefflexu@linux.alibaba.com>
  Date: Fri Apr 14 14:18:10 2023 +0800

    erofs: fix potential overflow calculating xattr_isize

    [Upstream commit 1b3567a1969b26f709d82a874498c0754ea841c3]

    Given on-disk i_xattr_icount is 16 bits and xattr_isize is calculated
    from i_xattr_icount multiplying 4, xattr_isize has a theoretical maximum
    of 256K (64K * 4).

    Thus declare xattr_isize as unsigned int to avoid the potential overflow.

    Fixes: bfb8674dc044 ("staging: erofs: add erofs in-memory stuffs")
    Signed-off-by: Jingbo Xu <jefflexu@linux.alibaba.com>
    Reviewed-by: Gao Xiang <hsiangkao@linux.alibaba.com>
    Reviewed-by: Chao Yu <chao@kernel.org>
    Link: https://lore.kernel.org/r/20230414061810.6479-1-jefflexu@linux.alibaba.com
    Signed-off-by: Gao Xiang <hsiangkao@linux.alibaba.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit e93f83478c420ab6c828f0b8dd7f3871d91eec49
  Author: Jingbo Xu <jefflexu@linux.alibaba.com>
  Date: Fri Apr 7 22:17:05 2023 +0800

    erofs: initialize packed inode after root inode is assigned

    [Upstream commit cb9bce79514392a9a216ff67148e05e2d72c28bd]

    As commit 8f7acdae2cd4 ("staging: erofs: kill all failure handling in

```
      fill_super()"), move the initialization of packed inode after root
      inode is assigned, so that the iput() in .put_super() is adequate as
      the failure handling.

      Otherwise, iput() is also needed in .kill_sb(), in case of the mounting
      fails halfway.

      Signed-off-by: Jingbo Xu <jefflexu@linux.alibaba.com>
      Reviewed-by: Yue Hu <huyue2@coolpad.com>
      Fixes: b15b2e307c3a ("erofs: support on-disk compressed fragments data")
      Reviewed-by: Gao Xiang <hsiangkao@linux.alibaba.com>
      Acked-by: Chao Yu <chao@kernel.org>
      Link: https://lore.kernel.org/r/20230407141710.113882-3-jefflexu@linux.alibaba.com
      Signed-off-by: Gao Xiang <hsiangkao@linux.alibaba.com>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 96a845419b3722869f09883319de4d55c44d9aef
  Author: Gao Xiang <xiang@kernel.org>
  Date:   Tue Apr 11 01:37:14 2023 +0800

      erofs: stop parsing non-compact HEAD index if clusterofs is invalid

      [ Upstream commit cc4efd3dd2ac9f89143e5d881609747ecff04164 ]

      Syzbot generated a crafted image [1] with a non-compact HEAD index of
      clusterofs 33024 while valid numbers should be 0 ~ lclustersize-1,
      which causes the following unexpected behavior as below:

       BUG: unable to handle page fault for address: fffff52101a3fff9
       #PF: supervisor read access in kernel mode
       #PF: error_code(0x0000) - not-present page
       PGD 23ffed067 P4D 23ffed067 PUD 0
       Oops: 0000 [#1] PREEMPT SMP KASAN
       CPU: 1 PID: 4398 Comm: kworker/u5:1 Not tainted 6.3.0-rc6-syzkaller-g09a9639e56c0 #0
       Hardware name: Google Google Compute Engine/Google Compute Engine, BIOS Google
  03/30/2023
       Workqueue: erofs_worker z_erofs_decompressqueue_work
       RIP: 0010:z_erofs_decompress_queue+0xb7e/0x2b40
       ...
       Call Trace:
        <TASK>
        z_erofs_decompressqueue_work+0x99/0xe0
        process_one_work+0x8f6/0x1170
        worker_thread+0xa63/0x1210
        kthread+0x270/0x300
        ret_from_fork+0x1f/0x30

      Note that normal images or images using compact indexes are not
      impacted.  Let's fix this now.

      [1] https://lore.kernel.org/r/000000000000ec75b005ee97fbaa@google.com

      Reported-and-tested-by: syzbot+aafb3f37cfeb6534c4ac@syzkaller.appspotmail.com
      Fixes: 02827e1796b3 ("staging: erofs: add erofs_map_blocks_iter")
      Fixes: 152a333a5895 ("staging: erofs: add compacted compression indexes support")
      Signed-off-by: Gao Xiang <hsiangkao@linux.alibaba.com>
      Reviewed-by: Chao Yu <chao@kernel.org>
      Link: https://lore.kernel.org/r/20230410173714.104604-1-hsiangkao@linux.alibaba.com
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 9e284d34b1c04d47c58abb77b327236f0e2f786d
  Author: Tobias Klauser <tklauser@distanz.ch>
  Date:   Tue Mar 14 13:34:14 2023 +0100

      selftests/clone3: fix number of tests in ksft_set_plan
```

```
    [ Upstream commit d95debbdc528d50042807754d6085c15abc21768 ]

    Commit 515bddf0ec41 ("selftests/clone3: test clone3 with CLONE_NEWTIME")
    added an additional test, so the number passed to ksft_set_plan needs to
    be bumped accordingly.

    Also use ksft_finished() to print results and exit. This will catch future
    mismatches between ksft_set_plan() and the number of tests being run.

    Fixes: 515bddf0ec41 ("selftests/clone3: test clone3 with CLONE_NEWTIME")
    Cc: Christian Brauner <brauner@kernel.org>
    Signed-off-by: Tobias Klauser <tklauser@distanz.ch>
    Reviewed-by: Christian Brauner <brauner@kernel.org>
    Signed-off-by: Christian Brauner <brauner@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 08217fb43689ab2fde5cbf0b9a5302494fe0cdde
Author: Lino Sanfilippo <l.sanfilippo@kunbus.com>
Date:   Thu Nov 24 14:55:36 2022 +0100

    tpm, tpm_tis: Claim locality when interrupts are reenabled on resume

    [ Upstream commit 955df4f87760b3bb2af253d3fbb12fb712b3ffa6 ]

    In tpm_tis_resume() make sure that the locality has been claimed when
    tpm_tis_reenable_interrupts() is called. Otherwise the writings to the
    register might not have any effect.

    Fixes: 45baa1d1fa39 ("tpm_tis: Re-enable interrupts upon (S3) resume")
    Signed-off-by: Lino Sanfilippo <l.sanfilippo@kunbus.com>
    Tested-by: Jarkko Sakkinen <jarkko@kernel.org>
    Reviewed-by: Jarkko Sakkinen <jarkko@kernel.org>
    Signed-off-by: Jarkko Sakkinen <jarkko@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 972d431fcc50e896c8079c179524559af3144548
Author: Lino Sanfilippo <l.sanfilippo@kunbus.com>
Date:   Thu Nov 24 14:55:33 2022 +0100

    tpm, tpm: Implement usage counter for locality

    [ Upstream commit 7a2f55d0be296c4e81fd782f3d6c43ed4ec7e265 ]

    Implement a usage counter for the (default) locality used by the TPM TIS
    driver:
    Request the locality from the TPM if it has not been claimed yet, otherwise
    only increment the counter. Also release the locality if the counter is 0
    otherwise only decrement the counter. Since in case of SPI the register
    accesses are locked by means of the SPI bus mutex use a sleepable lock
    (i.e. also a mutex) to ensure thread-safety of the counter which may be
    accessed by both a userspace thread and the interrupt handler.

    By doing this refactor the names of the amended functions to use a more
    appropriate prefix.

    Signed-off-by: Lino Sanfilippo <l.sanfilippo@kunbus.com>
    Tested-by: Michael Niewöhner <linux@mniewoehner.de>
    Tested-by: Jarkko Sakkinen <jarkko@kernel.org>
    Reviewed-by: Jarkko Sakkinen <jarkko@kernel.org>
    Signed-off-by: Jarkko Sakkinen <jarkko@kernel.org>
    Stable-dep-of: 955df4f87760 ("tpm, tpm_tis: Claim locality when interrupts are reenabled
on resume")
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```

```
commit bbc90fc0b8b5ee31f948e2e2df7dacaa28f5ce5c
Author: Lino Sanfilippo <l.sanfilippo@kunbus.com>
Date:   Thu Nov 24 14:55:29 2022 +0100

    tpm, tpm_tis: Claim locality before writing interrupt registers

    [ Upstream commit 15d7aa4e46eba87242a320f39773aa16faddadee ]

    In tpm_tis_probe_single_irq() interrupt registers TPM_INT_VECTOR,
    TPM_INT_STATUS and TPM_INT_ENABLE are modified to setup the interrupts.
    Currently these modifications are done without holding a locality thus they
    have no effect. Fix this by claiming the (default) locality before the
    registers are written.

    Since now tpm_tis_gen_interrupt() is called with the locality already
    claimed remove locality request and release from this function.

    Signed-off-by: Lino Sanfilippo <l.sanfilippo@kunbus.com>
    Tested-by: Jarkko Sakkinen <jarkko@kernel.org>
    Reviewed-by: Jarkko Sakkinen <jarkko@kernel.org>
    Signed-off-by: Jarkko Sakkinen <jarkko@kernel.org>
    Stable-dep-of: 955df4f87760 ("tpm, tpm_tis: Claim locality when interrupts are reenabled
on resume")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cc1b06934237ed3fb2b9f75a577fd102f2dba471
Author: Lino Sanfilippo <l.sanfilippo@kunbus.com>
Date:   Thu Nov 24 14:55:27 2022 +0100

    tpm, tpm_tis: Disable interrupts if tpm_tis_probe_irq() failed

    [ Upstream commit 6d789ad726950e612a7f31044260337237c5b490 ]

    Both functions tpm_tis_probe_irq_single() and tpm_tis_probe_irq() may setup
    the interrupts and then return with an error. This case is indicated by a
    missing TPM_CHIP_FLAG_IRQ flag in chip->flags.
    Currently the interrupt setup is only undone if tpm_tis_probe_irq_single()
    fails. Undo the setup also if tpm_tis_probe_irq() fails.

    Signed-off-by: Lino Sanfilippo <l.sanfilippo@kunbus.com>
    Tested-by: Michael Niewöhner <linux@mniewoehner.de>
    Tested-by: Jarkko Sakkinen <jarkko@kernel.org>
    Reviewed-by: Jarkko Sakkinen <jarkko@kernel.org>
    Signed-off-by: Jarkko Sakkinen <jarkko@kernel.org>
    Stable-dep-of: 955df4f87760 ("tpm, tpm_tis: Claim locality when interrupts are reenabled
on resume")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3fa405c23233a4ad1076a65d368fc6bf2392ee97
Author: Lino Sanfilippo <l.sanfilippo@kunbus.com>
Date:   Thu Nov 24 14:55:26 2022 +0100

    tpm, tpm_tis: Claim locality before writing TPM_INT_ENABLE register

    [ Upstream commit 282657a8bd7fddcf511b834f43705001668b33a7 ]

    In disable_interrupts() the TPM_GLOBAL_INT_ENABLE bit is unset in the
    TPM_INT_ENABLE register to shut the interrupts off. However modifying the
    register is only possible with a held locality. So claim the locality
    before disable_interrupts() is called.

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

```
      Signed-off-by: Jarkko Sakkinen <jarkko@kernel.org>
      Stable-dep-of: 955df4f87760 ("tpm, tpm_tis: Claim locality when interrupts are reenabled
  on resume")
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit d7bace634c889a2bbe971c299f0373dee4093e37
  Author: Lino Sanfilippo <l.sanfilippo@kunbus.com>
  Date:   Thu Nov 24 14:55:28 2022 +0100

      tpm, tpm_tis: Do not skip reset of original interrupt vector

      [ Upstream commit ed9be0e6c892a783800d77a41ca4c7255c6af8c5 ]

      If in tpm_tis_probe_irq_single() an error occurs after the original
      interrupt vector has been read, restore the interrupts before the error is
      returned.

      Since the caller does not check the error value, return -1 in any case that
      the TPM_CHIP_FLAG_IRQ flag is not set. Since the return value of function
      tpm_tis_gen_interrupt() is not longer used, make it a void function.

      Fixes: 1107d065fdf1 ("tpm_tis: Introduce intermediate layer for TPM access")
      Signed-off-by: Lino Sanfilippo <l.sanfilippo@kunbus.com>
      Tested-by: Jarkko Sakkinen <jarkko@kernel.org>
      Reviewed-by: Jarkko Sakkinen <jarkko@kernel.org>
      Signed-off-by: Jarkko Sakkinen <jarkko@kernel.org>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 266acd60aa18624317715c2dc351502f8da07c57
  Author: Paul Moore <paul@paul-moore.com>
  Date:   Wed Apr 12 13:29:11 2023 -0400

      selinux: ensure av_permissions.h is built when needed

      [ Upstream commit 4ce1f694eb5d8ca607fed8542d32a33b4f1217a5 ]

      The Makefile rule responsible for building flask.h and
      av_permissions.h only lists flask.h as a target which means that
      av_permissions.h is only generated when flask.h needs to be
      generated.  This patch fixes this by adding av_permissions.h as a
      target to the rule.

      Fixes: 8753f6bec352 ("selinux: generate flask headers during kernel build")
      Signed-off-by: Paul Moore <paul@paul-moore.com>
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 4405e2178344b529be09d95658098ce410421920
  Author: Ondrej Mosnacek <omosnace@redhat.com>
  Date:   Wed Apr 12 15:59:19 2023 +0200

      selinux: fix Makefile dependencies of flask.h

      [ Upstream commit bcab1adeaad4b39a1e04cb98979a367d08253f03 ]

      Make the flask.h target depend on the genheaders binary instead of
      classmap.h to ensure that it is rebuilt if any of the dependencies of
      genheaders are changed.

      Notably this fixes flask.h not being rebuilt when
      initial_sid_to_string.h is modified.

      Fixes: 8753f6bec352 ("selinux: generate flask headers during kernel build")
      Signed-off-by: Ondrej Mosnacek <omosnace@redhat.com>
      Acked-by: Stephen Smalley <stephen.smalley.work@gmail.com>
      Signed-off-by: Paul Moore <paul@paul-moore.com>
```

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

commit 50187d64fd92fc66c04c62bd05d8cd5b85f44062
Author: Ard Biesheuvel <ardb@kernel.org>
Date:   Tue Mar 21 12:01:51 2023 +0100

    ARM: 9293/1: vfp: Pass successful return address via register R3

    [ Upstream commit 3a2bdad0b46649cc73fb3b3f9e2b91ef97a7fa63 ]

    In preparation for reimplementing the do_vfp()->vfp_support_entry()
    handover in C code, switch to using R3 to pass the 'success' return
    address, rather than R9, as it cannot be used for parameter passing.

    Signed-off-by: Ard Biesheuvel <ardb@kernel.org>
    Reviewed-by: Linus Walleij <linus.walleij@linaro.org>
    Tested-by: Guenter Roeck <linux@roeck-us.net>
    Signed-off-by: Russell King (Oracle) <rmk+kernel@armlinux.org.uk>
    Stable-dep-of: c76c6c4ecbec ("ARM: 9294/2: vfp: Fix broken softirq handling with
instrumentation enabled")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cade9ebe11e3de972d16f8aebd85669b29265d83
Author: Ard Biesheuvel <ardb@kernel.org>
Date:   Tue Mar 21 12:01:02 2023 +0100

    ARM: 9292/1: vfp: Pass thread_info pointer to vfp_support_entry

    [ Upstream commit dae904d96ad6a5fa79bd9d99a3decf93685d398b ]

    Instead of dereferencing thread_info in do_vfp, pass the thread_info
    pointer to vfp_support_entry via R1. That way, we only use a single
    caller save register, which makes it easier to convert do_vfp to C code
    in a subsequent patch.

    Note that, unlike the CPU number, which can change due to preemption,
    passing the thread_info pointer can safely be done with preemption
    enabled.

    Signed-off-by: Ard Biesheuvel <ardb@kernel.org>
    Reviewed-by: Linus Walleij <linus.walleij@linaro.org>
    Tested-by: Guenter Roeck <linux@roeck-us.net>
    Signed-off-by: Russell King (Oracle) <rmk+kernel@armlinux.org.uk>
    Stable-dep-of: c76c6c4ecbec ("ARM: 9294/2: vfp: Fix broken softirq handling with
instrumentation enabled")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9d0e4e38c9519387518a92cd301bacf60726b698
Author: Ilpo Järvinen <ilpo.jarvinen@linux.intel.com>
Date:   Wed Feb 15 15:06:00 2023 +0200

    selftests/resctrl: Check for return value after write_schemata()

    [ Upstream commit 0d45c83b95da414e98ad333e723141a94f6e2c64 ]

    MBA test case writes schemata but it does not check if the write is
    successful or not.

    Add the error check and return error properly.

    Fixes: 01fee6b4d1f9 ("selftests/resctrl: Add MBA test")
    Co-developed-by: Fenghua Yu <fenghua.yu@intel.com>
    Signed-off-by: Fenghua Yu <fenghua.yu@intel.com>
    Signed-off-by: Ilpo Järvinen <ilpo.jarvinen@linux.intel.com>
    Reviewed-by: Reinette Chatre <reinette.chatre@intel.com>
```

```
    Signed-off-by: Shuah Khan <skhan@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9e01c8a1bb44b27082f3bf864129ea11a77faa1c
Author: Ilpo Järvinen <ilpo.jarvinen@linux.intel.com>
Date:   Wed Feb 15 15:05:59 2023 +0200

    selftests/resctrl: Allow ->setup() to return errors

    [ Upstream commit fa10366cc6f4cc862871f8938426d85c2481f084 ]

    resctrl_val() assumes ->setup() always returns either 0 to continue
    tests or < 0 in case of the normal termination of tests after x runs.
    The latter overlaps with normal error returns.

    Define END_OF_TESTS (=1) to differentiate the normal termination of
    tests and return errors as negative values. Alter callers of ->setup()
    to handle errors properly.

    Fixes: 790bf585b0ee ("selftests/resctrl: Add Cache Allocation Technology (CAT)
selftest")
    Fixes: ecdbb911f22d ("selftests/resctrl: Add MBM test")
    Signed-off-by: Ilpo Järvinen <ilpo.jarvinen@linux.intel.com>
    Reviewed-by: Reinette Chatre <reinette.chatre@intel.com>
    Signed-off-by: Shuah Khan <skhan@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 553f56b766605456130b0899a21d15de99e321e6
Author: Ilpo Järvinen <ilpo.jarvinen@linux.intel.com>
Date:   Wed Feb 15 15:05:58 2023 +0200

    selftests/resctrl: Move ->setup() call outside of test specific branches

    [ Upstream commit c90b3b588e369c20087699316259fa5ebbb16f2d ]

    resctrl_val() function is called only by MBM, MBA, and CMT tests which
    means the else branch is never used.

    Both test branches call param->setup().

    Remove the unused else branch and place the ->setup() call outside of
    the test specific branches reducing code duplication.

    Co-developed-by: Fenghua Yu <fenghua.yu@intel.com>
    Signed-off-by: Fenghua Yu <fenghua.yu@intel.com>
    Signed-off-by: Ilpo Järvinen <ilpo.jarvinen@linux.intel.com>
    Reviewed-by: Reinette Chatre <reinette.chatre@intel.com>
    Signed-off-by: Shuah Khan <skhan@linuxfoundation.org>
    Stable-dep-of: fa10366cc6f4 ("selftests/resctrl: Allow ->setup() to return errors")
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 24c31a8e22f8a2d4732dd3d30268fdf03768fdb8
Author: Ilpo Järvinen <ilpo.jarvinen@linux.intel.com>
Date:   Wed Feb 15 15:05:57 2023 +0200

    selftests/resctrl: Return NULL if malloc_and_init_memory() did not alloc mem

    [ Upstream commit 22a8be280383812235131dda18a8212a59fadd2d ]

    malloc_and_init_memory() in fill_buf isn't checking if memalign()
    successfully allocated memory or not before accessing the memory.

    Check the return value of memalign() and return NULL if allocating
    aligned memory fails.
```

```
    Fixes: a2561b12fe39 ("selftests/resctrl: Add built in benchmark")
    Co-developed-by: Fenghua Yu <fenghua.yu@intel.com>
    Signed-off-by: Fenghua Yu <fenghua.yu@intel.com>
    Signed-off-by: Ilpo Järvinen <ilpo.jarvinen@linux.intel.com>
    Reviewed-by: Reinette Chatre <reinette.chatre@intel.com>
    Signed-off-by: Shuah Khan <skhan@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b6735005776ea195b07bcaad6eac26facfd1f3c6
Author: Zqiang <qiang1.zhang@intel.com>
Date:   Tue Dec 20 22:16:25 2022 +0800

    rcu: Fix missing TICK_DEP_MASK_RCU_EXP dependency check

    [ Upstream commit db7b464df9d820186e98a65aa6a10f0d51fbf8ce ]

    This commit adds checks for the TICK_DEP_MASK_RCU_EXP bit, thus enabling
    RCU expedited grace periods to actually force-enable scheduling-clock
    interrupts on holdout CPUs.

    Fixes: df1e849ae455 ("rcu: Enable tick for nohz_full CPUs slow to provide expedited QS")
    Signed-off-by: Zqiang <qiang1.zhang@intel.com>
    Cc: Steven Rostedt <rostedt@goodmis.org>
    Cc: Masami Hiramatsu <mhiramat@kernel.org>
    Cc: Frederic Weisbecker <fweisbec@gmail.com>
    Cc: Thomas Gleixner <tglx@linutronix.de>
    Cc: Ingo Molnar <mingo@kernel.org>
    Cc: Anna-Maria Behnsen <anna-maria@linutronix.de>
    Acked-by: Frederic Weisbecker <frederic@kernel.org>
    Signed-off-by: Paul E. McKenney <paulmck@kernel.org>
    Signed-off-by: Joel Fernandes (Google) <joel@joelfernandes.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6dd75bc083612961c7b382d96d4e9d28cc2ebd8c
Author: Rae Moar <rmoar@google.com>
Date:   Wed Mar 8 20:39:51 2023 +0000

    kunit: fix bug in the order of lines in debugfs logs

    [ Upstream commit f9a301c3317daa921375da0aec82462ddf019928 ]

    Fix bug in debugfs logs that causes an incorrect order of lines in the
    debugfs log.

    Currently, the test counts lines that show the number of tests passed,
    failed, and skipped, as well as any suite diagnostic lines,
    appear prior to the individual results, which is a bug.

    Ensure the order of printing for the debugfs log is correct. Additionally,
    add a KTAP header to so the debugfs logs can be valid KTAP.

    This is an example of a log prior to these fixes:

         KTAP version 1

         # Subtest: kunit_status
         1..2
     # kunit_status: pass:2 fail:0 skip:0 total:2
     # Totals: pass:2 fail:0 skip:0 total:2
         ok 1 kunit_status_set_failure_test
         ok 2 kunit_status_mark_skipped_test
     ok 1 kunit_status

    Note the two lines with stats are out of order. This is the same debugfs
    log after the fixes (in combination with the third patch to remove the
```

```
     extra line):

      KTAP version 1
      1..1
          KTAP version 1
          # Subtest: kunit_status
          1..2
          ok 1 kunit_status_set_failure_test
          ok 2 kunit_status_mark_skipped_test
      # kunit_status: pass:2 fail:0 skip:0 total:2
      # Totals: pass:2 fail:0 skip:0 total:2
      ok 1 kunit_status

    Signed-off-by: Rae Moar <rmoar@google.com>
    Reviewed-by: David Gow <davidgow@google.com>
    Signed-off-by: Shuah Khan <skhan@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4fcadf57c69119e75595a62748633cb15177cbf8
Author: Quentin Schulz <quentin.schulz@theobroma-systems.com>
Date:   Thu Nov 17 13:04:31 2022 +0100

    clk: rockchip: rk3399: allow clk_cifout to force clk_cifout_src to reparent

    commit 933bf364e152cd60902cf9585c2ba310d593e69f upstream.

    clk_cifout is derived from clk_cifout_src through an integer divider
    limited to 32. clk_cifout_src is a child of either cpll, gpll or npll
    without any possibility of a divider of any sort. The default clock
    parent is cpll.

    Let's allow clk_cifout to ask its parent clk_cifout_src to reparent in
    order to find the real closest possible rate for clk_cifout and not one
    derived from cpll only.

    Cc: stable@vger.kernel.org # 4.10+
    Fixes: fd8bc829336a ("clk: rockchip: fix the rk3399 cifout clock")
    Signed-off-by: Quentin Schulz <quentin.schulz@theobroma-systems.com>
    Link: https://lore.kernel.org/r/20221117-rk3399-cifout-set-rate-parent-v1-0-
432548d04081@theobroma-systems.com
    Signed-off-by: Heiko Stuebner <heiko@sntech.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 934406b2d42eaf3fc57f5546cc68ff7ab9680bb3
Author: Conor Dooley <conor.dooley@microchip.com>
Date:   Thu Apr 13 23:20:45 2023 +0100

    clk: microchip: fix potential UAF in auxdev release callback

    commit 7455b7007b9e93bcc2bc9c1c6c73a228e3152069 upstream.

    Similar to commit 1c11289b34ab ("peci: cpu: Fix use-after-free in
    adev_release()"), the auxiliary device is not torn down in the correct
    order. If auxiliary_device_add() fails, the release callback will be
    called twice, resulting in a UAF. Due to timing, the auxdev code in this
    driver "took inspiration" from the aforementioned commit, and thus its
    bugs too!

    Moving auxiliary_device_uninit() to the unregister callback instead
    avoids the issue.

    CC: stable@vger.kernel.org
    Fixes: b56bae2dd6fd ("clk: microchip: mpfs: add reset controller")
    Signed-off-by: Conor Dooley <conor.dooley@microchip.com>
    Link: https://lore.kernel.org/r/20230413-critter-synopsis-dac070a86cb4@spud
```

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

commit b1b90c7df08ef385f95827ee3aee87bddd1ef5c5
Author: Ping-Ke Shih <pkshih@realtek.com>
Date:   Thu Mar 23 16:28:39 2023 +0800

    wifi: rtw89: fix potential race condition between napi_init and napi_enable

    commit 47515664ecfbde11425dff121f298ae4499425c9 upstream.

    A race condition can happen if netdev is registered, but NAPI isn't
    initialized yet, and meanwhile user space starts the netdev that will
    enable NAPI. Then, it hits BUG_ON():

     kernel BUG at net/core/dev.c:6423!
     invalid opcode: 0000 [#1] PREEMPT SMP NOPTI
     CPU: 0 PID: 417 Comm: iwd Not tainted 6.2.7-slab-dirty #3 eb0f5a8a9d91
     Hardware name: LENOVO 21DL/LNVNB161216, BIOS JPCN20WW(V1.06) 09/20/2022
     RIP: 0010:napi_enable+0x3f/0x50
     Code: 48 89 c2 48 83 e2 f6 f6 81 89 08 00 00 02 74 0d 48 83 ...
     RSP: 0018:ffffada1414f3548 EFLAGS: 00010246
     RAX: 0000000000000000 RBX: ffffa01425802080 RCX: 0000000000000000
     RDX: 00000000000002ff RSI: ffffada14e50c614 RDI: ffffa01425808dc0
     RBP: 0000000000000000 R08: 0000000000000000 R09: 0000000000000000
     R10: 0000000000000001 R11: 0000000000000100 R12: ffffa01425808f58
     R13: 0000000000000000 R14: ffffa01423498940 R15: 0000000000000001
     FS:  00007f5577c0a740(0000) GS:ffffa0169fc00000(0000) knlGS:0000000000000000
     CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
     CR2: 00007f5577a19972 CR3: 0000000125a7a000 CR4: 0000000000750ef0
     PKRU: 55555554
     Call Trace:
      <TASK>
      rtw89_pci_ops_start+0x1c/0x70 [rtw89_pci 6cbc75429515c181cbc386478d5cfb32ffc5a0f8]
      rtw89_core_start+0xbe/0x160 [rtw89_core fe07ecb874820b6d778370d4acb6ef8a37847f22]
      rtw89_ops_start+0x26/0x40 [rtw89_core fe07ecb874820b6d778370d4acb6ef8a37847f22]
      drv_start+0x42/0x100 [mac80211 c07fa22af8c3cf3f7d7ab3884ca990784d72e2d2]
      ieee80211_do_open+0x311/0x7d0 [mac80211 c07fa22af8c3cf3f7d7ab3884ca990784d72e2d2]
      ieee80211_open+0x6a/0x90 [mac80211 c07fa22af8c3cf3f7d7ab3884ca990784d72e2d2]
      __dev_open+0xe0/0x180
      __dev_change_flags+0x1da/0x250
      dev_change_flags+0x26/0x70
      do_setlink+0x37c/0x12c0
      ? ep_poll_callback+0x246/0x290
      ? __nla_validate_parse+0x61/0xd00
      ? __wake_up_common_lock+0x8f/0xd0

    To fix this, follow Jonas' suggestion to switch the order of these
    functions and move register netdev to be the last step of PCI probe.
    Also, correct the error handling of rtw89_core_register_hw().

    Fixes: e3ec7017f6a2 ("rtw89: add Realtek 802.11ax driver")
    Cc: stable@vger.kernel.org
    Reported-by: Hyeonggon Yoo <42.hyeyoo@gmail.com>
    Link: https://lore.kernel.org/linux-
wireless/CAOiHx=n7EwK2B9CnBR07FVA=sEzFagb8TkS4XC_qBNq8OwcYUg@mail.gmail.com/T/#t
    Suggested-by: Jonas Gorski <jonas.gorski@gmail.com>
    Tested-by: Larry Finger<Larry.Finger@lwfinger.net>
    Reviewed-by: Larry Finger<Larry.Finger@lwfinger.net>
    Signed-off-by: Ping-Ke Shih <pkshih@realtek.com>
    Signed-off-by: Kalle Valo <kvalo@kernel.org>
    Link: https://lore.kernel.org/r/20230323082839.20474-1-pkshih@realtek.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 57b45b153170484f88019e7ad417fb66dba55388
```

```
Author: Sascha Hauer <s.hauer@pengutronix.de>
Date:   Mon Apr 17 16:03:56 2023 +0200

    wifi: rtw88: rtw8821c: Fix rfe_option field width

    commit 14705f969d98187a1cc2682e0c9bd2e230b8098f upstream.

    On my RTW8821CU chipset rfe_option reads as 0x22. Looking at the
    vendor driver suggests that the field width of rfe_option is 5 bit,
    so rfe_option should be masked with 0x1f.

    Without this the rfe_option comparisons with 2 further down the
    driver evaluate as false when they should really evaluate as true.
    The effect is that 2G channels do not work.

    rfe_option is also used as an array index into rtw8821c_rfe_defs[].
    rtw8821c_rfe_defs[34] (0x22) was added as part of adding USB support,
    likely because rfe_option reads as 0x22. As this now becomes 0x2,
    rtw8821c_rfe_defs[34] is no longer used and can be removed.

    Note that this might not be the whole truth. In the vendor driver
    there are indeed places where the unmasked rfe_option value is used.
    However, the driver has several places where rfe_option is tested
    with the pattern if (rfe_option == 2 || rfe_option == 0x22) or
    if (rfe_option == 4 || rfe_option == 0x24), so that rfe_option BIT(5)
    has no influence on the code path taken. We therefore mask BIT(5)
    out from rfe_option entirely until this assumption is proved wrong
    by some chip variant we do not know yet.

    Signed-off-by: Sascha Hauer <s.hauer@pengutronix.de>
    Tested-by: Alexandru gagniuc <mr.nuke.me@gmail.com>
    Tested-by: Larry Finger <Larry.Finger@lwfinger.net>
    Tested-by: ValdikSS <iam@valdikss.org.ru>
    Cc: stable@vger.kernel.org
    Reviewed-by: Ping-Ke Shih <pkshih@realtek.com>
    Signed-off-by: Kalle Valo <kvalo@kernel.org>
    Link: https://lore.kernel.org/r/20230417140358.2240429-3-s.hauer@pengutronix.de
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1804c140467b4c6362a3b4d55e6b8058478a7b1f
Author: Bitterblue Smith <rtl8821cerfe2@gmail.com>
Date:   Mon Mar 13 15:42:59 2023 +0200

    wifi: rtl8xxxu: RTL8192EU always needs full init

    commit d46e04ccd40457a0119b76e11ab64a2ad403e138 upstream.

    Always run the entire init sequence (rtl8xxxu_init_device()) for
    RTL8192EU. It's what the vendor driver does too.

    This fixes a bug where the device is unable to connect after
    rebooting:

    wlp3s0f3u2: send auth to ... (try 1/3)
    wlp3s0f3u2: send auth to ... (try 2/3)
    wlp3s0f3u2: send auth to ... (try 3/3)
    wlp3s0f3u2: authentication with ... timed out

    Rebooting leaves the device powered on (partially? at least the
    firmware is still running), but not really in a working state.

    Cc: stable@vger.kernel.org
    Signed-off-by: Bitterblue Smith <rtl8821cerfe2@gmail.com>
    Acked-by: Jes Sorensen <jes@trained-monkey.org>
    Signed-off-by: Kalle Valo <kvalo@kernel.org>
```

```
    Link: https://lore.kernel.org/r/4eb111a9-d4c4-37d0-b376-4e202de7153c@gmail.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 67ce57b4c48b2bcb6b43d8d45d3c5ada02fc218b
Author: Huanhuan Wang <huanhuan.wang@corigine.com>
Date:   Thu Apr 20 16:01:25 2023 +0200

    nfp: fix incorrect pointer deference when offloading IPsec with bonding

    commit 63cfd210034c772fad047afa13dd5a4664b0a72e upstream.

    There are two pointers in struct xfrm_dev_offload, *dev, *real_dev.
    The *dev points whether bonding interface or real interface, if
    bonding IPsec offload is used, it points bonding interface; if not,
    it points real interface. And *real_dev always points real interface.
    So nfp should always use real_dev instead of dev.

    Prior to this change the system becomes unresponsive when offloading
    IPsec for a device which is a lower device to a bonding device.

    Fixes: 859a497fe80c ("nfp: implement xfrm callbacks and expose ipsec offload feature to
upper layer")
    CC: stable@vger.kernel.org
    Signed-off-by: Huanhuan Wang <huanhuan.wang@corigine.com>
    Acked-by: Simon Horman <simon.horman@corigine.com>
    Signed-off-by: Louis Peens <louis.peens@corigine.com>
    Link: https://lore.kernel.org/r/20230420140125.38521-1-louis.peens@corigine.com
    Signed-off-by: Jakub Kicinski <kuba@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e1c7177fafc2886ba494467d0f946148d6f873e7
Author: Tanmay Shah <tanmay.shah@amd.com>
Date:   Fri Mar 10 17:24:06 2023 -0800

    mailbox: zynqmp: Fix typo in IPI documentation

    commit 79963fbfc233759bd8a43462f120d15a1bd4f4fa upstream.

    Xilinx IPI message buffers allows 32-byte data transfer.
    Fix documentation that says 12 bytes

    Fixes: 4981b82ba2ff ("mailbox: ZynqMP IPI mailbox controller")
    Signed-off-by: Tanmay Shah <tanmay.shah@amd.com>
    Acked-by: Michal Simek <michal.simek@amd.com>
    Cc: stable@vger.kernel.org
    Link: https://lore.kernel.org/r/20230311012407.1292118-4-tanmay.shah@amd.com
    Signed-off-by: Mathieu Poirier <mathieu.poirier@linaro.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 9ef69f3c49d0222f8e7c8808e5d39a0f354fc2b0
Author: Tanmay Shah <tanmay.shah@amd.com>
Date:   Fri Mar 10 17:24:04 2023 -0800

    mailbox: zynqmp: Fix counts of child nodes

    commit f72f805e72882c361e2a612c64a6e549f3da7152 upstream.

    If child mailbox node status is disabled it causes
    crash in interrupt handler. Fix this by assigning
    only available child node during driver probe.

    Fixes: 4981b82ba2ff ("mailbox: ZynqMP IPI mailbox controller")
    Signed-off-by: Tanmay Shah <tanmay.shah@amd.com>
    Acked-by: Michal Simek <michal.simek@amd.com>
    Cc: stable@vger.kernel.org
```

```
    Link: https://lore.kernel.org/r/20230311012407.1292118-2-tanmay.shah@amd.com
    Signed-off-by: Mathieu Poirier <mathieu.poirier@linaro.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f8f2297355513e5e0631e604ef9d7e449c7dcd00
Author: Marco Elver <elver@google.com>
Date:   Thu Mar 9 11:17:52 2023 +0100

    kcsan: Avoid READ_ONCE() in read_instrumented_memory()

    commit 8dec88070d964bfeb4198f34cb5956d89dd1f557 upstream.

    Haibo Li reported:

     | Unable to handle kernel paging request at virtual address
     |   ffffff802a0d8d7171
     | Mem abort info:o:
     |   ESR = 0x9600002121
     |   EC = 0x25: DABT (current EL), IL = 32 bitsts
     |   SET = 0, FnV = 0 0
     |   EA = 0, S1PTW = 0 0
     |   FSC = 0x21: alignment fault
     | Data abort info:o:
     |   ISV = 0, ISS = 0x0000002121
     |   CM = 0, WnR = 0 0
     | swapper pgtable: 4k pages, 39-bit VAs, pgdp=000000002835200000
     | [ffffff802a0d8d71] pgd=180000005fbf9003, p4d=180000005fbf9003,
     | pud=180000005fbf9003, pmd=180000005fbe8003, pte=006800002a0d8707
     | Internal error: Oops: 96000021 [#1] PREEMPT SMP
     | Modules linked in:
     | CPU: 2 PID: 45 Comm: kworker/u8:2 Not tainted
     |   5.15.78-android13-8-g63561175bbda-dirty #1
     | ...
     | pc : kcsan_setup_watchpoint+0x26c/0x6bc
     | lr : kcsan_setup_watchpoint+0x88/0x6bc
     | sp : ffffffc00ab4b7f0
     | x29: ffffffc00ab4b800 x28: ffffff80294fe588 x27: 0000000000000001
     | x26: 0000000000000019 x25: 0000000000000001 x24: ffffff80294fdb80
     | x23: 0000000000000000 x22: ffffffc00a70fb68 x21: ffffff802a0d8d71
     | x20: 0000000000000002 x19: 0000000000000000 x18: ffffffc00a9bd060
     | x17: 0000000000000001 x16: 0000000000000000 x15: ffffffc00a59f000
     | x14: 0000000000000001 x13: 0000000000000000 x12: ffffffc00a70faa0
     | x11: 00000000aaaaaaab x10: 0000000000000054 x9 : ffffffc00839adf8
     | x8 : ffffffc009b4cf00 x7 : 0000000000000000 x6 : 0000000000000007
     | x5 : 0000000000000000 x4 : 0000000000000000 x3 : ffffffc00a70fb70
     | x2 : 0005ff802a0d8d71 x1 : 0000000000000000 x0 : 0000000000000000
     | Call trace:
     |  kcsan_setup_watchpoint+0x26c/0x6bc
     |  __tsan_read2+0x1f0/0x234
     |  inflate_fast+0x498/0x750
     |  zlib_inflate+0x1304/0x2384
     |  __gunzip+0x3a0/0x45c
     |  gunzip+0x20/0x30
     |  unpack_to_rootfs+0x2a8/0x3fc
     |  do_populate_rootfs+0xe8/0x11c
     |  async_run_entry_fn+0x58/0x1bc
     |  process_one_work+0x3ec/0x738
     |  worker_thread+0x4c4/0x838
     |  kthread+0x20c/0x258
     |  ret_from_fork+0x10/0x20
     | Code: b8bfc2a8 2a0803f7 14000007 d503249f (78bfc2a8) )
     | ---[ end trace 613a943cb0a572b6 ]-----

    The reason for this is that on certain arm64 configuration since
    e35123d83ee3 ("arm64: lto: Strengthen READ_ONCE() to acquire when
```

```
    CONFIG_LTO=y"), READ_ONCE() may be promoted to a full atomic acquire
    instruction which cannot be used on unaligned addresses.

    Fix it by avoiding READ_ONCE() in read_instrumented_memory(), and simply
    forcing the compiler to do the required access by casting to the
    appropriate volatile type. In terms of generated code this currently
    only affects architectures that do not use the default READ_ONCE()
    implementation.

    The only downside is that we are not guaranteed atomicity of the access
    itself, although on most architectures a plain load up to machine word
    size should still be atomic (a fact the default READ_ONCE() still relies
    on itself).

    Reported-by: Haibo Li <haibo.li@mediatek.com>
    Tested-by: Haibo Li <haibo.li@mediatek.com>
    Cc: <stable@vger.kernel.org> # 5.17+
    Signed-off-by: Marco Elver <elver@google.com>
    Signed-off-by: Paul E. McKenney <paulmck@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a3e30441f95876ff033af2b06dc815f9abb02238
Author: Tanmay Shah <tanmay.shah@amd.com>
Date:   Fri Mar 10 17:24:05 2023 -0800

    mailbox: zynqmp: Fix IPI isr handling

    commit 74ad37a30ffee3643bc34f9ca7225b20a66abaaf upstream.

    Multiple IPI channels are mapped to same interrupt handler.
    Current isr implementation handles only one channel per isr.
    Fix this behavior by checking isr status bit of all child
    mailbox nodes.

    Fixes: 4981b82ba2ff ("mailbox: ZynqMP IPI mailbox controller")
    Signed-off-by: Tanmay Shah <tanmay.shah@amd.com>
    Acked-by: Michal Simek <michal.simek@amd.com>
    Cc: stable@vger.kernel.org
    Link: https://lore.kernel.org/r/20230311012407.1292118-3-tanmay.shah@amd.com
    Signed-off-by: Mathieu Poirier <mathieu.poirier@linaro.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 15f9cffc9278da60ce587ff0829c19e851dcd5b5
Author: Tanmay Shah <tanmay.shah@amd.com>
Date:   Fri Mar 10 17:24:07 2023 -0800

    drivers: remoteproc: xilinx: Fix carveout names

    commit 81c18e08a609706c5c2887f267135fa0dece4119 upstream.

    If the unit address is appended to node name of memory-region,
    then adding rproc carveouts fails as node name and unit-address
    both are passed as carveout name (i.e. vdev0vring0@xxxxxxxx). However,
    only node name is expected by remoteproc framework. This patch moves
    memory-region node parsing from driver probe to prepare and
    only passes node-name and not unit-address

    Fixes: 6b291e8020a8 ("drivers: remoteproc: Add Xilinx r5 remoteproc driver")
    Signed-off-by: Tanmay Shah <tanmay.shah@amd.com>
    Cc: stable@vger.kernel.org
    Link: https://lore.kernel.org/r/20230311012407.1292118-5-tanmay.shah@amd.com
    Signed-off-by: Mathieu Poirier <mathieu.poirier@linaro.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit defee9bfb61c9cc908e974d9cf85b4a032a71018
```

```
Author: Tudor Ambarus <tudor.ambarus@linaro.org>
Date:   Fri Mar 31 07:46:04 2023 +0000

    mtd: spi-nor: core: Update flash's current address mode when changing address mode

    commit 37513c56139b79dd43c1774513c28f8ab2b05224 upstream.

    The bug was obswerved while reading code. There are not many users of
    addr_mode_nbytes. Anyway, we should update the flash's current address
    mode when changing the address mode, fix it. We don't care for now about
    the set_4byte_addr_mode(nor, false) from spi_nor_restore(), as it is
    used at driver remove and shutdown.

    Fixes: d7931a215063 ("mtd: spi-nor: core: Track flash's internal address mode")
    Cc: stable@vger.kernel.org
    Link: https://lore.kernel.org/r/20230331074606.3559258-9-tudor.ambarus@linaro.org
    Signed-off-by: Tudor Ambarus <tudor.ambarus@linaro.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit dd6c85f98ff33b009b819090268cbf679dc208b9
Author: Michael Walle <michael@walle.cc>
Date:   Wed Mar 8 09:20:20 2023 +0100

    mtd: core: fix error path for nvmem provider

    commit e0489f6e221f5ddee6cb3bd51b992b790c5fa4b9 upstream.

    If mtd_otp_nvmem_add() fails, the partitions won't be removed
    because there is simply no call to del_mtd_partitions().
    Unfortunately, add_mtd_partitions() will print all partitions to
    the kernel console. If mtd_otp_nvmem_add() returns -EPROBE_DEFER
    this would print the partitions multiple times to the kernel
    console. Instead move mtd_otp_nvmem_add() to the beginning of the
    function.

    Fixes: 4b361cfa8624 ("mtd: core: add OTP nvmem provider support")
    Cc: stable@vger.kernel.org
    Signed-off-by: Michael Walle <michael@walle.cc>
    Signed-off-by: Miquel Raynal <miquel.raynal@bootlin.com>
    Link: https://lore.kernel.org/linux-mtd/20230308082021.870459-3-michael@walle.cc
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit fe0c169e2dc71f6dd41b7a05273c4dbf03b53d7b
Author: Michael Walle <michael@walle.cc>
Date:   Wed Mar 8 09:20:19 2023 +0100

    mtd: core: fix nvmem error reporting

    commit 8bd1d24e6ca3c599dd455b0e1b22f77bab8290eb upstream.

    The master MTD will only have an associated device if
    CONFIG_MTD_PARTITIONED_MASTER is set, thus we cannot use dev_err() on
    mtd->dev. Instead use the parent device which is the physical flash
    memory.

    Fixes: 4b361cfa8624 ("mtd: core: add OTP nvmem provider support")
    Cc: stable@vger.kernel.org
    Signed-off-by: Michael Walle <michael@walle.cc>
    Signed-off-by: Miquel Raynal <miquel.raynal@bootlin.com>
    Link: https://lore.kernel.org/linux-mtd/20230308082021.870459-2-michael@walle.cc
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit cbedf71f0003fb951f0b53b8fb6018991d7ae29f
Author: Michael Walle <michael@walle.cc>
Date:   Wed Mar 8 09:20:18 2023 +0100
```

```
    mtd: core: provide unique name for nvmem device, take two

    commit 1cd9ceaa5282ff10ea20a7fbadde5a476a1cc99e upstream.

    Commit c048b60d39e1 ("mtd: core: provide unique name for nvmem device")
    tries to give the nvmem device a unique name, but fails badly if the mtd
    device doesn't have a "struct device" associated with it, i.e. if
    CONFIG_MTD_PARTITIONED_MASTER is not set. This will result in the name
    "(null)-user-otp", which is not unique. It seems the best we can do is
    to use the compatible name together with a unique identifier added by
    the nvmem subsystem by using NVMEM_DEVID_AUTO.

    Fixes: c048b60d39e1 ("mtd: core: provide unique name for nvmem device")
    Cc: stable@vger.kernel.org
    Signed-off-by: Michael Walle <michael@walle.cc>
    Signed-off-by: Miquel Raynal <miquel.raynal@bootlin.com>
    Link: https://lore.kernel.org/linux-mtd/20230308082021.870459-1-michael@walle.cc
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 8cab2c972012bd8794b657adf976da6721662cfd
Author: Mark Rutland <mark.rutland@arm.com>
Date:   Tue Apr 18 17:42:12 2023 +0100

    kasan: hw_tags: avoid invalid virt_to_page()

    commit 29083fd84da576bfb3563d044f98d38e6b338f00 upstream.

    When booting with 'kasan.vmalloc=off', a kernel configured with support
    for KASAN_HW_TAGS will explode at boot time due to bogus use of
    virt_to_page() on a vmalloc adddress.  With CONFIG_DEBUG_VIRTUAL selected
    this will be reported explicitly, and with or without CONFIG_DEBUG_VIRTUAL
    the kernel will dereference a bogus address:

    | ------------[ cut here ]------------
    | virt_to_phys used for non-linear address: (____ptrval____) (0xffff800008000000)
    | WARNING: CPU: 0 PID: 0 at arch/arm64/mm/physaddr.c:15 __virt_to_phys+0x78/0x80
    | Modules linked in:
    | CPU: 0 PID: 0 Comm: swapper/0 Not tainted 6.3.0-rc3-00073-g83865133300d-dirty #4
    | Hardware name: linux,dummy-virt (DT)
    | pstate: 600000c5 (nZCv daIF -PAN -UAO -TCO -DIT -SSBS BTYPE=--)
    | pc : __virt_to_phys+0x78/0x80
    | lr : __virt_to_phys+0x78/0x80
    | sp : ffffcd076afd3c80
    | x29: ffffcd076afd3c80 x28: 0068000000000f07 x27: ffff800008000000
    | x26: fffffbfff0000000 x25: fffffbffff000000 x24: ff00000000000000
    | x23: ffffcd076ad3c000 x22: fffffc0000000000 x21: ffff800008000000
    | x20: ffff800008004000 x19: ffff800008000000 x18: ffff800008004000
    | x17: 666678302820295f x16: ffffffffffffffff x15: 0000000000000004
    | x14: ffffcd076b009e88 x13: 0000000000000fff x12: 0000000000000003
    | x11: 00000000ffffefff x10: c0000000ffffefff x9 : 0000000000000000
    | x8 : 0000000000000000 x7 : 205d303030303030 x6 : 302e30202020205b
    | x5 : ffffcd076b41d63f x4 : ffffcd076afd3827 x3 : 0000000000000000
    | x2 : 0000000000000000 x1 : ffffcd076afd3a30 x0 : 000000000000004f
    | Call trace:
    |  __virt_to_phys+0x78/0x80
    |  __kasan_unpoison_vmalloc+0xd4/0x478
    |  __vmalloc_node_range+0x77c/0x7b8
    |  __vmalloc_node+0x54/0x64
    |  init_IRQ+0x94/0xc8
    |  start_kernel+0x194/0x420
    |  __primary_switched+0xbc/0xc4
    | ---[ end trace 0000000000000000 ]---
    | Unable to handle kernel paging request at virtual address 03fffacbe27b8000
    | Mem abort info:
```

```
     |   ESR = 0x0000000096000004
     |   EC = 0x25: DABT (current EL), IL = 32 bits
     |   SET = 0, FnV = 0
     |   EA = 0, S1PTW = 0
     |   FSC = 0x04: level 0 translation fault
     | Data abort info:
     |   ISV = 0, ISS = 0x00000004
     |   CM = 0, WnR = 0
     | swapper pgtable: 4k pages, 48-bit VAs, pgdp=0000000041bc5000
     | [03fffacbe27b8000] pgd=0000000000000000, p4d=0000000000000000
     | Internal error: Oops: 0000000096000004 [#1] PREEMPT SMP
     | Modules linked in:
     | CPU: 0 PID: 0 Comm: swapper/0 Tainted: G        W         6.3.0-rc3-00073-
 g83865133300d-dirty #4
     | Hardware name: linux,dummy-virt (DT)
     | pstate: 200000c5 (nzCv daIF -PAN -UAO -TCO -DIT -SSBS BTYPE=--)
     | pc : __kasan_unpoison_vmalloc+0xe4/0x478
     | lr : __kasan_unpoison_vmalloc+0xd4/0x478
     | sp : ffffcd076afd3ca0
     | x29: ffffcd076afd3ca0 x28: 0068000000000f07 x27: ffff800008000000
     | x26: 0000000000000000 x25: 03fffacbe27b8000 x24: ff00000000000000
     | x23: ffffcd076ad3c000 x22: fffffc0000000000 x21: ffff800008000000
     | x20: ffff800008004000 x19: ffff800008000000 x18: ffff800008004000
     | x17: 666678302820295f x16: ffffffffffffffff x15: 0000000000000004
     | x14: ffffcd076b009e88 x13: 0000000000000fff x12: 0000000000000001
     | x11: 0000800008000000 x10: ffff800008000000 x9 : ffffb2f8dee00000
     | x8 : 000ffffb2f8dee00 x7 : 205d303030303030 x6 : 302e30202020205b
     | x5 : ffffcd076b41d63f x4 : ffffcd076afd3827 x3 : 0000000000000000
     | x2 : 0000000000000000 x1 : ffffcd076afd3a30 x0 : ffffb2f8dee00000
     | Call trace:
     |  __kasan_unpoison_vmalloc+0xe4/0x478
     |  __vmalloc_node_range+0x77c/0x7b8
     |  __vmalloc_node+0x54/0x64
     |  init_IRQ+0x94/0xc8
     |  start_kernel+0x194/0x420
     |  __primary_switched+0xbc/0xc4
     | Code: d34cfc08 aa1f03fa 8b081b39 d503201f (f9400328)
     | ---[ end trace 0000000000000000 ]---
     | Kernel panic - not syncing: Attempted to kill the idle task!

    This is because init_vmalloc_pages() erroneously calls virt_to_page() on
    a vmalloc address, while virt_to_page() is only valid for addresses in
    the linear/direct map. Since init_vmalloc_pages() expects virtual
    addresses in the vmalloc range, it must use vmalloc_to_page() rather
    than virt_to_page().

    We call init_vmalloc_pages() from __kasan_unpoison_vmalloc(), where we
    check !is_vmalloc_or_module_addr(), suggesting that we might encounter a
    non-vmalloc address. Luckily, this never happens. By design, we only
    call __kasan_unpoison_vmalloc() on pointers in the vmalloc area, and I
    have verified that we don't violate that expectation. Given that,
    is_vmalloc_or_module_addr() must always be true for any legitimate
    argument to __kasan_unpoison_vmalloc().

    Correct init_vmalloc_pages() to use vmalloc_to_page(), and remove the
    redundant and misleading use of is_vmalloc_or_module_addr() in
    __kasan_unpoison_vmalloc().

    Link: https://lkml.kernel.org/r/20230418164212.1775741-1-mark.rutland@arm.com
    Fixes: 6c2f761dad7851d8 ("kasan: fix zeroing vmalloc memory with HW_TAGS")
    Signed-off-by: Mark Rutland <mark.rutland@arm.com>
    Cc: Alexander Potapenko <glider@google.com>
    Cc: Andrey Konovalov <andreyknvl@google.com>
    Cc: Andrey Ryabinin <ryabinin.a.a@gmail.com>
    Cc: Dmitry Vyukov <dvyukov@google.com>
```

```
    Cc: Marco Elver <elver@google.com>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 627c5af113f8b950ec6f9d2e17975ac853ccb3f2
Author: Jan Kara <jack@suse.cz>
Date:   Mon Apr 17 19:15:37 2023 +0200

    md/raid5: Improve performance for sequential IO

    commit fc05e06e6098ca2c28f7a10da0e00aeea20fa59e upstream.

    Commit 7e55c60acfbb ("md/raid5: Pivot raid5_make_request()") changed the
    order in which requests for underlying disks are created. Since for
    large sequential IO adding of requests frequently races with md_raid5
    thread submitting bios to underlying disks, this results in a change in
    IO pattern because intermediate states of new order of request creation
    result in more smaller discontiguous requests. For RAID5 on top of three
    rotational disks our performance testing revealed this results in
    regression in write throughput:

    iozone -a -s 131072000 -y 4 -q 8 -i 0 -i 1 -R

    before 7e55c60acfbb:
                  KB  reclen   write rewrite    read    reread
           131072000       4  493670  525964   524575   513384
           131072000       8  540467  532880   512028   513703

    after 7e55c60acfbb:
                  KB  reclen   write rewrite    read    reread
           131072000       4  421785  456184   531278   509248
           131072000       8  459283  456354   528449   543834

    To reduce the amount of discontiguous requests we can start generating
    requests with the stripe with the lowest chunk offset as that has the
    best chance of being adjacent to IO queued previously. This improves the
    performance to:
                  KB  reclen   write rewrite    read    reread
           131072000       4  497682  506317   518043   514559
           131072000       8  514048  501886   506453   504319

    restoring big part of the regression.

    Fixes: 7e55c60acfbb ("md/raid5: Pivot raid5_make_request()")
    Cc: stable@vger.kernel.org # v6.0+
    Signed-off-by: Jan Kara <jack@suse.cz>
    Reviewed-by: Logan Gunthorpe <logang@deltatee.com>
    Signed-off-by: Song Liu <song@kernel.org>
    Link: https://lore.kernel.org/r/20230417171537.17899-1-jack@suse.cz
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 9e9efc77efd1956cc244af975240f2513d78a371
Author: Li Nan <linan122@huawei.com>
Date:   Wed Feb 22 12:10:00 2023 +0800

    md/raid10: fix null-ptr-deref in raid10_sync_request

    commit a405c6f0229526160aa3f177f65e20c86fce84c5 upstream.

    init_resync() inits mempool and sets conf->have_replacemnt at the beginning
    of sync, close_sync() frees the mempool when sync is completed.

    After [1] recovery might be skipped and init_resync() is called but
    close_sync() is not. null-ptr-deref occurs with r10bio->dev[i].repl_bio.
```

```
    The following is one way to reproduce the issue.

      1) create a array, wait for resync to complete, mddev->recovery_cp is set
         to MaxSector.
      2) recovery is woken and it is skipped. conf->have_replacement is set to
         0 in init_resync(). close_sync() not called.
      3) some io errors and rdev A is set to WantReplacement.
      4) a new device is added and set to A's replacement.
      5) recovery is woken, A have replacement, but conf->have_replacemnt is
         0. r10bio->dev[i].repl_bio will not be alloced and null-ptr-deref
         occurs.

    Fix it by not calling init_resync() if recovery skipped.

    [1] commit 7e83ccbecd60 ("md/raid10: Allow skipping recovery when clean arrays are
assembled")
    Fixes: 7e83ccbecd60 ("md/raid10: Allow skipping recovery when clean arrays are
assembled")
    Cc: stable@vger.kernel.org
    Signed-off-by: Li Nan <linan122@huawei.com>
    Signed-off-by: Song Liu <song@kernel.org>
    Link: https://lore.kernel.org/r/20230222041000.3341651-3-linan666@huaweicloud.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3cd4698f126e6e21975a924d0ee56a2c0b900ec7
Author: Christoph Böhmwalder <christoph.boehmwalder@linbit.com>
Date:   Wed May 3 14:19:37 2023 +0200

    drbd: correctly submit flush bio on barrier

    commit 3899d94e3831ee07ea6821c032dc297aec80586a upstream.

    When we receive a flush command (or "barrier" in DRBD), we currently use
    a REQ_OP_FLUSH with the REQ_PREFLUSH flag set.

    The correct way to submit a flush bio is by using a REQ_OP_WRITE without
    any data, and set the REQ_PREFLUSH flag.

    Since commit b4a6bb3a67aa ("block: add a sanity check for non-write
    flush/fua bios"), this triggers a warning in the block layer, but this
    has been broken for quite some time before that.

    So use the correct set of flags to actually make the flush happen.

    Cc: Christoph Hellwig <hch@infradead.org>
    Cc: stable@vger.kernel.org
    Fixes: f9ff0da56437 ("drbd: allow parallel flushes for multi-volume resources")
    Reported-by: Thomas Voegtle <tv@lio96.de>
    Signed-off-by: Christoph Böhmwalder <christoph.boehmwalder@linbit.com>
    Reviewed-by: Christoph Hellwig <hch@lst.de>
    Link: https://lore.kernel.org/r/20230503121937.17232-1-christoph.boehmwalder@linbit.com
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit af1762a38510785003f15a018cb9341197c7dc88
Author: Jan Kara <jack@suse.cz>
Date:   Fri Apr 28 14:41:40 2023 +0200

    mm: do not reclaim private data from pinned page

    commit d824ec2a154677f63c56cc71ffe4578274f6e32e upstream.

    If the page is pinned, there's no point in trying to reclaim it.
    Furthermore if the page is from the page cache we don't want to reclaim
```

```
    fs-private data from the page because the pinning process may be writing
    to the page at any time and reclaiming fs private info on a dirty page can
    upset the filesystem (see link below).

    Link: https://lore.kernel.org/linux-mm/20180103100430.GE4911@quack2.suse.cz
    Link: https://lkml.kernel.org/r/20230428124140.30166-1-jack@suse.cz
    Signed-off-by: Jan Kara <jack@suse.cz>
    Reviewed-by: Matthew Wilcox (Oracle) <willy@infradead.org>
    Reviewed-by: Lorenzo Stoakes <lstoakes@gmail.com>
    Reviewed-by: Christoph Hellwig <hch@lst.de>
    Reviewed-by: John Hubbard <jhubbard@nvidia.com>
    Acked-by: David Hildenbrand <david@redhat.com>
    Acked-by: Peter Xu <peterx@redhat.com>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 5b29661669cb65b9750a3cf70ed3eaf947b92167
Author: Ryusuke Konishi <konishi.ryusuke@gmail.com>
Date:   Mon May 1 04:30:46 2023 +0900

    nilfs2: fix infinite loop in nilfs_mdt_get_block()

    commit a6a491c048882e7e424d407d32cba0b52d9ef2bf upstream.

    If the disk image that nilfs2 mounts is corrupted and a virtual block
    address obtained by block lookup for a metadata file is invalid,
    nilfs_bmap_lookup_at_level() may return the same internal return code as
    -ENOENT, meaning the block does not exist in the metadata file.

    This duplication of return codes confuses nilfs_mdt_get_block(), causing
    it to read and create a metadata block indefinitely.

    In particular, if this happens to the inode metadata file, ifile,
    semaphore i_rwsem can be left held, causing task hangs in lock_mount.

    Fix this issue by making nilfs_bmap_lookup_at_level() treat virtual block
    address translation failures with -ENOENT as metadata corruption instead
    of returning the error code.

    Link: https://lkml.kernel.org/r/20230430193046.6769-1-konishi.ryusuke@gmail.com
    Signed-off-by: Ryusuke Konishi <konishi.ryusuke@gmail.com>
    Tested-by: Ryusuke Konishi <konishi.ryusuke@gmail.com>
    Reported-by: syzbot+221d75710bde87fa0e97@syzkaller.appspotmail.com
      Link: https://syzkaller.appspot.com/bug?extid=221d75710bde87fa0e97
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 55f7810632f993cff622a0ddbc7c865892294b61
Author: Ryusuke Konishi <konishi.ryusuke@gmail.com>
Date:   Thu Apr 27 10:15:26 2023 +0900

    nilfs2: do not write dirty data after degenerating to read-only

    commit 28a65b49eb53e172d23567005465019658bfdb4d upstream.

    According to syzbot's report, mark_buffer_dirty() called from
    nilfs_segctor_do_construct() outputs a warning with some patterns after
    nilfs2 detects metadata corruption and degrades to read-only mode.

    After such read-only degeneration, page cache data may be cleared through
    nilfs_clear_dirty_page() which may also clear the uptodate flag for their
    buffer heads.  However, even after the degeneration, log writes are still
    performed by unmount processing etc., which causes mark_buffer_dirty() to
```

```
    be called for buffer heads without the "uptodate" flag and causes the
    warning.

    Since any writes should not be done to a read-only file system in the
    first place, this fixes the warning in mark_buffer_dirty() by letting
    nilfs_segctor_do_construct() abort early if in read-only mode.

    This also changes the retry check of nilfs_segctor_write_out() to avoid
    unnecessary log write retries if it detects -EROFS that
    nilfs_segctor_do_construct() returned.

    Link: https://lkml.kernel.org/r/20230427011526.13457-1-konishi.ryusuke@gmail.com
    Signed-off-by: Ryusuke Konishi <konishi.ryusuke@gmail.com>
    Tested-by: Ryusuke Konishi <konishi.ryusuke@gmail.com>
    Reported-by: syzbot+2af3bc9585be7f23f290@syzkaller.appspotmail.com
      Link: https://syzkaller.appspot.com/bug?extid=2af3bc9585be7f23f290
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 9bfbc4fbe38759b25a21c6ea3a0c06e70d4caa8d
Author: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
Date:   Wed May 3 16:41:02 2023 +0200

    ASoC: codecs: wcd938x: fix accessing regmap on unattached devices

    commit 84822215acd15bd86a7759a835271e63bba83a7b upstream.

    The WCD938x comes with three devices on two Linux drivers:
    1. RX Soundwire device (wcd938x-sdw.c driver),
    2. TX Soundwire device, which is used to access devices via regmap (also
       wcd938x-sdw.c driver),
    3. platform device (wcd938x.c driver) - glue and component master,
       actually having most of the code using TX Soundwire device regmap.

    When RX and TX Soundwire devices probe, the component master (platform
    device) bind tries to write micbias configuration via TX Soundwire
    regmap.  This might happen before TX Soundwire enumerates, so the regmap
    access fails.  On Qualcomm SM8550 board with WCD9385:

      qcom-soundwire 6d30000.soundwire-controller: Qualcomm Soundwire controller v2.0.0
Registered
      wcd938x_codec audio-codec: bound sdw:0:0217:010d:00:4 (ops wcd938x_sdw_component_ops)
      wcd938x_codec audio-codec: bound sdw:0:0217:010d:00:3 (ops wcd938x_sdw_component_ops)
      qcom-soundwire 6ad0000.soundwire-controller: swrm_wait_for_wr_fifo_avail err write
overflow

    Fix the issue by:
    1. Moving the regmap creation from platform device to TX Soundwire
       device.  The regmap settings are moved as-is with one difference:
       making the wcd938x_regmap_config const.
    2. Using regmap in cache only mode till the actual TX Soundwire device
       enumerates and then sync the regmap cache.

    Cc: <stable@vger.kernel.org> # v3.14+
    Signed-off-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Reviewed-by: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com>
    Message-Id: <20230503144102.242240-1-krzysztof.kozlowski@linaro.org>
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e0b4a8246de730f08f1026f98af230a0f8a90ab1
Author: Kai-Heng Feng <kai.heng.feng@canonical.com>
Date:   Fri May 5 14:59:23 2023 +0200
```

```
    ALSA: hda/realtek: Fix mute and micmute LEDs for an HP laptop

    commit 56fc217f0db4fc78e02a1b8450df06389474a5e5 upstream.

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

    Signed-off-by: Kai-Heng Feng <kai.heng.feng@canonical.com>
    Cc: <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20230505125925.543601-1-kai.heng.feng@canonical.com
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b6628c80c463362874af298bb304a9a660cb88f2
Author: Caleb Harper <calebharp2005@gmail.com>
Date:   Wed May 3 12:50:26 2023 -0500

    ALSA: hda/realtek: support HP Pavilion Aero 13-be0xxx Mute LED

    commit e7477cb97607b373d175a759c8c0270a640ab3f2 upstream.

    This patch adds support for the mute LED on the HP Pavilion Aero Laptop
    13-be0xxx. The current behavior is that the LED does not turn on at any
    time and does not indicate to the user whether the sound is muted.

    The solution is to add a PCI quirk to properly recognize and support the
    LED on this device.

    This change has been tested on the device in question using modified
    versions of kernels 6.0.7-6.2.12 on Arch Linux.

    Signed-off-by: Caleb Harper <calebharp2005@gmail.com>
    Cc: <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20230503175026.6796-1-calebharp2005@gmail.com
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f026727a9692ee740afe72a66e25cb5f919db06f
Author: Mark Asselstine <asselsm@gmail.com>
Date:   Mon May 1 19:13:46 2023 -0400

    ALSA: hda/realtek: Add quirk for ASUS UM3402YAR using CS35L41

    commit 7e2d06628aab6324e1ac885910a52f4c038d4043 upstream.

    This Asus Zenbook laptop uses Realtek HDA codec combined with
    2xCS35L41 Amplifiers using I2C with External Boost.

    Signed-off-by: Mark Asselstine <asselsm@gmail.com>
    Cc: <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20230501231346.54979-1-asselsm@gmail.com
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ce88eafc12644fc1abe77ec98206004dafa11499
Author: Vitaly Rodionov <vitalyr@opensource.cirrus.com>
Date:   Thu Apr 27 12:04:52 2023 +0100

    ALSA: hda/realtek: Add quirk for ThinkPad P1 Gen 6

    commit 067eb084592819ad59d07afcb5de3e61cee2757c upstream.

    Lenovo ThinkPad P1 Gen 6 laptop has 2 CS35L41 amplifies
    on I2C bus connected to Realtek codec.
```

```
    Signed-off-by: Vitaly Rodionov <vitalyr@opensource.cirrus.com>
    Cc: <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20230427110452.13787-1-vitalyr@opensource.cirrus.com
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b292e688bf8e82bd82db5fcb428fd2f68020b126
Author: Geraldo Nascimento <geraldogabriel@gmail.com>
Date:   Wed May 3 18:02:06 2023 -0300

    ALSA: usb-audio: Add quirk for Pioneer DDJ-800

    commit 7501f472977df233d039d86c6981e0641708e1ca upstream.

    One more Pioneer quirk, this time for DDJ-800, which is quite similar like
    other DJ DDJ models but with slightly different EPs or channels.

    Signed-off-by: Geraldo Nascimento <geraldogabriel@gmail.com>
    Tested-by: Grégory Desor <gregory.desor@free.fr>
    Cc: <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/ZFLLzgEcsSF5aIHG@geday
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ce914d3a8e29c34b4f423fe26794d0d327baff33
Author: Helge Deller <deller@gmx.de>
Date:   Wed Mar 15 19:25:15 2023 +0100

    parisc: Ensure page alignment in flush functions

    commit d755bd2caeb47fd806e12399fe8b56798fa5d2cc upstream.

    Matthew Wilcox noticed, that if ARCH_HAS_FLUSH_ON_KUNMAP is defined
    (which is the case for PA-RISC), __kunmap_local() calls
    kunmap_flush_on_unmap(), which may call the parisc flush functions with
    a non-page-aligned address and thus the page might not be fully flushed.

    This patch ensures that flush_kernel_dcache_page_asm() and
    flush_kernel_dcache_page_asm() will always operate on page-aligned
    addresses.

    Signed-off-by: Helge Deller <deller@gmx.de>
    Cc: <stable@vger.kernel.org> # v6.0+
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 409042e602958a2ecffe2925001d81b74cabae97
Author: Helge Deller <deller@gmx.de>
Date:   Wed May 3 16:39:56 2023 +0200

    parisc: Fix argument pointer in real64_call_asm()

    commit 6e3220ba3323a2c24be834aebf5d6e9f89d0993f upstream.

    Fix the argument pointer (ap) to point to real-mode memory
    instead of virtual memory.

    It's interesting that this issue hasn't shown up earlier, as this could
    have happened with any 64-bit PDC ROM code.

    I just noticed it because I suddenly faced a HPMC while trying to execute
    the 64-bit STI ROM code of an Visualize-FXe graphics card for the STI
    text console.

    Signed-off-by: Helge Deller <deller@gmx.de>
    Cc: <stable@vger.kernel.org>
```

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

commit 756a5358a098b329bddf7bee5e5e883e41785833
Author: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
Date:   Thu Mar 30 09:13:33 2023 +0200

    ASoC: dt-bindings: qcom,lpass-rx-macro: correct minItems for clocks

    commit 59257015ac8813d2430988aa01c2f4609a60e8e7 upstream.

    The RX macro codec comes on some platforms in two variants - ADSP
    and ADSP bypassed - thus the clock-names varies from 3 to 5.  The clocks
    must vary as well:

      sc7280-idp.dtb: codec@3200000: clocks: [[202, 8], [202, 7], [203]] is too short

    Fixes: 852fda58d99a ("ASoC: qcom: dt-bindings: Update bindings for clocks in lpass
digital codes")
    Cc: <stable@vger.kernel.org>
    Acked-by: Rob Herring <robh@kernel.org>
    Signed-off-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Link: https://lore.kernel.org/r/20230330071333.24308-1-krzysztof.kozlowski@linaro.org
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 4e584127ec2bd42a37c88badb49df409f21fa40a
Author: Jeffrey Hugo <quic_jhugo@quicinc.com>
Date:   Fri Mar 24 10:13:04 2023 -0600

    bus: mhi: host: Range check CHDBOFF and ERDBOFF

    commit 6a0c637bfee69a74c104468544d9f2a6579626d0 upstream.

    If the value read from the CHDBOFF and ERDBOFF registers is outside the
    range of the MHI register space then an invalid address might be computed
    which later causes a kernel panic.  Range check the read value to prevent
    a crash due to bad data from the device.

    Fixes: 6cd330ae76ff ("bus: mhi: core: Add support for ringing channel/event ring
doorbells")
    Cc: stable@vger.kernel.org
    Signed-off-by: Jeffrey Hugo <quic_jhugo@quicinc.com>
    Reviewed-by: Pranjal Ramajor Asha Kanojiya <quic_pkanojiy@quicinc.com>
    Reviewed-by: Manivannan Sadhasivam <mani@kernel.org>
    Link: https://lore.kernel.org/r/1679674384-27209-1-git-send-email-quic_jhugo@quicinc.com
    Signed-off-by: Manivannan Sadhasivam <manivannan.sadhasivam@linaro.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit fe1e47158ce119c5335997b6d08069d911768015
Author: Jeffrey Hugo <quic_jhugo@quicinc.com>
Date:   Mon Apr 10 09:58:12 2023 -0600

    bus: mhi: host: Use mhi_tryset_pm_state() for setting fw error state

    commit 1d1493bdc25f498468a606a4ece947d155cfa3a9 upstream.

    If firmware loading fails, the controller's pm_state is updated to
    MHI_PM_FW_DL_ERR unconditionally.  This can corrupt the pm_state as the
    update is not done under the proper lock, and also does not validate
    the state transition.  The firmware loading can fail due to a detected
    syserr, but if MHI_PM_FW_DL_ERR is unconditionally set as the pm_state,
    the handling of the syserr can break when it attempts to transition from
    syserr detect, to syserr process.

    By grabbing the lock, we ensure we don't race with some other pm_state
```

```
    update.  By using mhi_try_set_pm_state(), we check that the transition
    to MHI_PM_FW_DL_ERR is valid via the state machine logic.  If it is not
    valid, then some other transition is occurring like syserr processing, and
    we assume that will resolve the firmware loading error.

    Fixes: 12e050c77be0 ("bus: mhi: core: Move to an error state on any firmware load
failure")
    Cc: stable@vger.kernel.org
    Signed-off-by: Jeffrey Hugo <quic_jhugo@quicinc.com>
    Reviewed-by: Carl Vanderlip <quic_carlv@quicinc.com>
    Reviewed-by: Manivannan Sadhasivam <mani@kernel.org>
    Link: https://lore.kernel.org/r/1681142292-27571-3-git-send-email-quic_jhugo@quicinc.com
    Signed-off-by: Manivannan Sadhasivam <manivannan.sadhasivam@linaro.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 8db31cf8c8c2523c013ed0d1157cc555bf991cd6
Author: Jeffrey Hugo <quic_jhugo@quicinc.com>
Date:   Mon Apr 10 09:58:11 2023 -0600

    bus: mhi: host: Remove duplicate ee check for syserr

    commit d469d9448a0f1a33c175d3280b1542fa0158ad7a upstream.

    If we detect a system error via intvec, we only process the syserr if the
    current ee is different than the last observed ee.  The reason for this
    check is to prevent bhie from running multiple times, but with the single
    queue handling syserr, that is not possible.

    The check can cause an issue with device recovery.  If PBL loads a bad SBL
    via BHI, but that SBL hangs before notifying the host of an ee change,
    then issuing soc_reset to crash the device and retry (after supplying a
    fixed SBL) will not recover the device as the host will observe a PBL->PBL
    transition and not process the syserr.  The device will be stuck until
    either the driver is reloaded, or the host is rebooted.  Instead, remove
    the check so that we can attempt to recover the device.

    Fixes: ef2126c4e2ea ("bus: mhi: core: Process execution environment changes serially")
    Cc: stable@vger.kernel.org
    Signed-off-by: Jeffrey Hugo <quic_jhugo@quicinc.com>
    Reviewed-by: Carl Vanderlip <quic_carlv@quicinc.com>
    Reviewed-by: Manivannan Sadhasivam <mani@kernel.org>
    Link: https://lore.kernel.org/r/1681142292-27571-2-git-send-email-quic_jhugo@quicinc.com
    Signed-off-by: Manivannan Sadhasivam <manivannan.sadhasivam@linaro.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 28e03bcda33acaa26c5a3d05d2f395adf434a416
Author: Dan Williams <dan.j.williams@intel.com>
Date:   Fri Apr 14 11:54:11 2023 -0700

    cxl/port: Scan single-target ports for decoders

    commit 7bba261e0aa6e8e5f28a3a3def8338b6512534ee upstream.

    Do not assume that a single-target port falls back to a passthrough
    decoder configuration. Scan for decoders and only fallback after probing
    that the HDM decoder capability is not present.

    One user visible affect of this bug is the inability to enumerate
    present CXL regions as the decoder settings for the present decoders are
    skipped.

    Fixes: d17d0540a0db ("cxl/core/hdm: Add CXL standard decoder enumeration to the core")
    Reported-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Link: http://lore.kernel.org/r/20230227153128.8164-1-Jonathan.Cameron@huawei.com
    Cc: <stable@vger.kernel.org>
```

```
    Reviewed-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Reviewed-by: Dave Jiang <dave.jiang@intel.com>
    Reviewed-by: Alison Schofield <alison.schofield@intel.com>
    Link: https://lore.kernel.org/r/168149845130.792294.3210421233937427962.stgit@dwillia2-
xfh.jf.intel.com
    Signed-off-by: Dan Williams <dan.j.williams@intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 0336234656f9995715a279a79b788d711ac20725
Author: Dan Williams <dan.j.williams@intel.com>
Date:   Fri Apr 14 11:54:00 2023 -0700

    cxl/hdm: Use 4-byte reads to retrieve HDM decoder base+limit

    commit 1423885c84a5b3a53b79bcf241b18124d0d7cba6 upstream.

    The CXL specification mandates that 4-byte registers must be accessed
    with 4-byte access cycles. CXL 3.0 8.2.3 "Component Register Layout and
    Definition" states that the behavior is undefined if (2) 32-bit
    registers are accessed as an 8-byte quantity. It turns out that at least
    one hardware implementation is sensitive to this in practice. The @size
    variable results in zero with:

        size = readq(hdm + CXL_HDM_DECODER0_SIZE_LOW_OFFSET(which));

    ...and the correct size with:

        lo = readl(hdm + CXL_HDM_DECODER0_SIZE_LOW_OFFSET(which));
        hi = readl(hdm + CXL_HDM_DECODER0_SIZE_HIGH_OFFSET(which));
        size = (hi << 32) + lo;

    Fixes: d17d0540a0db ("cxl/core/hdm: Add CXL standard decoder enumeration to the core")
    Cc: <stable@vger.kernel.org>
    Reviewed-by: Dave Jiang <dave.jiang@intel.com>
    Reviewed-by: Alison Schofield <alison.schofield@intel.com>
    Link: https://lore.kernel.org/r/168149844056.792294.8224490474529733736.stgit@dwillia2-
xfh.jf.intel.com
    Signed-off-by: Dan Williams <dan.j.williams@intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 89e49a5770b7b84fd5878b269c67d1b23aea1dec
Author: Dan Williams <dan.j.williams@intel.com>
Date:   Fri Apr 14 11:53:55 2023 -0700

    cxl/hdm: Fail upon detecting 0-sized decoders

    commit 7701c8bef4f14bd9f7940c6ed0e6a73584115a96 upstream.

    Decoders committed with 0-size lead to later crashes on shutdown as
    __cxl_dpa_release() assumes a 'struct resource' has been established in
    the in 'cxlds->dpa_res'. Just fail the driver load in this instance
    since there are deeper problems with the enumeration or the setup when
    this happens.

    Fixes: 9c57cde0dcbd ("cxl/hdm: Enumerate allocated DPA")
    Cc: <stable@vger.kernel.org>
    Reviewed-by: Dave Jiang <dave.jiang@intel.com>
    Reviewed-by: Alison Schofield <alison.schofield@intel.com>
    Link: https://lore.kernel.org/r/168149843516.792294.11872242648319572632.stgit@dwillia2-
xfh.jf.intel.com
    Signed-off-by: Dan Williams <dan.j.williams@intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a973b4c82447f7bf2107a386c2d35bc4fb5b697f
Author: Dave Chinner <dchinner@redhat.com>
```

```
Date:   Wed Apr 12 15:48:50 2023 +1000

    xfs: don't consider future format versions valid

    commit aa88019851a85df80cb77f143758b13aee09e3d9 upstream.

    In commit fe08cc504448 we reworked the valid superblock version
    checks. If it is a V5 filesystem, it is always valid, then we
    checked if the version was less than V4 (reject) and then checked
    feature fields in the V4 flags to determine if it was valid.

    What we missed was that if the version is not V4 at this point,
    we shoudl reject the fs. i.e. the check current treats V6+
    filesystems as if it was a v4 filesystem. Fix this.

    cc: stable@vger.kernel.org
    Fixes: fe08cc504448 ("xfs: open code sb verifier feature checks")
    Signed-off-by: Dave Chinner <dchinner@redhat.com>
    Reviewed-by: Darrick J. Wong <djwong@kernel.org>
    Signed-off-by: Dave Chinner <david@fromorbit.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3b4d84bd5a7b847d49ece48dd87cce93b4750b79
Author: Alexander Aring <aahringo@redhat.com>
Date:   Mon Mar 6 15:48:08 2023 -0500

    fs: dlm: fix DLM_IFL_CB_PENDING gets overwritten

    commit a034c1370ded2ae6cbdc73a78241b3ed98c86d3d upstream.

    This patch introduce a new internal flag per lkb value to handle
    internal flags which are handled not on wire. The current lkb internal
    flags stored as lkb->lkb_flags are split in upper and lower bits, the
    lower bits are used to share internal flags over wire for other cluster
    wide lkb copies on other nodes.

    In commit 61bed0baa4db ("fs: dlm: use a non-static queue for callbacks")
    we introduced a new internal flag for pending callbacks for the dlm
    callback queue. This flag is protected by the lkb->lkb_cb_lock lock.
    This patch overlooked that on dlm receive path and the mentioned upper
    and lower bits, that dlm will read the flags, mask it and write it
    back. As example receive_flags() in fs/dlm/lock.c. This flag
    manipulation is not done atomically and is not protected by
    lkb->lkb_cb_lock. This has unknown side effects of the current callback
    handling.

    In future we should move to set/clear/test bit functionality and avoid
    read, mask and writing back flag values. In later patches we will move
    the upper parts to the new introduced internal lkb flags which are not
    shared between other cluster nodes to the new non shared internal flag
    field to avoid similar issues.

    Cc: stable@vger.kernel.org
    Fixes: 61bed0baa4db ("fs: dlm: use a non-static queue for callbacks")
    Reported-by: Bob Peterson <rpeterso@redhat.com>
    Signed-off-by: Alexander Aring <aahringo@redhat.com>
    Signed-off-by: David Teigland <teigland@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ae6e935618d99cdba11eab4714092e7e5f13cf7e
Author: Xiubo Li <xiubli@redhat.com>
Date:   Wed Apr 19 10:39:14 2023 +0800

    ceph: fix potential use-after-free bug when trimming caps
```

```
    commit aaf67de78807c59c35bafb5003d4fb457c764800 upstream.

    When trimming the caps and just after the 'session->s_cap_lock' is
    released in ceph_iterate_session_caps() the cap maybe removed by
    another thread, and when using the stale cap memory in the callbacks
    it will trigger use-after-free crash.

    We need to check the existence of the cap just after the 'ci->i_ceph_lock'
    being acquired. And do nothing if it's already removed.

    Cc: stable@vger.kernel.org
    Link: https://tracker.ceph.com/issues/43272
    Signed-off-by: Xiubo Li <xiubli@redhat.com>
    Reviewed-by: Luís Henriques <lhenriques@suse.de>
    Signed-off-by: Ilya Dryomov <idryomov@gmail.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 517ddc0259d7a7231486bdafde8035c478bc4088
Author: Mårten Lindahl <marten.lindahl@axis.com>
Date:   Thu Mar 30 16:40:59 2023 +0200

    ubifs: Fix memory leak in do_rename

    commit 3a36d20e012903f45714df2731261fdefac900cb upstream.

    If renaming a file in an encrypted directory, function
    fscrypt_setup_filename allocates memory for a file name. This name is
    never used, and before returning to the caller the memory for it is not
    freed.

    When running kmemleak on it we see that it is registered as a leak. The
    report below is triggered by a simple program 'rename' that renames a
    file in an encrypted directory:

      unreferenced object 0xffff888101502840 (size 32):
        comm "rename", pid 9404, jiffies 4302582475 (age 435.735s)
        backtrace:
          __kmem_cache_alloc_node
          __kmalloc
          fscrypt_setup_filename
          do_rename
          ubifs_rename
          vfs_rename
          do_renameat2

    To fix this we can remove the call to fscrypt_setup_filename as it's not
    needed.

    Fixes: 278d9a243635f26 ("ubifs: Rename whiteout atomically")
    Reported-by: Zhihao Cheng <chengzhihao1@huawei.com>
    Signed-off-by: Mårten Lindahl <marten.lindahl@axis.com>
    Reviewed-by: Zhihao Cheng <chengzhihao1@huawei.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Richard Weinberger <richard@nod.at>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1e43d4284bdc3bd34bd770fea13910ac37ab0618
Author: Mårten Lindahl <marten.lindahl@axis.com>
Date:   Thu Mar 30 11:32:14 2023 +0200

    ubifs: Free memory for tmpfile name

    commit 1fb815b38bb31d6af9bd0540b8652a0d6fe6cfd3 upstream.

    When opening a ubifs tmpfile on an encrypted directory, function
```

```
    fscrypt_setup_filename allocates memory for the name that is to be
    stored in the directory entry, but after the name has been copied to the
    directory entry inode, the memory is not freed.

    When running kmemleak on it we see that it is registered as a leak. The
    report below is triggered by a simple program 'tmpfile' just opening a
    tmpfile:

      unreferenced object 0xffff88810178f380 (size 32):
        comm "tmpfile", pid 509, jiffies 4294934744 (age 1524.742s)
        backtrace:
          __kmem_cache_alloc_node
          __kmalloc
          fscrypt_setup_filename
          ubifs_tmpfile
          vfs_tmpfile
          path_openat

    Free this memory after it has been copied to the inode.

    Signed-off-by: Mårten Lindahl <marten.lindahl@axis.com>
    Reviewed-by: Zhihao Cheng <chengzhihao1@huawei.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Richard Weinberger <richard@nod.at>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 74a27fde89b97638e79e489cc237ee11ad783124
Author: Wang YanQing <udknight@gmail.com>
Date:   Tue Mar 28 23:35:34 2023 +0800

    ubi: Fix return value overwrite issue in try_write_vid_and_data()

    commit 31a149d5c13c4cbcf97de3435817263a2d8c9d6e upstream.

    The commit 2d78aee426d8 ("UBI: simplify LEB write and atomic LEB change code")
    adds helper function, try_write_vid_and_data(), to simplify the code, but this
    helper function has bug, it will return 0 (success) when ubi_io_write_vid_hdr()
    or the ubi_io_write_data() return error number (-EIO, etc), because the return
    value of ubi_wl_put_peb() will overwrite the original return value.

    This issue will cause unexpected data loss issue, because the caller of this
    function and UBIFS willn't know the data is lost.

    Fixes: 2d78aee426d8 ("UBI: simplify LEB write and atomic LEB change code")
    Cc: stable@vger.kernel.org
    Signed-off-by: Wang YanQing <udknight@gmail.com>
    Reviewed-by: Zhihao Cheng <chengzhihao1@huawei.com>
    Signed-off-by: Richard Weinberger <richard@nod.at>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a6da0ab9847779e05a7416c7a98148b549de69ef
Author: Zhihao Cheng <chengzhihao1@huawei.com>
Date:   Wed Mar 1 20:29:19 2023 +0800

    ubifs: Fix memleak when insert_old_idx() failed

    commit b5fda08ef213352ac2df7447611eb4d383cce929 upstream.

    Following process will cause a memleak for copied up znode:

    dirty_cow_znode
      zn = copy_znode(c, znode);
      err = insert_old_idx(c, zbr->lnum, zbr->offs);
      if (unlikely(err))
         return ERR_PTR(err);   // No one refers to zn.
```

```
    Fetch a reproducer in [Link].

    Function copy_znode() is split into 2 parts: resource allocation
    and znode replacement, insert_old_idx() is split in similar way,
    so resource cleanup could be done in error handling path without
    corrupting metadata(mem & disk).
    It's okay that old index inserting is put behind of add_idx_dirt(),
    old index is used in layout_leb_in_gaps(), so the two processes do
    not depend on each other.

    Link: https://bugzilla.kernel.org/show_bug.cgi?id=216705
    Fixes: 1e51764a3c2a ("UBIFS: add new flash file system")
    Cc: stable@vger.kernel.org
    Signed-off-by: Zhihao Cheng <chengzhihao1@huawei.com>
    Signed-off-by: Richard Weinberger <richard@nod.at>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c2d3fd857f4c2bc881be3a1da694f5ce61a589be
Author: Zhihao Cheng <chengzhihao1@huawei.com>
Date:   Wed Mar 1 20:29:18 2023 +0800

    Revert "ubifs: dirty_cow_znode: Fix memleak in error handling path"

    commit 7d01cb27f6aebc54efbe28d8961a973b8f795b13 upstream.

    This reverts commit 122deabfe1428 (ubifs: dirty_cow_znode: Fix memleak
    in error handling path).
    After commit 122deabfe1428 applied, if insert_old_idx() failed, old
    index neither exists in TNC nor in old-index tree. Which means that
    old index node could be overwritten in layout_leb_in_gaps(), then
    ubifs image will be corrupted in power-cut.

    Fixes: 122deabfe1428 (ubifs: dirty_cow_znode: Fix memleak ... path)
    Cc: stable@vger.kernel.org
    Signed-off-by: Zhihao Cheng <chengzhihao1@huawei.com>
    Signed-off-by: Richard Weinberger <richard@nod.at>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 5fde414b10399c1d8465cc303c9229eb321dd845
Author: Andrew Jones <ajones@ventanamicro.com>
Date:   Thu Apr 27 18:36:26 2023 +0200

    RISC-V: Align SBI probe implementation with spec

    commit 41cad8284d5e6bf1d49d3c10a6b52ee1ae866a20 upstream.

    sbi_probe_extension() is specified with "Returns 0 if the given SBI
    extension ID (EID) is not available, or 1 if it is available unless
    defined as any other non-zero value by the implementation."
    Additionally, sbiret.value is a long. Fix the implementation to
    ensure any nonzero long value is considered a success, rather
    than only positive int values.

    Fixes: b9dcd9e41587 ("RISC-V: Add basic support for SBI v0.2")
    Signed-off-by: Andrew Jones <ajones@ventanamicro.com>
    Reviewed-by: Conor Dooley <conor.dooley@microchip.com>
    Cc: stable@vger.kernel.org
    Link: https://lore.kernel.org/r/20230427163626.101042-1-ajones@ventanamicro.com
    Signed-off-by: Palmer Dabbelt <palmer@rivosinc.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ccafe4b8a08725b671d4334dc44bab57f7717979
Author: Kishon Vijay Abraham I <kvijayab@amd.com>
Date:   Wed Apr 5 13:03:17 2023 +0000
```

```
    iommu/amd: Fix "Guest Virtual APIC Table Root Pointer" configuration in IRTE

    commit ccc62b827775915a9b82db42a29813d04f92df7a upstream.

    commit b9c6ff94e43a ("iommu/amd: Re-factor guest virtual APIC
    (de-)activation code") while refactoring guest virtual APIC
    activation/de-activation code, stored information for activate/de-activate
    in "struct amd_ir_data". It used 32-bit integer data type for storing the
    "Guest Virtual APIC Table Root Pointer" (ga_root_ptr), though the
    "ga_root_ptr" is actually a 40-bit field in IRTE (Interrupt Remapping
    Table Entry).

    This causes interrupts from PCIe devices to not reach the guest in the case
    of PCIe passthrough with SME (Secure Memory Encryption) enabled as _SME_
    bit in the "ga_root_ptr" is lost before writing it to the IRTE.

    Fix it by using 64-bit data type for storing the "ga_root_ptr". While at
    that also change the data type of "ga_tag" to u32 in order to match
    the IOMMU spec.

    Fixes: b9c6ff94e43a ("iommu/amd: Re-factor guest virtual APIC (de-)activation code")
    Cc: stable@vger.kernel.org # v5.4+
    Reported-by: Alejandro Jimenez <alejandro.j.jimenez@oracle.com>
    Reviewed-by: Suravee Suthikulpanit <suravee.suthikulpanit@amd.com>
    Signed-off-by: Kishon Vijay Abraham I <kvijayab@amd.com>
    Link: https://lore.kernel.org/r/20230405130317.9351-1-kvijayab@amd.com
    Signed-off-by: Joerg Roedel <jroedel@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 12c790dfe5d343f5088eeb41b3d0a62e469a8e2d
Author: Tim Huang <tim.huang@amd.com>
Date:   Wed Mar 22 14:39:16 2023 +0800

    drm/amd/pm: re-enable the gfx imu when smu resume

    commit f7f28f268b861c29dd18086bb636abedf0ff59ff upstream.

    If the gfx imu is poweroff when suspend, then
    it need to be re-enabled when resume.

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

commit c5a513f55dc3c0e284dc46f3dac342bdd50d4abe
Author: Ondrej Mosnacek <omosnace@redhat.com>
Date:   Wed May 3 16:01:14 2023 +0200

    tracing: Fix permissions for the buffer_percent file

    commit 4f94559f40ad06d627c0fdfc3319cec778a2845b upstream.

    This file defines both read and write operations, yet it is being
    created as read-only. This means that it can't be written to without the
    CAP_DAC_OVERRIDE capability. Fix the permissions to allow root to write
    to it without the need to override DAC perms.

    Link: https://lore.kernel.org/linux-trace-kernel/20230503140114.3280002-1-
omosnace@redhat.com

    Cc: stable@vger.kernel.org
    Cc: Masami Hiramatsu <mhiramat@kernel.org>
    Fixes: 03329f993978 ("tracing: Add tracefs file buffer_percentage")
```

```
    Signed-off-by: Ondrej Mosnacek <omosnace@redhat.com>
    Signed-off-by: Steven Rostedt (Google) <rostedt@goodmis.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a4cb459a35ca8fe58f4b27d7de5736b4ecddba8d
Author: Srinivas Pandruvada <srinivas.pandruvada@linux.intel.com>
Date:   Wed May 3 16:38:50 2023 -0700

    thermal: intel: powerclamp: Fix NULL pointer access issue

    commit b5d68f84f4c62c78bc3d004911d80da5aa22df8b upstream.

    If cur_state for the powerclamp cooling device is set to the default
    minimum state of 0, without setting first to cur_state > 0, this results
    in NULL pointer access.

    This NULL pointer access happens in the powercap core idle-inject
    function idle_inject_set_duration() as there is no NULL check for
    idle_inject_device pointer. This pointer must be allocated by calling
    idle_inject_register() or idle_inject_register_full().

    In the function powerclamp_set_cur_state(), idle_inject_device pointer
    is allocated only when the cur_state > 0. But setting 0 without changing
    to any other state, idle_inject_set_duration() will be called with a
    NULL idle_inject_device pointer.

    To address this, just return from powerclamp_set_cur_state() if the
    current cooling device state is the same as the last one. Since the
    power-up default cooling device state is 0, changing the state to 0
    again here will return without calling idle_inject_set_duration().

    Signed-off-by: Srinivas Pandruvada <srinivas.pandruvada@linux.intel.com>
    Fixes: 8526eb7fc75a ("thermal: intel: powerclamp: Use powercap idle-inject feature")
    Bugzilla: https://bugzilla.kernel.org/show_bug.cgi?id=217386
    Tested-by: Risto A. Paju <teknohog@iki.fi>
    Cc: 6.3+ <stable@kernel.org> # 6.3+
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1c13e69842085735798a2373b08e0699c9fc1e24
Author: Song Shuai <suagrfillet@gmail.com>
Date:   Wed Apr 26 18:00:09 2023 +0800

    riscv: mm: remove redundant parameter of create_fdt_early_page_table

    commit e4ef93edd4e0b022529303db1915766ff9de450e upstream.

    create_fdt_early_page_table() explicitly uses early_pg_dir for
    32-bit fdt mapping and the pgdir parameter is redundant here.
    So remove it and its caller.

    Reviewed-by: Alexandre Ghiti <alexghiti@rivosinc.com>
    Signed-off-by: Song Shuai <suagrfillet@gmail.com>
    Reviewed-by: Conor Dooley <conor.dooley@microchip.com>
    Fixes: ef69d2559fe9 ("riscv: Move early dtb mapping into the fixmap region")
    Cc: stable@vger.kernel.org
    Link: https://lore.kernel.org/r/20230426100009.685435-1-suagrfillet@gmail.com
    Signed-off-by: Palmer Dabbelt <palmer@rivosinc.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ea2e98118b59ed775c7127838fffe4cfe2a8dafa
Author: Reid Tonking <reidt@ti.com>
Date:   Wed Apr 26 14:49:56 2023 -0500

    i2c: omap: Fix standard mode false ACK readings
```

```
    commit c770657bd2611b077ec1e7b1fe6aa92f249399bd upstream.

    Using standard mode, rare false ACK responses were appearing with
    i2cdetect tool. This was happening due to NACK interrupt triggering
    ISR thread before register access interrupt was ready. Removing the
    NACK interrupt's ability to trigger ISR thread lets register access
    ready interrupt do this instead.

    Cc: <stable@vger.kernel.org> # v3.7+
    Fixes: 3b2f8f82dad7 ("i2c: omap: switch to threaded IRQ support")
    Signed-off-by: Reid Tonking <reidt@ti.com>
    Acked-by: Vignesh Raghavendra <vigneshr@ti.com>
    Reviewed-by: Tony Lindgren <tony@atomide.com>
    Signed-off-by: Wolfram Sang <wsa@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 41b330fd04efcaac779166189457b45f1057800d
Author: Song Yoong Siang <yoong.siang.song@intel.com>
Date:   Tue May 2 08:48:06 2023 -0700

    igc: read before write to SRRCTL register

    commit 3ce29c17dc847bf4245e16aad78a7617afa96297 upstream.

    igc_configure_rx_ring() function will be called as part of XDP program
    setup. If Rx hardware timestamp is enabled prio to XDP program setup,
    this timestamp enablement will be overwritten when buffer size is
    written into SRRCTL register.

    Thus, this commit read the register value before write to SRRCTL
    register. This commit is tested by using xdp_hw_metadata bpf selftest
    tool. The tool enables Rx hardware timestamp and then attach XDP program
    to igc driver. It will display hardware timestamp of UDP packet with
    port number 9092. Below are detail of test steps and results.

    Command on DUT:
      sudo ./xdp_hw_metadata <interface name>

    Command on Link Partner:
      echo -n skb | nc -u -q1 <destination IPv4 addr> 9092

    Result before this patch:
      skb hwtstamp is not found!

    Result after this patch:
      found skb hwtstamp = 1677800973.642836757

    Optionally, read PHC to confirm the values obtained are almost the same:
    Command:
      sudo ./testptp -d /dev/ptp0 -g
    Result:
      clock time: 1677800973.913598978 or Fri Mar  3 07:49:33 2023

    Fixes: fc9df2a0b520 ("igc: Enable RX via AF_XDP zero-copy")
    Cc: <stable@vger.kernel.org> # 5.14+
    Signed-off-by: Song Yoong Siang <yoong.siang.song@intel.com>
    Reviewed-by: Jacob Keller <jacob.e.keller@intel.com>
    Reviewed-by: Jesper Dangaard Brouer <brouer@redhat.com>
    Tested-by: Jesper Dangaard Brouer <brouer@redhat.com>
    Tested-by: Naama Meir <naamax.meir@linux.intel.com>
    Signed-off-by: Tony Nguyen <anthony.l.nguyen@intel.com>
    Reviewed-by: Leon Romanovsky <leonro@nvidia.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit 2b9395772adf89ab39bb392176eb2b09fcf70439
Author: Hans de Goede <hdegoede@redhat.com>
Date:   Fri Apr 28 15:23:50 2023 +0200

    ACPI: video: Remove acpi_backlight=video quirk for Lenovo ThinkPad W530

    commit 3db66620ea90b0fd4134b31eabfec16d7b07d7e3 upstream.

    Remove the acpi_backlight=video quirk for Lenovo ThinkPad W530.

    This was intended to help users of the (unsupported) Nvidia binary driver,
    but this has been reported to cause backlight control issues for users
    who have the gfx configured in hybrid (dual-GPU) mode, so drop this.

    The Nvidia binary driver should call acpi_video_register_backlight()
    when necessary and this has been reported to Nvidia.

    Until this is fixed Nvidia binary driver users can work around this by
    passing "acpi_backlight=video" on the kernel commandline (with the latest
    6.1.y or newer stable series, kernels < 6.1.y don't need this).

    Fixes: a5b2781dcab2 ("ACPI: video: Add acpi_backlight=video quirk for Lenovo ThinkPad
W530")
    Reported-by: Русев Путин <rockeraliexpress@gmail.com>
    Link: https://lore.kernel.org/linux-
acpi/CAK4BXn0ngZRmzx1bodAF8nmYj0PWdUXzPGHofRrsyZj8MBpcVA@mail.gmail.com/
    Cc: 6.1+ <stable@vger.kernel.org> # 6.1+
    Signed-off-by: Hans de Goede <hdegoede@redhat.com>
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 920d5dd2d041484bf001c9713c2e3bcc6de79726
Author: Namjae Jeon <linkinjeon@kernel.org>
Date:   Wed May 3 14:03:40 2023 +0900

    ksmbd: fix racy issue from smb2 close and logoff with multichannel

    commit abcc506a9a71976a8b4c9bf3ee6efd13229c1e19 upstream.

    When smb client send concurrent smb2 close and logoff request
    with multichannel connection, It can cause racy issue. logoff request
    free tcon and can cause UAF issues in smb2 close. When receiving logoff
    request with multichannel, ksmbd should wait until all remaning requests
    complete as well as ones in the current connection, and then make
    session expired.

    Cc: stable@vger.kernel.org
    Reported-by: zdi-disclosures@trendmicro.com # ZDI-CAN-20796 ZDI-CAN-20595
    Signed-off-by: Namjae Jeon <linkinjeon@kernel.org>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 6775ee7ef4b37c521aa4cf3730f54554c4875542
Author: Namjae Jeon <linkinjeon@kernel.org>
Date:   Wed May 3 08:42:21 2023 +0900

    ksmbd: destroy expired sessions

    commit ea174a91893956450510945a0c5d1a10b5323656 upstream.

    client can indefinitely send smb2 session setup requests with
    the SessionId set to 0, thus indefinitely spawning new sessions,
    and causing indefinite memory usage. This patch limit to the number
    of sessions using expired timeout and session state.
```

```
    Cc: stable@vger.kernel.org
    Reported-by: zdi-disclosures@trendmicro.com # ZDI-CAN-20478
    Signed-off-by: Namjae Jeon <linkinjeon@kernel.org>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a1fdf5e0ef5c65fcaa2dd400890d35725aa87248
Author: Namjae Jeon <linkinjeon@kernel.org>
Date:   Wed May 3 08:43:30 2023 +0900

    ksmbd: block asynchronous requests when making a delay on session setup

    commit b096d97f47326b1e2dbdef1c91fab69ffda54d17 upstream.

    ksmbd make a delay of 5 seconds on session setup to avoid dictionary
    attacks. But the 5 seconds delay can be bypassed by using asynchronous
    requests. This patch block all requests on current connection when
    making a delay on sesstion setup failure.

    Cc: stable@vger.kernel.org
    Reported-by: zdi-disclosures@trendmicro.com # ZDI-CAN-20482
    Signed-off-by: Namjae Jeon <linkinjeon@kernel.org>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 02f41d88f15d6b7d523e52cc3f87488f57e9265b
Author: Namjae Jeon <linkinjeon@kernel.org>
Date:   Wed May 3 16:45:00 2023 +0900

    ksmbd: fix racy issue from session setup and logoff

    commit f5c779b7ddbda30866cf2a27c63e34158f858c73 upstream.

    This racy issue is triggered by sending concurrent session setup and
    logoff requests. This patch does not set connection status as
    KSMBD_SESS_GOOD if state is KSMBD_SESS_NEED_RECONNECT in session setup.
    And relookup session to validate if session is deleted in logoff.

    Cc: stable@vger.kernel.org
    Reported-by: zdi-disclosures@trendmicro.com # ZDI-CAN-20481, ZDI-CAN-20590, ZDI-CAN-
20596
    Signed-off-by: Namjae Jeon <linkinjeon@kernel.org>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 710405942b31d92cd871f95c36f1bb9c1b83a8f0
Author: Namjae Jeon <linkinjeon@kernel.org>
Date:   Wed May 3 08:44:14 2023 +0900

    ksmbd: fix deadlock in ksmbd_find_crypto_ctx()

    commit 7b4323373d844954bb76e0e9f39c4e5fc785fa7b upstream.

    Deadlock is triggered by sending multiple concurrent session setup
    requests. It should be reused after releasing when getting ctx for crypto.
    Multiple consecutive ctx uses cause deadlock while waiting for releasing
    due to the limited number of ctx.

    Cc: stable@vger.kernel.org
    Reported-by: zdi-disclosures@trendmicro.com # ZDI-CAN-20591
    Signed-off-by: Namjae Jeon <linkinjeon@kernel.org>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit 088131b7b01099720a528a72005ff17868705d40
Author: Namjae Jeon <linkinjeon@kernel.org>
Date:   Wed May 3 08:45:08 2023 +0900

    ksmbd: not allow guest user on multichannel

    commit 3353ab2df5f68dab7da8d5ebb427a2d265a1f2b2 upstream.

    This patch return STATUS_NOT_SUPPORTED if binding session is guest.

    Cc: stable@vger.kernel.org
    Reported-by: zdi-disclosures@trendmicro.com # ZDI-CAN-20480
    Signed-off-by: Namjae Jeon <linkinjeon@kernel.org>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 608c8d0b49908ba7edadf0df056b42f98ac05ee2
Author: Namjae Jeon <linkinjeon@kernel.org>
Date:   Wed May 3 08:26:45 2023 +0900

    ksmbd: fix memleak in session setup

    commit 6d7cb549c2ca20e1f07593f15e936fd54b763028 upstream.

    If client send session setup request with unknown NTLMSSP message type,
    session that does not included channel can be created. It will cause
    session memleak. because ksmbd_sessions_deregister() does not destroy
    session if channel is not included. This patch return error response if
    client send the request unknown NTLMSSP message type.

    Cc: stable@vger.kernel.org
    Reported-by: zdi-disclosures@trendmicro.com # ZDI-CAN-20593
    Signed-off-by: Namjae Jeon <linkinjeon@kernel.org>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1636e09779f83e10e6ed57d91ef94abcefdd206b
Author: Namjae Jeon <linkinjeon@kernel.org>
Date:   Wed May 3 08:38:33 2023 +0900

    ksmbd: fix NULL pointer dereference in smb2_get_info_filesystem()

    commit 3ac00a2ab69b34189942afa9e862d5170cdcb018 upstream.

    If share is , share->path is NULL and it cause NULL pointer
    dereference issue.

    Cc: stable@vger.kernel.org
    Reported-by: zdi-disclosures@trendmicro.com # ZDI-CAN-20479
    Signed-off-by: Namjae Jeon <linkinjeon@kernel.org>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 5a7090ccc242ab009ee7769e9d7fad6644dbe9bd
Author: Namjae Jeon <linkinjeon@kernel.org>
Date:   Wed May 3 08:51:51 2023 +0900

    ksmbd: call rcu_barrier() in ksmbd_server_exit()

    commit eb307d09fe15844fdaebeb8cc8c9b9e925430aa5 upstream.

    racy issue is triggered the bug by racing between closing a connection
    and rmmod. In ksmbd, rcu_barrier() is not called at module unload time,
    so nothing prevents ksmbd from getting unloaded while it still has RCU
    callbacks pending. It leads to trigger unintended execution of kernel
```

```
    code locally and use to defeat protections such as Kernel Lockdown

    Cc: stable@vger.kernel.org
    Reported-by: zdi-disclosures@trendmicro.com # ZDI-CAN-20477
    Signed-off-by: Namjae Jeon <linkinjeon@kernel.org>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 39366b47a59d46af15ac57beb0996268bf911f6a
Author: Namjae Jeon <linkinjeon@kernel.org>
Date:   Wed May 3 08:51:12 2023 +0900

    ksmbd: fix racy issue under cocurrent smb2 tree disconnect

    commit 30210947a343b6b3ca13adc9bfc88e1543e16dd5 upstream.

    There is UAF issue under cocurrent smb2 tree disconnect.
    This patch introduce TREE_CONN_EXPIRE flags for tcon to avoid cocurrent
    access.

    Cc: stable@vger.kernel.org
    Reported-by: zdi-disclosures@trendmicro.com # ZDI-CAN-20592
    Signed-off-by: Namjae Jeon <linkinjeon@kernel.org>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 2ec1fe292d6edb3bd112f900692d9ef292b1fa8b
Author: Sean Christopherson <seanjc@google.com>
Date:   Wed Apr 26 15:03:23 2023 -0700

    KVM: x86: Preserve TDP MMU roots until they are explicitly invalidated

    commit edbdb43fc96b11b3bfa531be306a1993d9fe89ec upstream.

    Preserve TDP MMU roots until they are explicitly invalidated by gifting
    the TDP MMU itself a reference to a root when it is allocated.  Keeping a
    reference in the TDP MMU fixes a flaw where the TDP MMU exhibits terrible
    performance, and can potentially even soft-hang a vCPU, if a vCPU
    frequently unloads its roots, e.g. when KVM is emulating SMI+RSM.

    When KVM emulates something that invalidates _all_ TLB entries, e.g. SMI
    and RSM, KVM unloads all of the vCPUs roots (KVM keeps a small per-vCPU
    cache of previous roots).  Unloading roots is a simple way to ensure KVM
    flushes and synchronizes all roots for the vCPU, as KVM flushes and syncs
    when allocating a "new" root (from the vCPU's perspective).

    In the shadow MMU, KVM keeps track of all shadow pages, roots included, in
    a per-VM hash table.  Unloading a shadow MMU root just wipes it from the
    per-vCPU cache; the root is still tracked in the per-VM hash table.  When
    KVM loads a "new" root for the vCPU, KVM will find the old, unloaded root
    in the per-VM hash table.

    Unlike the shadow MMU, the TDP MMU doesn't track "inactive" roots in a
    per-VM structure, where "active" in this case means a root is either
    in-use or cached as a previous root by at least one vCPU.  When a TDP MMU
    root becomes inactive, i.e. the last vCPU reference to the root is put,
    KVM immediately frees the root (asterisk on "immediately" as the actual
    freeing may be done by a worker, but for all intents and purposes the root
    is gone).

    The TDP MMU behavior is especially problematic for 1-vCPU setups, as
    unloading all roots effectively frees all roots.  The issue is mitigated
    to some degree in multi-vCPU setups as a different vCPU usually holds a
    reference to an unloaded root and thus keeps the root alive, allowing the
    vCPU to reuse its old root after unloading (with a flush+sync).
```

```
    The TDP MMU flaw has been known for some time, as until very recently,
    KVM's handling of CR0.WP also triggered unloading of all roots.  The
    CR0.WP toggling scenario was eventually addressed by not unloading roots
    when _only_ CR0.WP is toggled, but such an approach doesn't Just Work
    for emulating SMM as KVM must emulate a full TLB flush on entry and exit
    to/from SMM.  Given that the shadow MMU plays nice with unloading roots
    at will, teaching the TDP MMU to do the same is far less complex than
    modifying KVM to track which roots need to be flushed before reuse.

    Note, preserving all possible TDP MMU roots is not a concern with respect
    to memory consumption.  Now that the role for direct MMUs doesn't include
    information about the guest, e.g. CR0.PG, CR0.WP, CR4.SMEP, etc., there
    are _at most_ six possible roots (where "guest_mode" here means L2):

      1. 4-level !SMM !guest_mode
      2. 4-level  SMM !guest_mode
      3. 5-level !SMM !guest_mode
      4. 5-level  SMM !guest_mode
      5. 4-level !SMM guest_mode
      6. 5-level !SMM guest_mode

    And because each vCPU can track 4 valid roots, a VM can already have all
    6 root combinations live at any given time.  Not to mention that, in
    practice, no sane VMM will advertise different guest.MAXPHYADDR values
    across vCPUs, i.e. KVM won't ever use both 4-level and 5-level roots for
    a single VM.  Furthermore, the vast majority of modern hypervisors will
    utilize EPT/NPT when available, thus the guest_mode=%true cases are also
    unlikely to be utilized.

    Reported-by: Jeremi Piotrowski <jpiotrowski@linux.microsoft.com>
    Link: https://lore.kernel.org/all/959c5bce-beb5-b463-7158-
33fc4a4f910c@linux.microsoft.com
    Link: https://lkml.kernel.org/r/20220209170020.1775368-1-pbonzini%40redhat.com
    Link: https://lore.kernel.org/all/20230322013731.102955-1-minipli@grsecurity.net
    Link: https://lore.kernel.org/all/000000000000a0bc2b05f9dd7fab@google.com
    Link: https://lore.kernel.org/all/000000000000eca0b905fa0f7756@google.com
    Cc: Ben Gardon <bgardon@google.com>
    Cc: David Matlack <dmatlack@google.com>
    Cc: stable@vger.kernel.org
    Tested-by: Jeremi Piotrowski <jpiotrowski@linux.microsoft.com>
    Link: https://lore.kernel.org/r/20230426220323.3079789-1-seanjc@google.com
    Signed-off-by: Sean Christopherson <seanjc@google.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 842bf6b61d57b4b462844911de9afa94bc99389a
Author: David Matlack <dmatlack@google.com>
Date:   Fri Mar 17 14:11:06 2023 -0700

    KVM: RISC-V: Retry fault if vma_lookup() results become invalid

    commit 2ed90cb0938a45b12eb947af062d12c7af0067b3 upstream.

    Read mmu_invalidate_seq before dropping the mmap_lock so that KVM can
    detect if the results of vma_lookup() (e.g. vma_shift) become stale
    before it acquires kvm->mmu_lock. This fixes a theoretical bug where a
    VMA could be changed by userspace after vma_lookup() and before KVM
    reads the mmu_invalidate_seq, causing KVM to install page table entries
    based on a (possibly) no-longer-valid vma_shift.

    Re-order the MMU cache top-up to earlier in user_mem_abort() so that it
    is not done after KVM has read mmu_invalidate_seq (i.e. so as to avoid
    inducing spurious fault retries).

    It's unlikely that any sane userspace currently modifies VMAs in such a
```

```
    way as to trigger this race. And even with directed testing I was unable
    to reproduce it. But a sufficiently motivated host userspace might be
    able to exploit this race.

    Note KVM/ARM had the same bug and was fixed in a separate, near
    identical patch (see Link).

    Link: https://lore.kernel.org/kvm/20230313235454.2964067-1-dmatlack@google.com/
    Fixes: 9955371cc014 ("RISC-V: KVM: Implement MMU notifiers")
    Cc: stable@vger.kernel.org
    Signed-off-by: David Matlack <dmatlack@google.com>
    Tested-by: Anup Patel <anup@brainfault.org>
    Signed-off-by: Anup Patel <anup@brainfault.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 35ca4fb494c0c9f226fbcfa1c1688e6cc1e5062e
Author: Zhang Zhengming <zhang.zhengming@h3c.com>
Date:   Wed Apr 19 12:02:03 2023 +0800

    relayfs: fix out-of-bounds access in relay_file_read

    commit 43ec16f1450f4936025a9bdf1a273affdb9732c1 upstream.

    There is a crash in relay_file_read, as the var from
    point to the end of last subbuf.

    The oops looks something like:
    pc : __arch_copy_to_user+0x180/0x310
    lr : relay_file_read+0x20c/0x2c8
    Call trace:
     __arch_copy_to_user+0x180/0x310
     full_proxy_read+0x68/0x98
     vfs_read+0xb0/0x1d0
     ksys_read+0x6c/0xf0
     __arm64_sys_read+0x20/0x28
     el0_svc_common.constprop.3+0x84/0x108
     do_el0_svc+0x74/0x90
     el0_svc+0x1c/0x28
     el0_sync_handler+0x88/0xb0
     el0_sync+0x148/0x180

    We get the condition by analyzing the vmcore:

    1). The last produced byte and last consumed byte
        both at the end of the last subbuf

    2). A softirq calls function(e.g __blk_add_trace)
        to write relay buffer occurs when an program is calling
        relay_file_read_avail().

            relay_file_read
                    relay_file_read_avail
                            relay_file_read_consume(buf, 0, 0);
                            //interrupted by softirq who will write subbuf
                            ....
                            return 1;
                    //read_start point to the end of the last subbuf
                    read_start = relay_file_read_start_pos
                    //avail is equal to subsize
                    avail = relay_file_read_subbuf_avail
                    //from  points to an invalid memory address
                    from = buf->start + read_start
                    //system is crashed
                    copy_to_user(buffer, from, avail)
```

```
    Link: https://lkml.kernel.org/r/20230419040203.37676-1-zhang.zhengming@h3c.com
    Fixes: 8d62fdebdaf9 ("relay file read: start-pos fix")
    Signed-off-by: Zhang Zhengming <zhang.zhengming@h3c.com>
    Reviewed-by: Zhao Lei <zhao_lei1@hoperun.com>
    Reviewed-by: Zhou Kete <zhou.kete@h3c.com>
    Reviewed-by: Pengcheng Yang <yangpc@wangsu.com>
    Cc: Jens Axboe <axboe@kernel.dk>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1ecb1a4ecc183f4f33d7de84adf448f8a1992cac
Author: Oliver Upton <oliver.upton@linux.dev>
Date:   Wed Apr 12 06:27:33 2023 +0000

    KVM: arm64: vgic: Don't acquire its_lock before config_lock

    commit 49e5d16b6fc003407a33a9961b4bcbb970bd1c76 upstream.

    commit f00327731131 ("KVM: arm64: Use config_lock to protect vgic
    state") was meant to rectify a longstanding lock ordering issue in KVM
    where the kvm->lock is taken while holding vcpu->mutex. As it so
    happens, the aforementioned commit introduced yet another locking issue
    by acquiring the its_lock before acquiring the config lock.

    This is obviously wrong, especially considering that the lock ordering
    is well documented in vgic.c. Reshuffle the locks once more to take the
    config_lock before the its_lock. While at it, sprinkle in the lockdep
    hinting that has become popular as of late to keep lockdep apprised of
    our ordering.

    Cc: stable@vger.kernel.org
    Fixes: f00327731131 ("KVM: arm64: Use config_lock to protect vgic state")
    Signed-off-by: Oliver Upton <oliver.upton@linux.dev>
    Signed-off-by: Marc Zyngier <maz@kernel.org>
    Link: https://lore.kernel.org/r/20230412062733.988229-1-oliver.upton@linux.dev
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 9c8a5f647c863c2529a1893e82daed32e4380e7a
Author: Oliver Upton <oliver.upton@linux.dev>
Date:   Mon Mar 27 16:47:47 2023 +0000

    KVM: arm64: Use config_lock to protect vgic state

    commit f00327731131d1b5aa6a1aa9f50bcf8d620ace4c upstream.

    Almost all of the vgic state is VM-scoped but accessed from the context
    of a vCPU. These accesses were serialized on the kvm->lock which cannot
    be nested within a vcpu->mutex critical section.

    Move over the vgic state to using the config_lock. Tweak the lock
    ordering where necessary to ensure that the config_lock is acquired
    after the vcpu->mutex. Acquire the config_lock in kvm_vgic_create() to
    avoid a race between the converted flows and GIC creation. Where
    necessary, continue to acquire kvm->lock to avoid a race with vCPU
    creation (i.e. flows that use lock_all_vcpus()).

    Finally, promote the locking expectations in comments to lockdep
    assertions and update the locking documentation for the config_lock as
    well as vcpu->mutex.

    Cc: stable@vger.kernel.org
    Signed-off-by: Oliver Upton <oliver.upton@linux.dev>
    Signed-off-by: Marc Zyngier <maz@kernel.org>
    Link: https://lore.kernel.org/r/20230327164747.2466958-5-oliver.upton@linux.dev
```

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

commit 633facd0368087a08937170018ce76d58344f8df
Author: Oliver Upton <oliver.upton@linux.dev>
Date:   Mon Mar 27 16:47:46 2023 +0000

    KVM: arm64: Use config_lock to protect data ordered against KVM_RUN

    commit 4bba7f7def6f278266dadf845da472cfbfed784e upstream.

    There are various bits of VM-scoped data that can only be configured
    before the first call to KVM_RUN, such as the hypercall bitmaps and
    the PMU. As these fields are protected by the kvm->lock and accessed
    while holding vcpu->mutex, this is yet another example of lock
    inversion.

    Change out the kvm->lock for kvm->arch.config_lock in all of these
    instances. Opportunistically simplify the locking mechanics of the
    PMU configuration by holding the config_lock for the entirety of
    kvm_arm_pmu_v3_set_attr().

    Note that this also addresses a couple of bugs. There is an unguarded
    read of the PMU version in KVM_ARM_VCPU_PMU_V3_FILTER which could race
    with KVM_ARM_VCPU_PMU_V3_SET_PMU. Additionally, until now writes to the
    per-vCPU vPMU irq were not serialized VM-wide, meaning concurrent calls
    to KVM_ARM_VCPU_PMU_V3_IRQ could lead to a false positive in
    pmu_irq_is_valid().

    Cc: stable@vger.kernel.org
    Tested-by: Jeremy Linton <jeremy.linton@arm.com>
    Signed-off-by: Oliver Upton <oliver.upton@linux.dev>
    Signed-off-by: Marc Zyngier <maz@kernel.org>
    Link: https://lore.kernel.org/r/20230327164747.2466958-4-oliver.upton@linux.dev
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a480f28199046afe27dd1c68e6c6cdd8742f1822
Author: Oliver Upton <oliver.upton@linux.dev>
Date:   Mon Mar 27 16:47:45 2023 +0000

    KVM: arm64: Avoid lock inversion when setting the VM register width

    commit c43120afb5c66a3465c7468f5cf9806a26484cde upstream.

    kvm->lock must be taken outside of the vcpu->mutex. Of course, the
    locking documentation for KVM makes this abundantly clear. Nonetheless,
    the locking order in KVM/arm64 has been wrong for quite a while; we
    acquire the kvm->lock while holding the vcpu->mutex all over the shop.

    All was seemingly fine until commit 42a90008f890 ("KVM: Ensure lockdep
    knows about kvm->lock vs. vcpu->mutex ordering rule") caught us with our
    pants down, leading to lockdep barfing:

     ======================================================
     WARNING: possible circular locking dependency detected
     6.2.0-rc7+ #19 Not tainted
     ------------------------------------------------------
     qemu-system-aar/859 is trying to acquire lock:
     ffff5aa69269eba0 (&host_kvm->lock){+.+.}-{3:3}, at: kvm_reset_vcpu+0x34/0x274

     but task is already holding lock:
     ffff5aa68768c0b8 (&vcpu->mutex){+.+.}-{3:3}, at: kvm_vcpu_ioctl+0x8c/0xba0

     which lock already depends on the new lock.

    Add a dedicated lock to serialize writes to VM-scoped configuration from
```

```
    the context of a vCPU. Protect the register width flags with the new
    lock, thus avoiding the need to grab the kvm->lock while holding
    vcpu->mutex in kvm_reset_vcpu().

    Cc: stable@vger.kernel.org
    Reported-by: Jeremy Linton <jeremy.linton@arm.com>
    Link: https://lore.kernel.org/kvmarm/f6452cdd-65ff-34b8-bab0-5c06416da5f6@arm.com/
    Tested-by: Jeremy Linton <jeremy.linton@arm.com>
    Signed-off-by: Oliver Upton <oliver.upton@linux.dev>
    Signed-off-by: Marc Zyngier <maz@kernel.org>
    Link: https://lore.kernel.org/r/20230327164747.2466958-3-oliver.upton@linux.dev
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3dee8e34ae87ab494a7f73feb222372bd80126d3
Author: Oliver Upton <oliver.upton@linux.dev>
Date:   Mon Mar 27 16:47:44 2023 +0000

    KVM: arm64: Avoid vcpu->mutex v. kvm->lock inversion in CPU_ON

    commit 0acc7239c20a8401b8968c2adace8f7c9b0295ae upstream.

    KVM/arm64 had the lock ordering backwards on vcpu->mutex and kvm->lock
    from the very beginning. One such example is the way vCPU resets are
    handled: the kvm->lock is acquired while handling a guest CPU_ON PSCI
    call.

    Add a dedicated lock to serialize writes to kvm_vcpu_arch::{mp_state,
    reset_state}. Promote all accessors of mp_state to {READ,WRITE}_ONCE()
    as readers do not acquire the mp_state_lock. While at it, plug yet
    another race by taking the mp_state_lock in the KVM_SET_MP_STATE ioctl
    handler.

    As changes to MP state are now guarded with a dedicated lock, drop the
    kvm->lock acquisition from the PSCI CPU_ON path. Similarly, move the
    reader of reset_state outside of the kvm->lock and instead protect it
    with the mp_state_lock. Note that writes to reset_state::reset have been
    demoted to regular stores as both readers and writers acquire the
    mp_state_lock.

    While the kvm->lock inversion still exists in kvm_reset_vcpu(), at least
    now PSCI CPU_ON no longer depends on it for serializing vCPU reset.

    Cc: stable@vger.kernel.org
    Tested-by: Jeremy Linton <jeremy.linton@arm.com>
    Signed-off-by: Oliver Upton <oliver.upton@linux.dev>
    Signed-off-by: Marc Zyngier <maz@kernel.org>
    Link: https://lore.kernel.org/r/20230327164747.2466958-2-oliver.upton@linux.dev
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit fed618e7cf8a7a6d90d69cc23ba2aaa196761985
Author: Sean Christopherson <seanjc@google.com>
Date:   Tue Apr 4 17:23:59 2023 -0700

    KVM: nVMX: Emulate NOPs in L2, and PAUSE if it's not intercepted

    commit 4984563823f0034d3533854c1b50e729f5191089 upstream.

    Extend VMX's nested intercept logic for emulated instructions to handle
    "pause" interception, in quotes because KVM's emulator doesn't filter out
    NOPs when checking for nested intercepts.  Failure to allow emulation of
    NOPs results in KVM injecting a #UD into L2 on any NOP that collides with
    the emulator's definition of PAUSE, i.e. on all single-byte NOPs.

    For PAUSE itself, honor L1's PAUSE-exiting control, but ignore PLE to
    avoid unnecessarily injecting a #UD into L2.  Per the SDM, the first
```

```
    execution of PAUSE after VM-Entry is treated as the beginning of a new
    loop, i.e. will never trigger a PLE VM-Exit, and so L1 can't expect any
    given execution of PAUSE to deterministically exit.

      ... the processor considers this execution to be the first execution of
      PAUSE in a loop. (It also does so for the first execution of PAUSE at
      CPL 0 after VM entry.)

    All that said, the PLE side of things is currently a moot point, as KVM
    doesn't expose PLE to L1.

    Note, vmx_check_intercept() is still wildly broken when L1 wants to
    intercept an instruction, as KVM injects a #UD instead of synthesizing a
    nested VM-Exit.  That issue extends far beyond NOP/PAUSE and needs far
    more effort to fix, i.e. is a problem for the future.

    Fixes: 07721feee46b ("KVM: nVMX: Don't emulate instructions in guest mode")
    Cc: Mathias Krause <minipli@grsecurity.net>
    Cc: stable@vger.kernel.org
    Reviewed-by: Paolo Bonzini <pbonzini@redhat.com>
    Link: https://lore.kernel.org/r/20230405002359.418138-1-seanjc@google.com
    Signed-off-by: Sean Christopherson <seanjc@google.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 97fe96c75e8970405bf143f384834187de75ae59
Author: Sean Christopherson <seanjc@google.com>
Date:   Sat Jan 28 00:14:27 2023 +0000

    KVM: x86/pmu: Disallow legacy LBRs if architectural LBRs are available

    commit 098f4c061ea10b777033b71c10bd9fd706820ee9 upstream.

    Disallow enabling LBR support if the CPU supports architectural LBRs.
    Traditional LBR support is absent on CPU models that have architectural
    LBRs, and KVM doesn't yet support arch LBRs, i.e. KVM will pass through
    non-existent MSRs if userspace enables LBRs for the guest.

    Cc: stable@vger.kernel.org
    Cc: Yang Weijiang <weijiang.yang@intel.com>
    Cc: Like Xu <like.xu.linux@gmail.com>
    Reported-by: Paolo Bonzini <pbonzini@redhat.com>
    Fixes: be635e34c284 ("KVM: vmx/pmu: Expose LBR_FMT in the MSR_IA32_PERF_CAPABILITIES")
    Tested-by: Like Xu <likexu@tencent.com>
    Link: https://lore.kernel.org/r/20230128001427.2548858-1-seanjc@google.com
    Signed-off-by: Sean Christopherson <seanjc@google.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit d9bad836cf156ee87d577f0bd1ed01501b31a253
Author: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
Date:   Thu Mar 23 18:40:26 2023 +0100

    cpufreq: qcom-cpufreq-hw: fix double IO unmap and resource release on exit

    commit ba5e770c9698782bc203bbf5cf3b36a77720bdbe upstream.

    Commit 054a3ef683a1 ("cpufreq: qcom-hw: Allocate qcom_cpufreq_data
    during probe") moved getting memory resource and iomap from
    qcom_cpufreq_hw_cpu_init() to the probe function, however it left
    untouched cleanup in qcom_cpufreq_hw_cpu_exit().

    During device unbind this will lead to doule release of resource and
    double iounmap(), first by qcom_cpufreq_hw_cpu_exit() and second via
    managed resources:

      resource: Trying to free nonexistent resource <0x0000000018593000-0x0000000018593fff>
```

```
       Trying to vunmap() nonexistent vm area (0000000088a7d4dc)
       ...
       vunmap (mm/vmalloc.c:2771 (discriminator 1))
       iounmap (mm/ioremap.c:60)
       devm_ioremap_release (lib/devres.c:19)
       devres_release_all (drivers/base/devres.c:506 drivers/base/devres.c:535)
       device_unbind_cleanup (drivers/base/dd.c:523)
       device_release_driver_internal (drivers/base/dd.c:1248 drivers/base/dd.c:1263)
       device_driver_detach (drivers/base/dd.c:1300)
       unbind_store (drivers/base/bus.c:243)
       drv_attr_store (drivers/base/bus.c:127)
       sysfs_kf_write (fs/sysfs/file.c:137)
       kernfs_fop_write_iter (fs/kernfs/file.c:334)
       vfs_write (include/linux/fs.h:1851 fs/read_write.c:491 fs/read_write.c:584)
       ksys_write (fs/read_write.c:637)
       __arm64_sys_write (fs/read_write.c:646)
       invoke_syscall (arch/arm64/include/asm/current.h:19 arch/arm64/kernel/syscall.c:57)
       el0_svc_common.constprop.0 (arch/arm64/include/asm/daifflags.h:28
 arch/arm64/kernel/syscall.c:150)
       do_el0_svc (arch/arm64/kernel/syscall.c:194)
       el0_svc (arch/arm64/include/asm/daifflags.h:28 arch/arm64/kernel/entry-common.c:133
 arch/arm64/kernel/entry-common.c:142 arch/arm64/kernel/entry-common.c:638)
       el0t_64_sync_handler (arch/arm64/kernel/entry-common.c:656)
       el0t_64_sync (arch/arm64/kernel/entry.S:591)

    Fixes: 054a3ef683a1 ("cpufreq: qcom-hw: Allocate qcom_cpufreq_data during probe")
    Cc: <stable@vger.kernel.org>
    Cc: Manivannan Sadhasivam <mani@kernel.org>
    Signed-off-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Reviewed-by: Manivannan Sadhasivam <mani@kernel.org>
    Reviewed-by: Bjorn Andersson <andersson@kernel.org>
    Signed-off-by: Viresh Kumar <viresh.kumar@linaro.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

 commit e6f1fbd33fa308cab0164075673dabe7e20380c6
 Author: Roberto Sassu <roberto.sassu@huawei.com>
 Date:   Fri Mar 31 14:32:18 2023 +0200

    reiserfs: Add security prefix to xattr name in reiserfs_security_write()

    commit d82dcd9e21b77d338dc4875f3d4111f0db314a7c upstream.

    Reiserfs sets a security xattr at inode creation time in two stages: first,
    it calls reiserfs_security_init() to obtain the xattr from active LSMs;
    then, it calls reiserfs_security_write() to actually write that xattr.

    Unfortunately, it seems there is a wrong expectation that LSMs provide the
    full xattr name in the form 'security.<suffix>'. However, LSMs always
    provided just the suffix, causing reiserfs to not write the xattr at all
    (if the suffix is shorter than the prefix), or to write an xattr with the
    wrong name.

    Add a temporary buffer in reiserfs_security_write(), and write to it the
    full xattr name, before passing it to reiserfs_xattr_set_handle().

    Also replace the name length check with a check that the full xattr name is
    not larger than XATTR_NAME_MAX.

    Cc: stable@vger.kernel.org # v2.6.x
    Fixes: 57fe60df6241 ("reiserfs: add atomic addition of selinux attributes during inode
 creation")
    Signed-off-by: Roberto Sassu <roberto.sassu@huawei.com>
    Signed-off-by: Paul Moore <paul@paul-moore.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit 7b5a97333e920b69356e097f185bdc51d61e66ee
Author: Zheng Yejian <zhengyejian1@huawei.com>
Date:   Fri Jan 6 15:09:34 2023 +0800

    rcu: Avoid stack overflow due to __rcu_irq_enter_check_tick() being kprobe-ed

    commit 7a29fb4a4771124bc61de397dbfc1554dbbcc19c upstream.

    Registering a kprobe on __rcu_irq_enter_check_tick() can cause kernel
    stack overflow as shown below. This issue can be reproduced by enabling
    CONFIG_NO_HZ_FULL and booting the kernel with argument "nohz_full=",
    and then giving the following commands at the shell prompt:

      # cd /sys/kernel/tracing/
      # echo 'p:mp1 __rcu_irq_enter_check_tick' >> kprobe_events
      # echo 1 > events/kprobes/enable

    This commit therefore adds __rcu_irq_enter_check_tick() to the kprobes
    blacklist using NOKPROBE_SYMBOL().

    Insufficient stack space to handle exception!
    ESR: 0x00000000f2000004 -- BRK (AArch64)
    FAR: 0x0000ffffccf3e510
    Task stack:     [0xffff80000ad30000..0xffff80000ad38000]
    IRQ stack:      [0xffff800008050000..0xffff800008058000]
    Overflow stack: [0xffff089c36f9f310..0xffff089c36fa0310]
    CPU: 5 PID: 190 Comm: bash Not tainted 6.2.0-rc2-00320-g1f5abbd77e2c #19
    Hardware name: linux,dummy-virt (DT)
    pstate: 400003c5 (nZcv DAIF -PAN -UAO -TCO -DIT -SSBS BTYPE=--)
    pc : __rcu_irq_enter_check_tick+0x0/0x1b8
    lr : ct_nmi_enter+0x11c/0x138
    sp : ffff80000ad30080
    x29: ffff80000ad30080 x28: ffff089c82e20000 x27: 0000000000000000
    x26: 0000000000000000 x25: ffff089c02a8d100 x24: 0000000000000000
    x23: 00000000400003c5 x22: 0000ffffccf3e510 x21: ffff089c36fae148
    x20: ffff80000ad30120 x19: ffffa8da8fcce148 x18: 0000000000000000
    x17: 0000000000000000 x16: 0000000000000000 x15: ffffa8da8e44ea6c
    x14: ffffa8da8e44e968 x13: ffffa8da8e03136c x12: 1fffe113804d6809
    x11: ffff6113804d6809 x10: 0000000000000a60 x9 : dfff800000000000
    x8 : ffff089c026b404f x7 : 00009eec7fb297f7 x6 : 0000000000000001
    x5 : ffff80000ad30120 x4 : dfff800000000000 x3 : ffffa8da8e3016f4
    x2 : 0000000000000003 x1 : 0000000000000000 x0 : 0000000000000000
    Kernel panic - not syncing: kernel stack overflow
    CPU: 5 PID: 190 Comm: bash Not tainted 6.2.0-rc2-00320-g1f5abbd77e2c #19
    Hardware name: linux,dummy-virt (DT)
    Call trace:
     dump_backtrace+0xf8/0x108
     show_stack+0x20/0x30
     dump_stack_lvl+0x68/0x84
     dump_stack+0x1c/0x38
     panic+0x214/0x404
     add_taint+0x0/0xf8
     panic_bad_stack+0x144/0x160
     handle_bad_stack+0x38/0x58
     __bad_stack+0x78/0x7c
     __rcu_irq_enter_check_tick+0x0/0x1b8
     arm64_enter_el1_dbg.isra.0+0x14/0x20
     el1_dbg+0x2c/0x90
     el1h_64_sync_handler+0xcc/0xe8
     el1h_64_sync+0x64/0x68
     __rcu_irq_enter_check_tick+0x0/0x1b8
     arm64_enter_el1_dbg.isra.0+0x14/0x20
     el1_dbg+0x2c/0x90
     el1h_64_sync_handler+0xcc/0xe8
     el1h_64_sync+0x64/0x68
```

```
      __rcu_irq_enter_check_tick+0x0/0x1b8
      arm64_enter_el1_dbg.isra.0+0x14/0x20
      el1_dbg+0x2c/0x90
      el1h_64_sync_handler+0xcc/0xe8
      el1h_64_sync+0x64/0x68
      __rcu_irq_enter_check_tick+0x0/0x1b8
      [...]
      el1_dbg+0x2c/0x90
      el1h_64_sync_handler+0xcc/0xe8
      el1h_64_sync+0x64/0x68
      __rcu_irq_enter_check_tick+0x0/0x1b8
      arm64_enter_el1_dbg.isra.0+0x14/0x20
      el1_dbg+0x2c/0x90
      el1h_64_sync_handler+0xcc/0xe8
      el1h_64_sync+0x64/0x68
      __rcu_irq_enter_check_tick+0x0/0x1b8
      arm64_enter_el1_dbg.isra.0+0x14/0x20
      el1_dbg+0x2c/0x90
      el1h_64_sync_handler+0xcc/0xe8
      el1h_64_sync+0x64/0x68
      __rcu_irq_enter_check_tick+0x0/0x1b8
      el1_interrupt+0x28/0x60
      el1h_64_irq_handler+0x18/0x28
      el1h_64_irq+0x64/0x68
      __ftrace_set_clr_event_nolock+0x98/0x198
      __ftrace_set_clr_event+0x58/0x80
      system_enable_write+0x144/0x178
      vfs_write+0x174/0x738
      ksys_write+0xd0/0x188
      __arm64_sys_write+0x4c/0x60
      invoke_syscall+0x64/0x180
      el0_svc_common.constprop.0+0x84/0x160
      do_el0_svc+0x48/0xe8
      el0_svc+0x34/0xd0
      el0t_64_sync_handler+0xb8/0xc0
      el0t_64_sync+0x190/0x194
    SMP: stopping secondary CPUs
    Kernel Offset: 0x28da86000000 from 0xffff800008000000
    PHYS_OFFSET: 0xfffff76600000000
    CPU features: 0x00000,01a00100,0000421b
    Memory Limit: none

    Acked-by: Joel Fernandes (Google) <joel@joelfernandes.org>
    Link: https://lore.kernel.org/all/20221119040049.795065-1-zhengyejian1@huawei.com/
    Fixes: aaf2bc50df1f ("rcu: Abstract out rcu_irq_enter_check_tick() from
rcu_nmi_enter()")
    Signed-off-by: Zheng Yejian <zhengyejian1@huawei.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Paul E. McKenney <paulmck@kernel.org>
    Signed-off-by: Joel Fernandes (Google) <joel@joelfernandes.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a343d7d3d8fb8650743d3659e9e368c938bc29d6
Author: Mario Limonciello <mario.limonciello@amd.com>
Date:   Mon Apr 3 12:38:01 2023 -0500

    crypto: ccp - Don't initialize CCP for PSP 0x1649

    commit c79a3169b9f3633c215b55857eba5921e5b49217 upstream.

    A number of platforms are emitting the error:
    ```ccp: unable to access the device: you might be running a broken BIOS.```

    This is expected behavior as CCP is no longer accessible from the PSP's
    PCIe BAR so stop trying to probe CCP for 0x1649.
```
```

```
    Cc: stable@vger.kernel.org
    Signed-off-by: Mario Limonciello <mario.limonciello@amd.com>
    Acked-by: Tom Lendacky <thomas.lendacky@amd.com>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ce672a4447df8dbeb203c72a0d3b6aa82cea31e5
Author: Eric Biggers <ebiggers@google.com>
Date:   Mon Feb 27 10:29:47 2023 -0800

    crypto: testmgr - fix RNG performance in fuzz tests

    commit f900fde28883602b6c5e1027a6c912b673382aaf upstream.

    The performance of the crypto fuzz tests has greatly regressed since
    v5.18.  When booting a kernel on an arm64 dev board with all software
    crypto algorithms and CONFIG_CRYPTO_MANAGER_EXTRA_TESTS enabled, the
    fuzz tests now take about 200 seconds to run, or about 325 seconds with
    lockdep enabled, compared to about 5 seconds before.

    The root cause is that the random number generation has become much
    slower due to commit d4150779e60f ("random32: use real rng for
    non-deterministic randomness").  On my same arm64 dev board, at the time
    the fuzz tests are run, get_random_u8() is about 345x slower than
    prandom_u32_state(), or about 469x if lockdep is enabled.

    Lockdep makes a big difference, but much of the rest comes from the
    get_random_*() functions taking a *very* slow path when the CRNG is not
    yet initialized.  Since the crypto self-tests run early during boot,
    even having a hardware RNG driver enabled (CONFIG_CRYPTO_DEV_QCOM_RNG in
    my case) doesn't prevent this.  x86 systems don't have this issue, but
    they still see a significant regression if lockdep is enabled.

    Converting the "Fully random bytes" case in generate_random_bytes() to
    use get_random_bytes() helps significantly, improving the test time to
    about 27 seconds.  But that's still over 5x slower than before.

    This is all a bit silly, though, since the fuzz tests don't actually
    need cryptographically secure random numbers.  So let's just make them
    use a non-cryptographically-secure RNG as they did before.  The original
    prandom_u32() is gone now, so let's use prandom_u32_state() instead,
    with an explicitly managed state, like various other self-tests in the
    kernel source tree (rbtree_test.c, test_scanf.c, etc.) already do.  This
    also has the benefit that no locking is required anymore, so performance
    should be even better than the original version that used prandom_u32().

    Fixes: d4150779e60f ("random32: use real rng for non-deterministic randomness")
    Cc: stable@vger.kernel.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 3598a2e4bdaab22e4119c48278f970375661e8a9
Author: Eric Biggers <ebiggers@google.com>
Date:   Sun Feb 26 22:32:23 2023 -0800

    crypto: arm64/aes-neonbs - fix crash with CFI enabled

    commit 47446d7cd42358ca7d7a544f2f7823db03f616ff upstream.

    aesbs_ecb_encrypt(), aesbs_ecb_decrypt(), aesbs_xts_encrypt(), and
    aesbs_xts_decrypt() are called via indirect function calls.  Therefore
    they need to use SYM_TYPED_FUNC_START instead of SYM_FUNC_START to cause
    their type hashes to be emitted when the kernel is built with
```

```
    CONFIG_CFI_CLANG=y.  Otherwise, the code crashes with a CFI failure if
    the compiler doesn't happen to optimize out the indirect calls.

    Fixes: c50d32859e70 ("arm64: Add types to indirect called assembly functions")
    Cc: stable@vger.kernel.org
    Signed-off-by: Eric Biggers <ebiggers@google.com>
    Reviewed-by: Ard Biesheuvel <ardb@kernel.org>
    Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ab573af2655ba509e2a167897de9b5585c2ca44d
Author: Jonathan McDowell <noodles@earth.li>
Date:   Tue Feb 28 18:28:58 2023 +0000

    crypto: safexcel - Cleanup ring IRQ workqueues on load failure

    commit ca25c00ccbc5f942c63897ed23584cfc66e8ec81 upstream.

    A failure loading the safexcel driver results in the following warning
    on boot, because the IRQ affinity has not been correctly cleaned up.
    Ensure we clean up the affinity and workqueues on a failure to load the
    driver.

    crypto-safexcel: probe of f2800000.crypto failed with error -2
    ------------[ cut here ]------------
    WARNING: CPU: 1 PID: 232 at kernel/irq/manage.c:1913 free_irq+0x300/0x340
    Modules linked in: hwmon mdio_i2c crypto_safexcel(+) md5 sha256_generic libsha256
authenc libdes omap_rng rng_core nft_masq nft_nat nft_chain_nat nf_nat nft_ct nf_conntrack
nf_defrag_ipv6 nf_defrag_ipv4 nf_tables libcrc32c nfnetlink fuse autofs4
    CPU: 1 PID: 232 Comm: systemd-udevd Tainted: G        W          6.1.6-00002-
g9d4898824677 #3
    Hardware name: MikroTik RB5009 (DT)
    pstate: 600000c5 (nZCv daIF -PAN -UAO -TCO -DIT -SSBS BTYPE=--)
    pc : free_irq+0x300/0x340
    lr : free_irq+0x2e0/0x340
    sp : ffff800008fa3890
    x29: ffff800008fa3890 x28: 0000000000000000 x27: 0000000000000000
    x26: ffff8000008e6dc0 x25: ffff000009034cac x24: ffff000009034d50
    x23: 0000000000000000 x22: 000000000000004a x21: ffff0000093e0d80
    x20: ffff000009034c00 x19: ffff00000615fc00 x18: 0000000000000000
    x17: 0000000000000000 x16: 0000000000000000 x15: 000075f5c1584c5e
    x14: 0000000000000017 x13: 0000000000000000 x12: 0000000000000040
    x11: ffff000000579b60 x10: ffff000000579b62 x9 : ffff800008bbe370
    x8 : ffff000000579dd0 x7 : 0000000000000000 x6 : ffff000000579e18
    x5 : ffff000000579da8 x4 : ffff800008ca0000 x3 : ffff800008ca0188
    x2 : 0000000013033204 x1 : ffff000009034c00 x0 : ffff8000087eadf0
    Call trace:
     free_irq+0x300/0x340
     devm_irq_release+0x14/0x20
     devres_release_all+0xa0/0x100
     device_unbind_cleanup+0x14/0x60
     really_probe+0x198/0x2d4
     __driver_probe_device+0x74/0xdc
     driver_probe_device+0x3c/0x110
     __driver_attach+0x8c/0x190
     bus_for_each_dev+0x6c/0xc0
     driver_attach+0x20/0x30
     bus_add_driver+0x148/0x1fc
     driver_register+0x74/0x120
     __platform_driver_register+0x24/0x30
     safexcel_init+0x48/0x1000 [crypto_safexcel]
     do_one_initcall+0x4c/0x1b0
     do_init_module+0x44/0x1cc
     load_module+0x1724/0x1be4
     __do_sys_finit_module+0xbc/0x110
```

```
       __arm64_sys_finit_module+0x1c/0x24
       invoke_syscall+0x44/0x110
       el0_svc_common.constprop.0+0xc0/0xe0
       do_el0_svc+0x20/0x80
       el0_svc+0x14/0x4c
       el0t_64_sync_handler+0xb0/0xb4
       el0t_64_sync+0x148/0x14c
      ---[ end trace 0000000000000000 ]---
```

Fixes: 1b44c5a60c13 ("inside-secure - add SafeXcel EIP197 crypto engine driver")
Signed-off-by: Jonathan McDowell <noodles@earth.li>
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 12a90b9de602cceeab0dc169f1234e235cdcf4aa
Author: Toke Høiland-Jørgensen <toke@redhat.com>
Date: Mon Mar 13 10:17:24 2023 +0100

crypto: api - Demote BUG_ON() in crypto_unregister_alg() to a WARN_ON()

commit a543ada7db729514ddd3ba4efa45f4c7b802ad85 upstream.

The crypto_unregister_alg() function expects callers to ensure that any algorithm that is unregistered has a refcnt of exactly 1, and issues a BUG_ON() if this is not the case. However, there are in fact drivers that will call crypto_unregister_alg() without ensuring that the refcnt has been lowered first, most notably on system shutdown. This causes the BUG_ON() to trigger, which prevents a clean shutdown and hangs the system.

To avoid such hangs on shutdown, demote the BUG_ON() in crypto_unregister_alg() to a WARN_ON() with early return. Cc stable because this problem was observed on a 6.2 kernel, cf the link below.

Link: https://lore.kernel.org/r/87r0tyq8ph.fsf@toke.dk
Cc: stable@vger.kernel.org
Signed-off-by: Toke Høiland-Jørgensen <toke@redhat.com>
Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 0a65165bd24ee9231191597b7c232376fcd70cdb
Author: Johannes Berg <johannes.berg@intel.com>
Date: Thu Apr 27 17:59:20 2023 +0200

ring-buffer: Sync IRQ works before buffer destruction

commit 675751bb20634f981498c7d66161584080cc061e upstream.

If something was written to the buffer just before destruction, it may be possible (maybe not in a real system, but it did happen in ARCH=um with time-travel) to destroy the ringbuffer before the IRQ work ran, leading this KASAN report (or a crash without KASAN):

```
    BUG: KASAN: slab-use-after-free in irq_work_run_list+0x11a/0x13a
    Read of size 8 at addr 000000006d640a48 by task swapper/0

    CPU: 0 PID: 0 Comm: swapper Tainted: G        W  O       6.3.0-rc1 #7
    Stack:
     60c4f20f 0c203d48 41b58ab3 60f224fc
     600477fa 60f35687 60c4f20f 601273dd
     00000008 6101eb00 6101eab0 615be548
    Call Trace:
     [<60047a58>] show_stack+0x25e/0x282
     [<60c609e0>] dump_stack_lvl+0x96/0xfd
```

```
          [<60c50d4c>] print_report+0x1a7/0x5a8
          [<603078d3>] kasan_report+0xc1/0xe9
          [<60308950>] __asan_report_load8_noabort+0x1b/0x1d
          [<60232844>] irq_work_run_list+0x11a/0x13a
          [<602328b4>] irq_work_tick+0x24/0x34
          [<6017f9dc>] update_process_times+0x162/0x196
          [<6019f335>] tick_sched_handle+0x1a4/0x1c3
          [<6019fd9e>] tick_sched_timer+0x79/0x10c
          [<601812b9>] __hrtimer_run_queues.constprop.0+0x425/0x695
          [<60182913>] hrtimer_interrupt+0x16c/0x2c4
          [<600486a3>] um_timer+0x164/0x183
          [...]

         Allocated by task 411:
          save_stack_trace+0x99/0xb5
          stack_trace_save+0x81/0x9b
          kasan_save_stack+0x2d/0x54
          kasan_set_track+0x34/0x3e
          kasan_save_alloc_info+0x25/0x28
          ____kasan_kmalloc+0x8b/0x97
          __kasan_kmalloc+0x10/0x12
          __kmalloc+0xb2/0xe8
          load_elf_phdrs+0xee/0x182
          [...]

         The buggy address belongs to the object at 000000006d640800
          which belongs to the cache kmalloc-1k of size 1024
         The buggy address is located 584 bytes inside of
          freed 1024-byte region [000000006d640800, 000000006d640c00)
```

    Add the appropriate irq_work_sync() so the work finishes before
    the buffers are destroyed.

    Prior to the commit in the Fixes tag below, there was only a
    single global IRQ work, so this issue didn't exist.

    Link: https://lore.kernel.org/linux-trace-
kernel/20230427175920.a76159263122.I8295e405c44362a86c995e9c2c37e3e03810aa56@changeid

    Cc: stable@vger.kernel.org
    Cc: Masami Hiramatsu <mhiramat@kernel.org>
    Fixes: 15693458c4bc ("tracing/ring-buffer: Move poll wake ups into ring buffer code")
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Steven Rostedt (Google) <rostedt@goodmis.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 23d72960b776e5dc04a65067779ad8f841caece3
Author: Tze-nan Wu <Tze-nan.Wu@mediatek.com>
Date: Wed Apr 26 14:20:23 2023 +0800

    ring-buffer: Ensure proper resetting of atomic variables in
ring_buffer_reset_online_cpus

    commit 7c339fb4d8577792378136c15fde773cfb863cb8 upstream.

    In ring_buffer_reset_online_cpus, the buffer_size_kb write operation
    may permanently fail if the cpu_online_mask changes between two
    for_each_online_buffer_cpu loops. The number of increases and decreases
    on both cpu_buffer->resize_disabled and cpu_buffer->record_disabled may be
    inconsistent, causing some CPUs to have non-zero values for these atomic
    variables after the function returns.

    This issue can be reproduced by "echo 0 > trace" while hotplugging cpu.
    After reproducing success, we can find out buffer_size_kb will not be
    functional anymore.

```
    To prevent leaving 'resize_disabled' and 'record_disabled' non-zero after
    ring_buffer_reset_online_cpus returns, we ensure that each atomic variable
    has been set up before atomic_sub() to it.

    Link: https://lore.kernel.org/linux-trace-kernel/20230426062027.17451-1-Tze-
nan.Wu@mediatek.com

    Cc: stable@vger.kernel.org
    Cc: <mhiramat@kernel.org>
    Cc: npiggin@gmail.com
    Fixes: b23d7a5f4a07 ("ring-buffer: speed up buffer resets by avoiding synchronize_rcu
for each CPU")
    Reviewed-by: Cheng-Jui Wang <cheng-jui.wang@mediatek.com>
    Signed-off-by: Tze-nan Wu <Tze-nan.Wu@mediatek.com>
    Signed-off-by: Steven Rostedt (Google) <rostedt@goodmis.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit fd3b420dd66ba23188e17bb834ae013b0fb0fd4e
Author: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
Date:   Thu Mar 9 16:49:46 2023 +0100

    pinctrl: qcom: lpass-lpi: set output value before enabling output

    commit 163bfb0cb1f6fbf961cf912cbde57399ea1ae0e8 upstream.

    As per Hardware Programming Guide, when configuring pin as output,
    set the pin value before setting output-enable (OE).  Similar approach
    is in main SoC TLMM pin controller.

    Cc: <stable@vger.kernel.org>
    Fixes: 6e261d1090d6 ("pinctrl: qcom: Add sm8250 lpass lpi pinctrl driver")
    Signed-off-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Link: https://lore.kernel.org/r/20230309154949.658380-1-krzysztof.kozlowski@linaro.org
    Signed-off-by: Linus Walleij <linus.walleij@linaro.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b6ad9fa8e38ac188b8bc85a27c6525e55f093563
Author: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
Date:   Wed Feb 22 15:03:43 2023 +0100

    soundwire: qcom: correct setting ignore bit on v1.5.1

    commit bd934f77eeac377e81ddac8673803e7334b82d3d upstream.

    According to the comment and to downstream sources, the
    SWRM_CONTINUE_EXEC_ON_CMD_IGNORE in SWRM_CMD_FIFO_CFG_ADDR register
    should be set for v1.5.1 and newer, so fix the >= operator.

    Fixes: 542d3491cdd7 ("soundwire: qcom: set continue execution flag for ignored
commands")
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Krzysztof Kozlowski <krzysztof.kozlowski@linaro.org>
    Reviewed-by: Konrad Dybcio <konrad.dybcio@linaro.org>
    Link: https://lore.kernel.org/r/20230222140343.188691-1-krzysztof.kozlowski@linaro.org
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 9d47cfc7a83b2635e0dbc53ed48b76ec12d6e627
Author: Heiner Kallweit <hkallweit1@gmail.com>
Date:   Tue Apr 11 07:34:11 2023 +0200

    pwm: meson: Fix g12a ao clk81 name
```

```
    commit 9e4fa80ab7ef9eb4f7b1ea9fc31e0eb040e85e25 upstream.

    Fix the name of the aoclk81 clock. Apparently name aoclk81 as used by
    the vendor driver was changed when mainlining the g12a clock driver.

    Fixes: f41efceb46e6 ("pwm: meson: Add clock source configuration for Meson G12A")
    Cc: stable@vger.kernel.org
    Signed-off-by: Heiner Kallweit <hkallweit1@gmail.com>
    Reviewed-by: Martin Blumenstingl <martin.blumenstingl@googlemail.com>
    Signed-off-by: Thierry Reding <thierry.reding@gmail.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit fda44431109e69649a2bac9b1778c52e3a76b8c1
Author: Heiner Kallweit <hkallweit1@gmail.com>
Date:   Sun Apr 9 17:15:52 2023 +0200

    pwm: meson: Fix axg ao mux parents

    commit eb411c0cf59ae6344b34bc6f0d298a22b300627e upstream.

    This fix is basically the same as 9bce02ef0dfa ("pwm: meson: Fix the
    G12A AO clock parents order"). Vendor driver referenced there has
    xtal as first parent also for axg ao. In addition fix the name
    of the aoclk81 clock. Apparently name aoclk81 as used by the vendor
    driver was changed when mainlining the axg clock driver.

    Fixes: bccaa3f917c9 ("pwm: meson: Add clock source configuration for Meson-AXG")
    Cc: stable@vger.kernel.org
    Signed-off-by: Heiner Kallweit <hkallweit1@gmail.com>
    Reviewed-by: Martin Blumenstingl <martin.blumenstingl@googlemail.com>
    Signed-off-by: Thierry Reding <thierry.reding@gmail.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1a41c85386b45c407325e85f83c6891d01dd1dd7
Author: Eric Huang <echuang@realtek.com>
Date:   Thu Apr 6 15:28:41 2023 +0800

    wifi: rtw89: correct 5 MHz mask setting

    commit d33fc8d0368c180fe2338bfae4f5367a66a719f4 upstream.

    Use primary channel index to determine which 5 MHz mask should be enable.
    This mask is used to prevent noise from channel edge to effect CCA
    threshold in wide bandwidth (>= 40 MHZ).

    Fixes: 1b00e9236a71 ("rtw89: 8852c: add set channel of BB part")
    Fixes: 6b0698984eb0 ("wifi: rtw89: 8852b: add chip_ops::set_channel")
    Cc: stable@vger.kernel.org
    Signed-off-by: Eric Huang <echuang@realtek.com>
    Signed-off-by: Ping-Ke Shih <pkshih@realtek.com>
    Signed-off-by: Kalle Valo <kvalo@kernel.org>
    Link: https://lore.kernel.org/r/20230406072841.8308-1-pkshih@realtek.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 24dec13b5fbceca81bcf74d07bad24ff3b165607
Author: Felix Fietkau <nbd@nbd.name>
Date:   Fri Apr 14 14:10:54 2023 +0200

    wifi: mt76: add missing locking to protect against concurrent rx/status calls

    commit 5b8ccdfb943f6a03c676d2ea816dd38c149e920b upstream.

    According to the documentation, ieee80211_rx_list must not run concurrently
    with ieee80211_tx_status (or its variants).
```

```
    Cc: stable@vger.kernel.org
    Fixes: 88046b2c9f6d ("mt76: add support for reporting tx status with skb")
    Reported-by: Brian Coverstone <brian@mainsequence.net>
    Signed-off-by: Felix Fietkau <nbd@nbd.name>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 82d2e01b95c439fe55fab5e04fc83387c42d3a48
Author: Kees Cook <keescook@chromium.org>
Date:   Thu Mar 2 14:49:50 2023 -0800

    kheaders: Use array declaration instead of char

    commit b69edab47f1da8edd8e7bfdf8c70f51a2a5d89fb upstream.

    Under CONFIG_FORTIFY_SOURCE, memcpy() will check the size of destination
    and source buffers. Defining kernel_headers_data as "char" would trip
    this check. Since these addresses are treated as byte arrays, define
    them as arrays (as done everywhere else).

    This was seen with:

      $ cat /sys/kernel/kheaders.tar.xz >> /dev/null

      detected buffer overflow in memcpy
      kernel BUG at lib/string_helpers.c:1027!
      ...
      RIP: 0010:fortify_panic+0xf/0x20
      [...]
      Call Trace:
       <TASK>
       ikheaders_read+0x45/0x50 [kheaders]
       kernfs_fop_read_iter+0x1a4/0x2f0
      ...

    Reported-by: Jakub Kicinski <kuba@kernel.org>
    Link: https://lore.kernel.org/bpf/20230302112130.6e402a98@kernel.org/
    Acked-by: Joel Fernandes (Google) <joel@joelfernandes.org>
    Reviewed-by: Alexander Lobakin <aleksander.lobakin@intel.com>
    Tested-by: Jakub Kicinski <kuba@kernel.org>
    Fixes: 43d8ce9d65a5 ("Provide in-kernel headers to make extending kernel easier")
    Cc: stable@vger.kernel.org
    Signed-off-by: Kees Cook <keescook@chromium.org>
    Link: https://lore.kernel.org/r/20230302224946.never.243-kees@kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 6ef67358f9f2b75d4716a7f4747e25cfe0f1328c
Author: William Breathitt Gray <william.gray@linaro.org>
Date:   Thu Apr 6 10:40:10 2023 -0400

    iio: addac: stx104: Fix race condition for stx104_write_raw()

    commit 9740827468cea80c42db29e7171a50e99acf7328 upstream.

    The priv->chan_out_states array and actual DAC value can become
    mismatched if stx104_write_raw() is called concurrently. Prevent such a
    race condition by utilizing a mutex.

    Fixes: 97a445dad37a ("iio: Add IIO support for the DAC on the Apex Embedded Systems
STX104")
    Signed-off-by: William Breathitt Gray <william.gray@linaro.org>
    Link:
https://lore.kernel.org/r/c95c9a77fcef36b2a052282146950f23bbc1ebdc.1680790580.git.william.gr
ay@linaro.org
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
```

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

commit 1890dcbb12e0c75bdfbc80a59dd433c5fec3eec6
Author: William Breathitt Gray <william.gray@linaro.org>
Date:   Thu Apr 6 10:40:11 2023 -0400

    iio: addac: stx104: Fix race condition when converting analog-to-digital

    commit 4f9b80aefb9e2f542a49d9ec087cf5919730e1dd upstream.

    The ADC conversion procedure requires several device I/O operations
    performed in a particular sequence. If stx104_read_raw() is called
    concurrently, the ADC conversion procedure could be clobbered. Prevent
    such a race condition by utilizing a mutex.

    Fixes: 4075a283ae83 ("iio: stx104: Add IIO support for the ADC channels")
    Signed-off-by: William Breathitt Gray <william.gray@linaro.org>
    Link:
https://lore.kernel.org/r/2ae5e40eed5006ca735e4c12181a9ff5ced65547.1680790580.git.william.gr
ay@linaro.org
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit d63f8f5a8812ec7023bd8c90cb08e5091a7d0d2c
Author: Zhang Yuchen <zhangyuchen.lcr@bytedance.com>
Date:   Wed Apr 12 15:49:07 2023 +0800

    ipmi: fix SSIF not responding under certain cond.

    commit 6d2555cde2918409b0331560e66f84a0ad4849c6 upstream.

    The ipmi communication is not restored after a specific version of BMC is
    upgraded on our server.
    The ipmi driver does not respond after printing the following log:

        ipmi_ssif: Invalid response getting flags: 1c 1

    I found that after entering this branch, ssif_info->ssif_state always
    holds SSIF_GETTING_FLAGS and never return to IDLE.

    As a result, the driver cannot be loaded, because the driver status is
    checked during the unload process and must be IDLE in shutdown_ssif():

            while (ssif_info->ssif_state != SSIF_IDLE)
                    schedule_timeout(1);

    The process trigger this problem is:

    1. One msg timeout and next msg start send, and call
    ssif_set_need_watch().

    2. ssif_set_need_watch()->watch_timeout()->start_flag_fetch() change
    ssif_state to SSIF_GETTING_FLAGS.

    3. In msg_done_handler() ssif_state == SSIF_GETTING_FLAGS, if an error
    message is received, the second branch does not modify the ssif_state.

    4. All retry action need IS_SSIF_IDLE() == True. Include retry action in
    watch_timeout(), msg_done_handler(). Sending msg does not work either.
    SSIF_IDLE is also checked in start_next_msg().

    5. The only thing that can be triggered in the SSIF driver is
    watch_timeout(), after destory_user(), this timer will stop too.
```

```
    So, if enter this branch, the ssif_state will remain SSIF_GETTING_FLAGS
    and can't send msg, no timer started, can't unload.

    We did a comparative test before and after adding this patch, and the
    result is effective.

    Fixes: 259307074bfc ("ipmi: Add SMBus interface driver (SSIF)")

    Cc: stable@vger.kernel.org
    Signed-off-by: Zhang Yuchen <zhangyuchen.lcr@bytedance.com>
    Message-Id: <20230412074907.80046-1-zhangyuchen.lcr@bytedance.com>
    Signed-off-by: Corey Minyard <minyard@acm.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 57f53984466e12340b1ea8f021e30712ca393f6b
Author: Corey Minyard <minyard@acm.org>
Date:   Tue Apr 4 12:09:14 2023 +0000

    ipmi:ssif: Add send_retries increment

    commit 6ce7995a43febe693d4894033c6e29314970646a upstream.

    A recent change removed an increment of send_retries, re-add it.

    Fixes: 95767ed78a18 ipmi:ssif: resend_msg() cannot fail
    Reported-by: Pavel Machek <pavel@denx.de>
    Cc: stable@vger.kernel.org
    Signed-off-by: Corey Minyard <minyard@acm.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit aeed787bbbbe1b842beec9a065a36c915226f704
Author: Jiaxun Yang <jiaxun.yang@flygoat.com>
Date:   Tue Apr 11 12:14:26 2023 +0100

    MIPS: fw: Allow firmware to pass a empty env

    commit ee1809ed7bc456a72dc8410b475b73021a3a68d5 upstream.

    fw_getenv will use env entry to determine style of env,
    however it is legal for firmware to just pass a empty list.

    Check if first entry exist before running strchr to avoid
    null pointer dereference.

    Cc: stable@vger.kernel.org
    Link: https://github.com/clbr/n64bootloader/issues/5
    Signed-off-by: Jiaxun Yang <jiaxun.yang@flygoat.com>
    Signed-off-by: Thomas Bogendoerfer <tsbogend@alpha.franken.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 59756c4ea291721e5203e0b85e3974f2e65e5e82
Author: Kefeng Wang <wangkefeng.wang@huawei.com>
Date:   Fri Mar 31 16:45:02 2023 +0800

    fs: fix sysctls.c built

    commit e3184de9d46c2eebdb776face2e2662c6733331d upstream.

    'obj-$(CONFIG_SYSCTL) += sysctls.o' must be moved after "obj-y :=",
    or it won't be built as it is overwrited.

    Note that there is nothing that is going to break by linking
    sysctl.o later, we were just being way to cautious and patches
    have been updated to reflect these considerations and sent for
    stable as well with the whole "base" stuff needing to be linked
```

```
    prior to child sysctl tables that use that directory. All of
    the kernel sysctl APIs always share the same directory, and races
    against using it should end up re-using the same single created
    directory.

    And so something we can do eventually is do away with all the base stuff.
    For now it's fine, it's not creating an issue. It is just a bit pedantic
    and careful.

    Fixes: ab171b952c6e ("fs: move namespace sysctls and declare fs base directory")
    Cc: stable@vger.kernel.org # v5.17
    Cc: Christian Brauner <brauner@kernel.org>
    Cc: Kefeng Wang <wangkefeng.wang@huawei.com>
    Signed-off-by: Kefeng Wang <wangkefeng.wang@huawei.com>
    [mcgrof: enhanced commit log for stable criteria and clarify base stuff ]
    Signed-off-by: Luis Chamberlain <mcgrof@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit af27e76cd649f8e6a9e305f121c16d64b7bc5cec
Author: Joel Fernandes (Google) <joel@joelfernandes.org>
Date:   Tue Jan 24 17:31:26 2023 +0000

    tick/nohz: Fix cpu_is_hotpluggable() by checking with nohz subsystem

    commit 58d7668242647e661a20efe065519abd6454287e upstream.

    For CONFIG_NO_HZ_FULL systems, the tick_do_timer_cpu cannot be offlined.
    However, cpu_is_hotpluggable() still returns true for those CPUs. This causes
    torture tests that do offlining to end up trying to offline this CPU causing
    test failures. Such failure happens on all architectures.

    Fix the repeated error messages thrown by this (even if the hotplug errors are
    harmless) by asking the opinion of the nohz subsystem on whether the CPU can be
    hotplugged.

    [ Apply Frederic Weisbecker feedback on refactoring tick_nohz_cpu_down(). ]

    For drivers/base/ portion:
    Acked-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Acked-by: Frederic Weisbecker <frederic@kernel.org>
    Cc: Frederic Weisbecker <frederic@kernel.org>
    Cc: "Paul E. McKenney" <paulmck@kernel.org>
    Cc: Zhouyi Zhou <zhouzhouyi@gmail.com>
    Cc: Will Deacon <will@kernel.org>
    Cc: Marc Zyngier <maz@kernel.org>
    Cc: rcu <rcu@vger.kernel.org>
    Cc: stable@vger.kernel.org
    Fixes: 2987557f52b9 ("driver-core/cpu: Expose hotpluggability to the rest of the
kernel")
    Signed-off-by: Paul E. McKenney <paulmck@kernel.org>
    Signed-off-by: Joel Fernandes (Google) <joel@joelfernandes.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 855490a4334a7fb7927c375b9cde6175656c710c
Author: Jan Kundrát <jan.kundrat@cesnet.cz>
Date:   Wed Apr 5 22:14:23 2023 +0200

    serial: max310x: fix IO data corruption in batched operations

    commit 3f42b142ea1171967e40e10e4b0241c0d6d28d41 upstream.

    After upgrading from 5.16 to 6.1, our board with a MAX14830 started
    producing lots of garbage data over UART. Bisection pointed out commit
    285e76fc049c as the culprit. That patch tried to replace hand-written
    code which I added in 2b4bac48c1084 ("serial: max310x: Use batched reads
```

```
    when reasonably safe") with the generic regmap infrastructure for
    batched operations.

    Unfortunately, the `regmap_raw_read` and `regmap_raw_write` which were
    used are actually functions which perform IO over *multiple* registers.
    That's not what is needed for accessing these Tx/Rx FIFOs; the
    appropriate functions are the `_noinc_` versions, not the `_raw_` ones.

    Fix this regression by using `regmap_noinc_read()` and
    `regmap_noinc_write()` along with the necessary `regmap_config` setup;
    with this patch in place, our board communicates happily again. Since
    our board uses SPI for talking to this chip, the I2C part is completely
    untested.

    Fixes: 285e76fc049c ("serial: max310x: use regmap methods for SPI batch operations")
    Cc: stable@vger.kernel.org
    Reviewed-by: Andy Shevchenko <andy.shevchenko@gmail.com>
    Signed-off-by: Jan Kundrát <jan.kundrat@cesnet.cz>
    Link:
https://lore.kernel.org/r/79db8e82aadb0e174bc82b9996423c3503c8fb37.1680732084.git.jan.kundra
t@cesnet.cz
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 07e3c26c200a492b21e0867b4278d410a09a134d
Author: Ilpo Järvinen <ilpo.jarvinen@linux.intel.com>
Date:   Fri Mar 17 13:33:18 2023 +0200

    serial: 8250: Fix serial8250_tx_empty() race with DMA Tx

    commit 146a37e05d620cef4ad430e5d1c9c077fe6fa76f upstream.

    There's a potential race before THRE/TEMT deasserts when DMA Tx is
    starting up (or the next batch of continuous Tx is being submitted).
    This can lead to misdetecting Tx empty condition.

    It is entirely normal for THRE/TEMT to be set for some time after the
    DMA Tx had been setup in serial8250_tx_dma(). As Tx side is definitely
    not empty at that point, it seems incorrect for serial8250_tx_empty()
    claim Tx is empty.

    Fix the race by also checking in serial8250_tx_empty() whether there's
    DMA Tx active.

    Note: This fix only addresses in-kernel race mainly to make using
    TCSADRAIN/FLUSH robust. Userspace can still cause other races but they
    seem userspace concurrency control problems.

    Fixes: 9ee4b83e51f74 ("serial: 8250: Add support for dmaengine")
    Cc: stable@vger.kernel.org
    Signed-off-by: Ilpo Järvinen <ilpo.jarvinen@linux.intel.com>
    Link: https://lore.kernel.org/r/20230317113318.31327-3-ilpo.jarvinen@linux.intel.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit deb309ffda4facaedbcbe47dcfe37a5a6d0cc788
Author: Johan Hovold <johan@kernel.org>
Date:   Wed Apr 12 14:48:11 2023 +0200

    serial: fix TIOCSRS485 locking

    commit 9e4f2a8004213339e9d837d891a59cc80e082966 upstream.

    The RS485 multipoint addressing support for some reason added a new
    ADDRB termios cflag which is (only!) updated from one of the RS485
    ioctls.
```

```
    Make sure to take the termios rw semaphore for the right ioctl (i.e.
    set, not get).

    Fixes: ae50bb275283 ("serial: take termios_rwsem for ->rs485_config() & pass termios as
param")
    Cc: stable@vger.kernel.org      # 6.0
    Cc: Ilpo Järvinen <ilpo.jarvinen@linux.intel.com>
    Reviewed-by: Ilpo Järvinen <ilpo.jarvinen@linux.intel.com>
    Signed-off-by: Johan Hovold <johan@kernel.org>
    Link: https://lore.kernel.org/r/20230412124811.11217-1-johan@kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1a82446d409bad70d6333a250c43c1f35823da7d
Author: Johan Hovold <johan+linaro@kernel.org>
Date:   Wed Apr 5 11:03:42 2023 +0200

    xhci: fix debugfs register accesses while suspended

    commit 735baf1b23458f71a8b15cb924af22c9ff9cd125 upstream.

    Wire up the debugfs regset device pointer so that the controller is
    resumed before accessing registers to avoid crashing or locking up if it
    happens to be runtime suspended.

    Fixes: 02b6fdc2a153 ("usb: xhci: Add debugfs interface for xHCI driver")
    Cc: stable@vger.kernel.org # 4.15: 30332eeefec8: debugfs: regset32: Add Runtime PM
support
    Cc: stable@vger.kernel.org # 4.15
    Signed-off-by: Johan Hovold <johan+linaro@kernel.org>
    Link: https://lore.kernel.org/r/20230405090342.7363-1-johan+linaro@kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3f9535ae2e7a86a06b261acc153cb685e24d0b1a
Author: Ilpo Järvinen <ilpo.jarvinen@linux.intel.com>
Date:   Fri Mar 17 13:33:17 2023 +0200

    tty: Prevent writing chars during tcsetattr TCSADRAIN/FLUSH

    commit 094fb49a2d0d6827c86d2e0840873e6db0c491d2 upstream.

    If userspace races tcsetattr() with a write, the drained condition
    might not be guaranteed by the kernel. There is a race window after
    checking Tx is empty before tty_set_termios() takes termios_rwsem for
    write. During that race window, more characters can be queued by a
    racing writer.

    Any ongoing transmission might produce garbage during HW's
    ->set_termios() call. The intent of TCSADRAIN/FLUSH seems to be
    preventing such a character corruption. If those flags are set, take
    tty's write lock to stop any writer before performing the lower layer
    Tx empty check and wait for the pending characters to be sent (if any).

    The initial wait for all-writers-done must be placed outside of tty's
    write lock to avoid deadlock which makes it impossible to use
    tty_wait_until_sent(). The write lock is retried if a racing write is
    detected.

    Fixes: 1da177e4c3f4 ("Linux-2.6.12-rc2")
    Cc: stable@vger.kernel.org
    Signed-off-by: Ilpo Järvinen <ilpo.jarvinen@linux.intel.com>
    Link: https://lore.kernel.org/r/20230317113318.31327-2-ilpo.jarvinen@linux.intel.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f079bffbdda5858ee60b718cb7adfbb8213d94b8
Author: Nuno Sá <nuno.sa@analog.com>
```

```
Date:   Mon Mar 27 16:54:14 2023 +0200

    staging: iio: resolver: ads1210: fix config mode

    commit 16313403d873ff17a587818b61f84c8cb4971cef upstream.

    As stated in the device datasheet [1], bits a0 and a1 have to be set to
    1 for the configuration mode.

    [1]: https://www.analog.com/media/en/technical-documentation/data-sheets/ad2s1210.pdf

    Fixes: b19e9ad5e2cb9 ("staging:iio:resolver:ad2s1210 general driver cleanup")
    Cc: stable <stable@kernel.org>
    Signed-off-by: Nuno Sá <nuno.sa@analog.com>
    Link: https://lore.kernel.org/r/20230327145414.1505537-1-nuno.sa@analog.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 5c62852942667c613de0458fc797c5b8c36112b5
Author: Eric Biggers <ebiggers@google.com>
Date:   Wed Mar 15 11:39:04 2023 -0700

    blk-crypto: make blk_crypto_evict_key() more robust

    commit 5c7cb94452901a93e90c2230632e2c12a681bc92 upstream.

    If blk_crypto_evict_key() sees that the key is still in-use (due to a
    bug) or that ->keyslot_evict failed, it currently just returns while
    leaving the key linked into the keyslot management structures.

    However, blk_crypto_evict_key() is only called in contexts such as inode
    eviction where failure is not an option.  So actually the caller
    proceeds with freeing the blk_crypto_key regardless of the return value
    of blk_crypto_evict_key().

    These two assumptions don't match, and the result is that there can be a
    use-after-free in blk_crypto_reprogram_all_keys() after one of these
    errors occurs.  (Note, these errors *shouldn't* happen; we're just
    talking about what happens if they do anyway.)

    Fix this by making blk_crypto_evict_key() unlink the key from the
    keyslot management structures even on failure.

    Also improve some comments.

    Fixes: 1b2628397058 ("block: Keyslot Manager for Inline Encryption")
    Cc: stable@vger.kernel.org
    Signed-off-by: Eric Biggers <ebiggers@google.com>
    Reviewed-by: Christoph Hellwig <hch@lst.de>
    Link: https://lore.kernel.org/r/20230315183907.53675-2-ebiggers@kernel.org
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3a8cae24cb19ce41576efa259fa67c93114ca91f
Author: Eric Biggers <ebiggers@google.com>
Date:   Wed Mar 15 11:39:03 2023 -0700

    blk-crypto: make blk_crypto_evict_key() return void

    commit 70493a63ba04f754f7a7dd53a4fcc82700181490 upstream.

    blk_crypto_evict_key() is only called in contexts such as inode eviction
    where failure is not an option.  So there is nothing the caller can do
    with errors except log them.  (dm-table.c does "use" the error code, but
    only to pass on to upper layers, so it doesn't really count.)
```

```
    Just make blk_crypto_evict_key() return void and log errors itself.

    Cc: stable@vger.kernel.org
    Signed-off-by: Eric Biggers <ebiggers@google.com>
    Reviewed-by: Christoph Hellwig <hch@lst.de>
    Link: https://lore.kernel.org/r/20230315183907.53675-2-ebiggers@kernel.org
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 92d5d233b9ff531cf9cc36ab4251779e07adb633
Author: Eric Biggers <ebiggers@google.com>
Date:   Wed Mar 15 11:39:02 2023 -0700

    blk-mq: release crypto keyslot before reporting I/O complete

    commit 9cd1e566676bbcb8a126acd921e4e194e6339603 upstream.

    Once all I/O using a blk_crypto_key has completed, filesystems can call
    blk_crypto_evict_key().  However, the block layer currently doesn't call
    blk_crypto_put_keyslot() until the request is being freed, which happens
    after upper layers have been told (via bio_endio()) the I/O has
    completed.  This causes a race condition where blk_crypto_evict_key()
    can see 'slot_refs != 0' without there being an actual bug.

    This makes __blk_crypto_evict_key() hit the
    'WARN_ON_ONCE(atomic_read(&slot->slot_refs) != 0)' and return without
    doing anything, eventually causing a use-after-free in
    blk_crypto_reprogram_all_keys().  (This is a very rare bug and has only
    been seen when per-file keys are being used with fscrypt.)

    There are two options to fix this: either release the keyslot before
    bio_endio() is called on the request's last bio, or make
    __blk_crypto_evict_key() ignore slot_refs.  Let's go with the first
    solution, since it preserves the ability to report bugs (via
    WARN_ON_ONCE) where a key is evicted while still in-use.

    Fixes: a892c8d52c02 ("block: Inline encryption support for blk-mq")
    Cc: stable@vger.kernel.org
    Reviewed-by: Nathan Huckleberry <nhuck@google.com>
    Reviewed-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Eric Biggers <ebiggers@google.com>
    Link: https://lore.kernel.org/r/20230315183907.53675-2-ebiggers@kernel.org
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f03e066199dbd7193d06665b27c469b37b8b56f2
Author: Chengming Zhou <zhouchengming@bytedance.com>
Date:   Thu Apr 13 14:28:04 2023 +0800

    blk-stat: fix QUEUE_FLAG_STATS clear

    commit 20de765f6d9da0c47b756429c60b41063b990a10 upstream.

    We need to set QUEUE_FLAG_STATS for two cases:
    1. blk_stat_enable_accounting()
    2. blk_stat_add_callback()

    So we should clear it only when ((q->stats->accounting == 0) &&
    list_empty(&q->stats->callbacks)).

    blk_stat_disable_accounting() only check if q->stats->accounting
    is 0 before clear the flag, this patch fix it.

    Also add list_empty(&q->stats->callbacks)) check when enable, or
    the flag is already set.
```

```
    The bug can be reproduced on kernel without BLK_DEV_THROTTLING
    (since it unconditionally enable accounting, see the next patch).

      # cat /sys/block/sr0/queue/scheduler
      none mq-deadline [bfq]

      # cat /sys/kernel/debug/block/sr0/state
      SAME_COMP|IO_STAT|INIT_DONE|STATS|REGISTERED|NOWAIT|30

      # echo none > /sys/block/sr0/queue/scheduler

      # cat /sys/kernel/debug/block/sr0/state
      SAME_COMP|IO_STAT|INIT_DONE|REGISTERED|NOWAIT

      # cat /sys/block/sr0/queue/wbt_lat_usec
      75000

    We can see that after changing elevator from "bfq" to "none",
    "STATS" flag is lost even though WBT callback still need it.

    Fixes: 68497092bde9 ("block: make queue stat accounting a reference")
    Cc: <stable@vger.kernel.org> # v5.17+
    Signed-off-by: Chengming Zhou <zhouchengming@bytedance.com>
    Acked-by: Tejun Heo <tj@kernel.org>
    Link: https://lore.kernel.org/r/20230413062805.2081970-1-chengming.zhou@linux.dev
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 8af736aeb3586835f2079a837d928e17323b1617
Author: Martin Krastev <krastevm@vmware.com>
Date:   Mon Mar 20 22:09:49 2023 -0400

    drm/vmwgfx: Fix Legacy Display Unit atomic drm support

    commit a37a512db3fa1b65fe9087003e5b2072cefb3667 upstream.

    Legacy Display Unit (LDU) fb dirty support used a custom fb dirty callback. Latter
    handled only the DIRTYFB IOCTL presentation path but not the ADDFB2/PAGE_FLIP/RMFB
    IOCTL path, common for Wayland compositors.

    Get rid of the custom callback in favor of drm_atomic_helper_dirtyfb and unify the
    handling of the presentation paths inside of vmw_ldu_primary_plane_atomic_update.
    This also homogenizes the fb dirty callbacks across all DUs: LDU, SOU and STDU.

    Signed-off-by: Martin Krastev <krastevm@vmware.com>
    Reviewed-by: Maaz Mombasawala <mombasawalam@vmware.com>
    Fixes: 2f5544ff0300 ("drm/vmwgfx: Use atomic helper function for dirty fb IOCTL")
    Cc: <stable@vger.kernel.org> # v5.0+
    Signed-off-by: Zack Rusin <zackr@vmware.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20230321020949.335012-3-zack@kde.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 79e09cbfb64c9673b6b4106866ccd832d51c4c32
Author: Ricardo Ribalda <ribalda@chromium.org>
Date:   Thu Mar 23 23:44:20 2023 +0100

    media: ov8856: Do not check for for module version

    commit 5a4e1b5aed2a36a10d6a3b30fafb6b3bf41c3186 upstream.

    It the device is probed in non-zero ACPI D state, the module
    identification is delayed until the first streamon.

    The module identification has two parts: deviceID and version. To rea
```

```
    the version we have to enable OTP read. This cannot be done during
    streamon, becase it modifies REG_MODE_SELECT.

    Since the driver has the same behaviour for all the module versions, do
    not read the module version from the sensor's OTP.

    Cc: stable@vger.kernel.org
    Fixes: 0e014f1a8d54 ("media: ov8856: support device probe in non-zero ACPI D state")
    Signed-off-by: Ricardo Ribalda <ribalda@chromium.org>
    Signed-off-by: Sakari Ailus <sakari.ailus@linux.intel.com>
    Signed-off-by: Hans Verkuil <hverkuil-cisco@xs4all.nl>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 2a83c96586253c3ece275bf80fa5a3ceb6bd5256
Author: Thomas Gleixner <tglx@linutronix.de>
Date:   Mon Apr 17 15:37:55 2023 +0200

    posix-cpu-timers: Implement the missing timer_wait_running callback

    commit f7abf14f0001a5a47539d9f60bbdca649e43536b upstream.

    For some unknown reason the introduction of the timer_wait_running callback
    missed to fixup posix CPU timers, which went unnoticed for almost four years.
    Marco reported recently that the WARN_ON() in timer_wait_running()
    triggers with a posix CPU timer test case.

    Posix CPU timers have two execution models for expiring timers depending on
    CONFIG_POSIX_CPU_TIMERS_TASK_WORK:

    1) If not enabled, the expiry happens in hard interrupt context so
       spin waiting on the remote CPU is reasonably time bound.

       Implement an empty stub function for that case.

    2) If enabled, the expiry happens in task work before returning to user
       space or guest mode. The expired timers are marked as firing and moved
       from the timer queue to a local list head with sighand lock held. Once
       the timers are moved, sighand lock is dropped and the expiry happens in
       fully preemptible context. That means the expiring task can be scheduled
       out, migrated, interrupted etc. So spin waiting on it is more than
       suboptimal.

       The timer wheel has a timer_wait_running() mechanism for RT, which uses
       a per CPU timer-base expiry lock which is held by the expiry code and the
       task waiting for the timer function to complete blocks on that lock.

       This does not work in the same way for posix CPU timers as there is no
       timer base and expiry for process wide timers can run on any task
       belonging to that process, but the concept of waiting on an expiry lock
       can be used too in a slightly different way:

        - Add a mutex to struct posix_cputimers_work. This struct is per task
          and used to schedule the expiry task work from the timer interrupt.

        - Add a task_struct pointer to struct cpu_timer which is used to store
          a the task which runs the expiry. That's filled in when the task
          moves the expired timers to the local expiry list. That's not
          affecting the size of the k_itimer union as there are bigger union
          members already

        - Let the task take the expiry mutex around the expiry function

        - Let the waiter acquire a task reference with rcu_read_lock() held and
          block on the expiry mutex
```

```
       This avoids spin-waiting on a task which might not even be on a CPU and
       works nicely for RT too.

    Fixes: ec8f954a40da ("posix-timers: Use a callback for cancel synchronization on
PREEMPT_RT")
    Reported-by: Marco Elver <elver@google.com>
    Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
    Tested-by: Marco Elver <elver@google.com>
    Tested-by: Sebastian Andrzej Siewior <bigeasy@linutronix.de>
    Reviewed-by: Frederic Weisbecker <frederic@kernel.org>
    Cc: stable@vger.kernel.org
    Link: https://lore.kernel.org/r/87zg764ojw.ffs@tglx
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 0af0a989e747248e05640980661225e5b94cdb9e
Author: Jarkko Sakkinen <jarkko@kernel.org>
Date:   Sun Apr 23 18:49:58 2023 +0300

    tpm: Add !tpm_amd_is_rng_defective() to the hwrng_unregister() call site

    commit bd8621ca1510e6e802df9855bdc35a04a3cfa932 upstream.

    The following crash was reported:

    [ 1950.279393] list_del corruption, ffff99560d485790->next is NULL
    [ 1950.279400] ------------[ cut here ]------------
    [ 1950.279401] kernel BUG at lib/list_debug.c:49!
    [ 1950.279405] invalid opcode: 0000 [#1] PREEMPT SMP NOPTI
    [ 1950.279407] CPU: 11 PID: 5886 Comm: modprobe Tainted: G O 6.2.8_1 #1
    [ 1950.279409] Hardware name: Gigabyte Technology Co., Ltd. B550M AORUS PRO-P/B550M
AORUS PRO-P,
    BIOS F15c 05/11/2022
    [ 1950.279410] RIP: 0010:__list_del_entry_valid+0x59/0xc0
    [ 1950.279415] Code: 48 8b 01 48 39 f8 75 5a 48 8b 72 08 48 39 c6 75 65 b8 01 00 00 00
c3 cc cc cc
    cc 48 89 fe 48 c7 c7 08 a8 13 9e e8 b7 0a bc ff <0f> 0b 48 89 fe 48 c7 c7 38 a8 13 9e e8
a6 0a bc
    ff 0f 0b 48 89 fe
    [ 1950.279416] RSP: 0018:ffffa96d05647e08 EFLAGS: 00010246
    [ 1950.279418] RAX: 0000000000000033 RBX: ffff99560d485750 RCX: 0000000000000000
    [ 1950.279419] RDX: 0000000000000000 RSI: ffffffff9e107c59 RDI: 00000000ffffffff
    [ 1950.279420] RBP: ffffffffc19c5168 R08: 0000000000000000 R09: ffffa96d05647cc8
    [ 1950.279421] R10: 0000000000000003 R11: ffffffff9ea2a568 R12: 0000000000000000
    [ 1950.279422] R13: ffff99560140a2e0 R14: ffff99560127d2e0 R15: 0000000000000000
    [ 1950.279422] FS: 00007f67da795380(0000) GS:ffff995d1f0c0000(0000)
knlGS:0000000000000000
    [ 1950.279424] CS: 0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    [ 1950.279424] CR2: 00007f67da7e65c0 CR3: 00000001feed2000 CR4: 0000000000750ee0
    [ 1950.279426] PKRU: 55555554
    [ 1950.279426] Call Trace:
    [ 1950.279428] <TASK>
    [ 1950.279430] hwrng_unregister+0x28/0xe0 [rng_core]
    [ 1950.279436] tpm_chip_unregister+0xd5/0xf0 [tpm]

    Add the forgotten !tpm_amd_is_rng_defective() invariant to the
    hwrng_unregister() call site inside tpm_chip_unregister().

    Cc: stable@vger.kernel.org
    Reported-by: Martin Dimov <martin@dmarto.com>
    Link: https://lore.kernel.org/linux-
integrity/3d1d7e9dbfb8c96125bc93b6b58b90a7@dmarto.com/
    Fixes: f1324bbc4011 ("tpm: disable hwrng for fTPM on some AMD designs")
    Fixes: b006c439d58d ("hwrng: core - start hwrng kthread also for untrusted sources")
    Tested-by: Martin Dimov <martin@dmarto.com>
    Signed-off-by: Jarkko Sakkinen <jarkko@kernel.org>
```

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

commit bc32bb7801966a210b024a4344dd8d75a4f08ef0
Author: Chris Packham <chris.packham@alliedtelesis.co.nz>
Date:   Wed Apr 19 11:36:55 2023 +1200

    hwmon: (adt7475) Use device_property APIs when configuring polarity

    commit 2a8e41ad337508fc5d598c0f9288890214f8e318 upstream.

    On DT unaware platforms of_property_read_u32_array() returns -ENOSYS
    which wasn't handled by the code treating adi,pwm-active-state as
    optional. Update the code to use device_property_read_u32_array() which
    deals gracefully with DT unaware platforms.

    Fixes: 86da28eed4fb ("hwmon: (adt7475) Add support for inverting pwm output")
    Reported-by: Mariusz Białończyk <manio@skyboo.net>
    Link: https://lore.kernel.org/linux-hwmon/52e26a67-9131-2dc0-40cb-
db5c07370027@alliedtelesis.co.nz/T/#mdd0505801e0a4e72340de009a47c0fca4f771ed3
    Signed-off-by: Chris Packham <chris.packham@alliedtelesis.co.nz>
    Link: https://lore.kernel.org/r/20230418233656.869055-2-
chris.packham@alliedtelesis.co.nz
    Cc: stable@vger.kernel.org
    Signed-off-by: Guenter Roeck <linux@roeck-us.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 2ea2aaf1a4a036522a03a1b00ae165d892e97bb1
Author: Babu Moger <Babu.Moger@amd.com>
Date:   Thu Apr 13 16:39:58 2023 -0500

    hwmon: (k10temp) Check range scale when CUR_TEMP register is read-write

    commit 0c072385348e3ac5229145644055d3e2afb5b3db upstream.

    Spec says, when CUR_TEMP_TJ_SEL == 3 and CUR_TEMP_RANGE_SEL == 0,
    it should use RangeUnadjusted is 0, which is (CurTmp*0.125 -49) C. The
    CUR_TEMP register is read-write when CUR_TEMP_TJ_SEL == 3 (bit 17-16).

    Add the check to detect it.

    Sensors command's output before the patch.
    $sensors
     k10temp-pci-00c3
     Adapter: PCI adapter
     Tctl:         +76.6°C <- Wrong value
     Tccd1:        +26.5°C
     Tccd2:        +27.5°C
     Tccd3:        +27.2°C
     Tccd4:        +27.5°C
     Tccd5:        +26.0°C
     Tccd6:        +26.2°C
     Tccd7:        +25.0°C
     Tccd8:        +26.5°C

    Sensors command's output after the patch.
    $sensors
     k10temp-pci-00c3
     Adapter: PCI adapter
     Tctl:         +28.8°C <- corrected value
     Tccd1:        +27.5°C
     Tccd2:        +28.5°C
     Tccd3:        +28.5°C
     Tccd4:        +28.5°C
     Tccd5:        +27.0°C
     Tccd6:        +27.5°C
```

```
       Tccd7:        +27.0°C
       Tccd8:        +27.5°C

    Signed-off-by: Babu Moger <babu.moger@amd.com>
    Fixes: 1b59788979ac ("hwmon: (k10temp) Add temperature offset for Ryzen 2700X")
    Link: https://lore.kernel.org/r/20230413213958.847634-1-babu.moger@amd.com
    Cc: stable@vger.kernel.org
    Signed-off-by: Guenter Roeck <linux@roeck-us.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 89fd2a153944939d02eb1440bfb195cae9a7b5b6
Author: Johan Hovold <johan+linaro@kernel.org>
Date:   Tue Apr 4 09:25:15 2023 +0200

    USB: dwc3: fix runtime pm imbalance on unbind

    commit 44d257e9012ee8040e41d224d0e5bfb5ef5427ea upstream.

    Make sure to balance the runtime PM usage count on driver unbind by
    adding back the pm_runtime_allow() call that had been erroneously
    removed.

    Fixes: 266d0493900a ("usb: dwc3: core: don't trigger runtime pm when remove driver")
    Cc: stable@vger.kernel.org      # 5.9
    Cc: Li Jun <jun.li@nxp.com>
    Acked-by: Thinh Nguyen <Thinh.Nguyen@synopsys.com>
    Signed-off-by: Johan Hovold <johan+linaro@kernel.org>
    Link: https://lore.kernel.org/r/20230404072524.19014-3-johan+linaro@kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 4ca2bbe3f684b05d7e9c92236a9617e10d3ea805
Author: Johan Hovold <johan+linaro@kernel.org>
Date:   Tue Apr 4 09:25:14 2023 +0200

    USB: dwc3: fix runtime pm imbalance on probe errors

    commit 9a8ad10c9f2e0925ff26308ec6756b93fc2f4977 upstream.

    Make sure not to suspend the device when probe fails to avoid disabling
    clocks and phys multiple times.

    Fixes: 328082376aea ("usb: dwc3: fix runtime PM in error path")
    Cc: stable@vger.kernel.org      # 4.8
    Cc: Roger Quadros <rogerq@ti.com>
    Acked-by: Thinh Nguyen <Thinh.Nguyen@synopsys.com>
    Signed-off-by: Johan Hovold <johan+linaro@kernel.org>
    Link: https://lore.kernel.org/r/20230404072524.19014-2-johan+linaro@kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3713276ae7b0af6263b5bddfa3a9802640bf9938
Author: Wesley Cheng <quic_wcheng@quicinc.com>
Date:   Thu Apr 13 12:57:40 2023 -0700

    usb: dwc3: gadget: Stall and restart EP0 if host is unresponsive

    commit 02435a739b81ae24aff5d6e930efef9458e2af3c upstream.

    It was observed that there are hosts that may complete pending SETUP
    transactions before the stop active transfers and controller halt occurs,
    leading to lingering endxfer commands on DEPs on subsequent pullup/gadget
    start iterations.

      dwc3_gadget_ep_disable   name=ep8in flags=0x3009  direction=1
      dwc3_gadget_ep_disable   name=ep4in flags=1  direction=1
      dwc3_gadget_ep_disable   name=ep3out flags=1  direction=0
```

```
        usb_gadget_disconnect   deactivated=0  connected=0  ret=0

    The sequence shows that the USB gadget disconnect (dwc3_gadget_pullup(0))
    routine completed successfully, allowing for the USB gadget to proceed with
    a USB gadget connect.  However, if this occurs the system runs into an
    issue where:

      BUG: spinlock already unlocked on CPU
      spin_bug+0x0
      dwc3_remove_requests+0x278
      dwc3_ep0_out_start+0xb0
      __dwc3_gadget_start+0x25c

    This is due to the pending endxfers, leading to gadget start (w/o lock
    held) to execute the remove requests, which will unlock the dwc3
    spinlock as part of giveback.

    To mitigate this, resolve the pending endxfers on the pullup disable
    path by re-locating the SETUP phase check after stop active transfers, since
    that is where the DWC3_EP_DELAY_STOP is potentially set.  This also allows
    for handling of a host that may be unresponsive by using the completion
    timeout to trigger the stall and restart for EP0.

    Fixes: c96683798e27 ("usb: dwc3: ep0: Don't prepare beyond Setup stage")
    Cc: stable@vger.kernel.org
    Acked-by: Thinh Nguyen <Thinh.Nguyen@synopsys.com>
    Signed-off-by: Wesley Cheng <quic_wcheng@quicinc.com>
    Link: https://lore.kernel.org/r/20230413195742.11821-2-quic_wcheng@quicinc.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f58a3daeb81583703553a5c8ac2dcddc63fed0fe
Author: Badhri Jagan Sridharan <badhri@google.com>
Date:   Fri Apr 7 03:07:41 2023 +0000

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

    commit a3afbf5cc887fc3401f012fe629810998ed61859 upstream.

    usb_gadget_connect calls gadget->ops->pullup without checking whether
    gadget->connected was previously set. Make this symmetric to
    usb_gadget_disconnect by returning early if gadget->connected is
    already set.

    Fixes: 5a1da544e572 ("usb: gadget: core: do not try to disconnect gadget if it is not
connected")
    Cc: stable@vger.kernel.org
    Signed-off-by: Badhri Jagan Sridharan <badhri@google.com>
    Link: https://lore.kernel.org/r/20230407030741.3163220-2-badhri@google.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3852f56ad02d02ce38543752a554c0383cb1593c
Author: Badhri Jagan Sridharan <badhri@google.com>
Date:   Fri Apr 7 03:07:40 2023 +0000

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

    commit 0db213ea8eed5534a5169e807f28103cbc9d23df upstream.

    usb_udc_connect_control does not check to see if the udc has already
    been started. This causes gadget->ops->pullup to be called through
    usb_gadget_connect when invoked from usb_udc_vbus_handler even before
    usb_gadget_udc_start is called. Guard this by checking for udc->started
    in usb_udc_connect_control before invoking usb_gadget_connect.

    Guarding udc->vbus, udc->started, gadget->connect, gadget->deactivate
```

```
    related functions with connect_lock. usb_gadget_connect_locked,
    usb_gadget_disconnect_locked, usb_udc_connect_control_locked,
    usb_gadget_udc_start_locked, usb_gadget_udc_stop_locked are called with
    this lock held as they can be simulataneously invoked from different code
    paths.

    Adding an additional check to make sure udc is started(udc->started)
    before pullup callback is invoked.

    Fixes: 628ef0d273a6 ("usb: udc: add usb_udc_vbus_handler")
    Cc: stable@vger.kernel.org
    Signed-off-by: Badhri Jagan Sridharan <badhri@google.com>
    Link: https://lore.kernel.org/r/20230407030741.3163220-1-badhri@google.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b252702f6257d73e3d0a7065dd930f5dc34a8af0
Author: Sascha Hauer <s.hauer@pengutronix.de>
Date:   Mon Apr 17 16:03:55 2023 +0200

    wifi: rtw88: usb: fix priority queue to endpoint mapping

    commit a6f187f92bcc2b17821538b4a11d61764e68b091 upstream.

    The RTW88 chipsets have four different priority queues in hardware. For
    the USB type chipsets the packets destined for a specific priority queue
    must be sent through the endpoint corresponding to the queue. This was
    not fully understood when porting from the RTW88 USB out of tree driver
    and thus violated.

    This patch implements the qsel to endpoint mapping as in
    get_usb_bulkout_id_88xx() in the downstream driver.

    Without this the driver often issues "timed out to flush queue 3"
    warnings and often TX stalls completely.

    Signed-off-by: Sascha Hauer <s.hauer@pengutronix.de>
    Tested-by: ValdikSS <iam@valdikss.org.ru>
    Tested-by: Alexandru gagniuc <mr.nuke.me@gmail.com>
    Tested-by: Larry Finger <Larry.Finger@lwfinger.net>
    Cc: stable@vger.kernel.org
    Reviewed-by: Ping-Ke Shih <pkshih@realtek.com>
    Signed-off-by: Kalle Valo <kvalo@kernel.org>
    Link: https://lore.kernel.org/r/20230417140358.2240429-2-s.hauer@pengutronix.de
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 9ff59a907c48cb181e7440ef05fdebac3d5b8c1a
Author: Randy Dunlap <rdunlap@infradead.org>
Date:   Thu Feb 23 19:27:03 2023 -0800

    IMA: allow/fix UML builds

    commit 644f17412f5acf01a19af9d04a921937a2bc86c6 upstream.

    UML supports HAS_IOMEM since 0bbadafdc49d (um: allow disabling
    NO_IOMEM).

    Current IMA build on UML fails on allmodconfig (with TCG_TPM=m):

    ld: security/integrity/ima/ima_queue.o: in function `ima_add_template_entry':
    ima_queue.c:(.text+0x2d9): undefined reference to `tpm_pcr_extend'
    ld: security/integrity/ima/ima_init.o: in function `ima_init':
    ima_init.c:(.init.text+0x43f): undefined reference to `tpm_default_chip'
    ld: security/integrity/ima/ima_crypto.o: in function `ima_calc_boot_aggregate_tfm':
    ima_crypto.c:(.text+0x1044): undefined reference to `tpm_pcr_read'
    ld: ima_crypto.c:(.text+0x10d8): undefined reference to `tpm_pcr_read'
```

```
    Modify the IMA Kconfig entry so that it selects TCG_TPM if HAS_IOMEM
    is set, regardless of the UML Kconfig setting.
    This updates TCG_TPM from =m to =y and fixes the linker errors.

    Fixes: f4a0391dfa91 ("ima: fix Kconfig dependencies")
    Cc: Stable <stable@vger.kernel.org> # v5.14+
    Signed-off-by: Randy Dunlap <rdunlap@infradead.org>
    Cc: Fabio Estevam <festevam@gmail.com>
    Cc: Richard Weinberger <richard@nod.at>
    Cc: Anton Ivanov <anton.ivanov@cambridgegreys.com>
    Cc: Johannes Berg <johannes@sipsolutions.net>
    Cc: linux-um@lists.infradead.org
    Signed-off-by: Mimi Zohar <zohar@linux.ibm.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f01cb9965267fa0966cf22ef8dd3af854c333341
Author: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
Date:   Fri Mar 31 18:12:49 2023 +0300

    phy: qcom-qmp-pcie: sc8180x PCIe PHY has 2 lanes

    commit 1db6b0a4246ce708b89f5136571130b9987741d1 upstream.

    All PCIe PHYs on sc8180x platform have 2 lanes, so change the number of
    lanes to 2.

    Fixes: f839f14e24f2 ("phy: qcom-qmp: Add sc8180x PCIe support")
    Cc: stable@vger.kernel.org # 5.15
    Sgned-off-by: Dmitry Baryshkov <dmitry.baryshkov@linaro.org>
    Link: https://lore.kernel.org/r/20230331151250.4049-1-dmitry.baryshkov@linaro.org
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 2ca93bf147ded50dc2a13ac86e23cfd137439b34
Author: Slark Xiao <slark_xiao@163.com>
Date:   Fri Mar 10 18:17:15 2023 +0800

    bus: mhi: host: pci_generic: Revert "Add a secondary AT port to Telit FN990"

    commit 14853a0676c126aad836eb249499552fa9d6e85a upstream.

    This reverts commit 2d5253a096c6057bbf7caa5520856dcdf7eca8bb.
    There are 2 commits with commit message "Add a secondary AT port to Telit
    FN990":

    commit 2d5253a096c6 ("bus: mhi: host: pci_generic: Add a secondary AT port
    to Telit FN990")
    commit 479aa3b0ec2e ("bus: mhi: host: pci_generic: Add a secondary AT port
    to Telit FN990")

    This turned out to be due to the patch getting applied through different
    trees and git settled on a resolution while applying it second time. But
    the second AT port of Foxconn devices don't work in PCIe mode. So the
    second commit needs to be reverted.

    Cc: stable@vger.kernel.org # 6.2
    Fixes: 2d5253a096c6 ("bus: mhi: host: pci_generic: Add a secondary AT port to Telit
FN990")
    Signed-off-by: Slark Xiao <slark_xiao@163.com>
    Reviewed-by: Fabio Porcedda <fabio.porcedda@gmail.com>
    Reviewed-by: Manivannan Sadhasivam <mani@kernel.org>
    Link: https://lore.kernel.org/r/20230310101715.69209-1-slark_xiao@163.com
    [mani: massaged the commit message a bit, added fixes tag and CCed stable]
    Signed-off-by: Manivannan Sadhasivam <manivannan.sadhasivam@linaro.org>
```

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

commit 2e9c8529102a0f6ab2b9ff4f9620765310f3a662
Author: Manivannan Sadhasivam <mani@kernel.org>
Date:   Thu Mar 16 13:40:59 2023 +0530

    PCI: qcom: Fix the incorrect register usage in v2.7.0 config

    commit 2542e16c392508800f1d9037feee881a9c444951 upstream.

    Qcom PCIe IP version v2.7.0 and its derivatives don't contain the
    PCIE20_PARF_AXI_MSTR_WR_ADDR_HALT register. Instead, they have the new
    PCIE20_PARF_AXI_MSTR_WR_ADDR_HALT_V2 register. So fix the incorrect
    register usage which is modifying a different register.

    Also in this IP version, this register change doesn't depend on MSI
    being enabled. So remove that check also.

    Link: https://lore.kernel.org/r/20230316081117.14288-2-manivannan.sadhasivam@linaro.org
    Fixes: ed8cc3b1fc84 ("PCI: qcom: Add support for SDM845 PCIe controller")
    Signed-off-by: Manivannan Sadhasivam <manivannan.sadhasivam@linaro.org>
    Signed-off-by: Lorenzo Pieralisi <lpieralisi@kernel.org>
    Cc: <stable@vger.kernel.org> # 5.6+
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 01fa3dad41bffcb20860ea512873d1f7e53190e6
Author: Lukas Wunner <lukas@wunner.de>
Date:   Tue Apr 11 08:21:02 2023 +0200

    PCI: pciehp: Fix AB-BA deadlock between reset_lock and device_lock

    commit f5eff5591b8f9c5effd25c92c758a127765f74c1 upstream.

    In 2013, commits

      2e35afaefe64 ("PCI: pciehp: Add reset_slot() method")
      608c388122c7 ("PCI: Add slot reset option to pci_dev_reset()")

    amended PCIe hotplug to mask Presence Detect Changed events during a
    Secondary Bus Reset.  The reset thus no longer causes gratuitous slot
    bringdown and bringup.

    However the commits neglected to serialize reset with code paths reading
    slot registers.  For instance, a slot bringup due to an earlier hotplug
    event may see the Presence Detect State bit cleared during a concurrent
    Secondary Bus Reset.

    In 2018, commit

      5b3f7b7d062b ("PCI: pciehp: Avoid slot access during reset")

    retrofitted the missing locking.  It introduced a reset_lock which
    serializes a Secondary Bus Reset with other parts of pciehp.

    Unfortunately the locking turns out to be overzealous:  reset_lock is
    held for the entire enumeration and de-enumeration of hotplugged devices,
    including driver binding and unbinding.

    Driver binding and unbinding acquires device_lock while the reset_lock
    of the ancestral hotplug port is held.  A concurrent Secondary Bus Reset
    acquires the ancestral reset_lock while already holding the device_lock.
    The asymmetric locking order in the two code paths can lead to AB-BA
    deadlocks.

    Michael Haeuptle reports such deadlocks on simultaneous hot-removal and
```

vfio release (the latter implies a Secondary Bus Reset):

```
  pciehp_ist()                                      # down_read(reset_lock)
    pciehp_handle_presence_or_link_change()
      pciehp_disable_slot()
        __pciehp_disable_slot()
          remove_board()
            pciehp_unconfigure_device()
              pci_stop_and_remove_bus_device()
                pci_stop_bus_device()
                  pci_stop_dev()
                    device_release_driver()
                      device_release_driver_internal()
                        __device_driver_lock()    # device_lock()

  SYS_munmap()
    vfio_device_fops_release()
      vfio_device_group_close()
        vfio_device_close()
          vfio_device_last_close()
            vfio_pci_core_close_device()
              vfio_pci_core_disable()             # device_lock()
                __pci_reset_function_locked()
                  pci_reset_bus_function()
                    pci_dev_reset_slot_function()
                      pci_reset_hotplug_slot()
                        pciehp_reset_slot()       # down_write(reset_lock)
```

Ian May reports the same deadlock on simultaneous hot-removal and an AER-induced Secondary Bus Reset:

```
  aer_recover_work_func()
    pcie_do_recovery()
      aer_root_reset()
        pci_bus_error_reset()
          pci_slot_reset()
            pci_slot_lock()                       # device_lock()
            pci_reset_hotplug_slot()
              pciehp_reset_slot()                 # down_write(reset_lock)
```

Fix by releasing the reset_lock during driver binding and unbinding, thereby splitting and shrinking the critical section.

Driver binding and unbinding is protected by the device_lock() and thus serialized with a Secondary Bus Reset. There's no need to additionally protect it with the reset_lock. However, pciehp does not bind and unbind devices directly, but rather invokes PCI core functions which also perform certain enumeration and de-enumeration steps.

The reset_lock's purpose is to protect slot registers, not enumeration and de-enumeration of hotplugged devices. That would arguably be the job of the PCI core, not the PCIe hotplug driver. After all, an AER-induced Secondary Bus Reset may as well happen during boot-time enumeration of the PCI hierarchy and there's no locking to prevent that either.

Exempting *de-enumeration* from the reset_lock is relatively harmless: A concurrent Secondary Bus Reset may foil config space accesses such as PME interrupt disablement. But if the device is physically gone, those accesses are pointless anyway. If the device is physically present and only logically removed through an Attention Button press or the sysfs "power" attribute, PME interrupts as well as DMA cannot come through because pciehp_unconfigure_device() disables INTx and Bus Master bits. That's still protected by the reset_lock in the present commit.

```
    Exempting *enumeration* from the reset_lock also has limited impact:
    The exempted call to pci_bus_add_device() may perform device accesses
    through pcibios_bus_add_device() and pci_fixup_device() which are now
    no longer protected from a concurrent Secondary Bus Reset.  Otherwise
    there should be no impact.

    In essence, the present commit seeks to fix the AB-BA deadlocks while
    still retaining a best-effort reset protection for enumeration and
    de-enumeration of hotplugged devices -- until a general solution is
    implemented in the PCI core.

    Link: https://lore.kernel.org/linux-
pci/CS1PR8401MB0728FC6FDAB8A35C22BD90EC95F10@CS1PR8401MB0728.NAMPRD84.PROD.OUTLOOK.COM
    Link: https://lore.kernel.org/linux-pci/20200615143250.438252-1-ian.may@canonical.com
    Link: https://lore.kernel.org/linux-pci/ce878dab-c0c4-5bd0-a725-9805a075682d@amd.com
    Link: https://lore.kernel.org/linux-pci/ed831249-384a-6d35-0831-70af191e9bce@huawei.com
    Link: https://bugzilla.kernel.org/show_bug.cgi?id=215590
    Fixes: 5b3f7b7d062b ("PCI: pciehp: Avoid slot access during reset")
    Link:
https://lore.kernel.org/r/fef2b2e9edf245c049a8c5b94743c0f74ff5008a.1681191902.git.lukas@wunn
er.de
    Reported-by: Michael Haeuptle <michael.haeuptle@hpe.com>
    Reported-by: Ian May <ian.may@canonical.com>
    Reported-by: Andrey Grodzovsky <andrey2805@gmail.com>
    Reported-by: Rahul Kumar <rahul.kumar1@amd.com>
    Reported-by: Jialin Zhang <zhangjialin11@huawei.com>
    Tested-by: Anatoli Antonovitch <Anatoli.Antonovitch@amd.com>
    Signed-off-by: Lukas Wunner <lukas@wunner.de>
    Signed-off-by: Bjorn Helgaas <bhelgaas@google.com>
    Cc: stable@vger.kernel.org # v4.19+
    Cc: Dan Stein <dstein@hpe.com>
    Cc: Ashok Raj <ashok.raj@intel.com>
    Cc: Alex Michon <amichon@kalrayinc.com>
    Cc: Xiongfeng Wang <wangxiongfeng2@huawei.com>
    Cc: Alex Williamson <alex.williamson@redhat.com>
    Cc: Mika Westerberg <mika.westerberg@linux.intel.com>
    Cc: Sathyanarayanan Kuppuswamy <sathyanarayanan.kuppuswamy@linux.intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 7bcabb104a798162aaa64f2664087699f0f0404b
Author: Josh Triplett <josh@joshtriplett.org>
Date:   Sun Nov 13 23:23:26 2022 -0800

    PCI: kirin: Select REGMAP_MMIO

    commit 3a2776e8a0e156a61f5b59ae341d8fffc730b962 upstream.

    pcie-kirin uses regmaps, and needs to pull them in; otherwise, with
    CONFIG_PCIE_KIRIN=y and without CONFIG_REGMAP_MMIO pcie-kirin produces
    a linker failure looking for __devm_regmap_init_mmio_clk().

    Fixes: d19afe7be126 ("PCI: kirin: Use regmap for APB registers")
    Link:
https://lore.kernel.org/r/04636141da1d6d592174eefb56760511468d035d.1668410580.git.josh@josht
riplett.org
    Signed-off-by: Josh Triplett <josh@joshtriplett.org>
    [lpieralisi@kernel.org: commit log and removed REGMAP select]
    Signed-off-by: Lorenzo Pieralisi <lpieralisi@kernel.org>
    Cc: stable@vger.kernel.org # 5.16+
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c1c92ed22c91dc039009365de5086282ef79f219
Author: Nicholas Piggin <npiggin@gmail.com>
Date:   Fri Apr 7 14:09:09 2023 +1000
```

```
    powerpc/boot: Fix boot wrapper code generation with CONFIG_POWER10_CPU

    commit 648a1783fe2551f5a091c9a5f8f463cb2cbf8745 upstream.

    -mcpu=power10 will generate prefixed and pcrel code by default, which
    we do not support. The general kernel disables these with cflags, but
    those were missed for the boot wrapper.

    Fixes: 4b2a9315f20d ("powerpc/64s: POWER10 CPU Kconfig build option")
    Cc: stable@vger.kernel.org # v6.1+
    Reported-by: Danny Tsen <dtsen@linux.ibm.com>
    Signed-off-by: Nicholas Piggin <npiggin@gmail.com>
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Link: https://msgid.link/20230407040909.230998-1-npiggin@gmail.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 47e67a57347e47e3026d0f2f55072733d4075b0c
Author: Syed Saba Kareem <Syed.SabaKareem@amd.com>
Date:   Fri Mar 31 10:50:23 2023 +0530

    ASoC: amd: ps: update the acp clock source.

    commit a4d432e9132c0b29d857b09ca2ec4c1f455b5948 upstream.

    Updating the clock source from ACLK to default clock

    Signed-off-by: Syed Saba Kareem <Syed.SabaKareem@amd.com>
    Link: https://lore.kernel.org/r/20230331052102.2211115-1-Syed.SabaKareem@amd.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Cc: Mario Limonciello <mario.limonciello@amd.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 5bcc0ffeeddd52684091e5b77a06ddcec3cdaff5
Author: Mario Limonciello <mario.limonciello@amd.com>
Date:   Wed Mar 29 14:57:58 2023 -0500

    wifi: mt76: mt7921e: Set memory space enable in PCI_COMMAND if unset

    commit 09d4d6da1b65d09414e7bce61459593f3c80ead1 upstream.

    When the BIOS has been configured for Fast Boot, systems with mt7921e
    have non-functional wifi.  Turning on Fast boot caused both bus master
    enable and memory space enable bits in PCI_COMMAND not to get configured.

    The mt7921 driver already sets bus master enable, but explicitly check
    and set memory access enable as well to fix this problem.

    Tested-by: Anson Tsao <anson.tsao@amd.com>
    Signed-off-by: Mario Limonciello <mario.limonciello@amd.com>
    Acked-by: Sean Wang <sean.wang@mediatek.com>
    Signed-off-by: Felix Fietkau <nbd@nbd.name>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```