```
commit 67f65fb8318a56a3d6b9cd966d76680c1bdc07b0
Author: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
Date:   Thu Sep 17 13:55:47 2020 +0200

    Linux 5.8.10

    Tested-by: Jon Hunter <jonathanh@nvidia.com>
    Tested-by: Linux Kernel Functional Testing <lkft@linaro.org>
    Tested-by: Guenter Roeck <linux@roeck-us.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 08cb27d67d534bf19edfd2e343caaca79762a88b
Author: Peter Oberparleiter <oberpar@linux.ibm.com>
Date:   Thu Sep 10 14:52:01 2020 +0200

    gcov: add support for GCC 10.1

    [ Upstream commit 40249c6962075c040fd071339acae524f18bfac9 ]

    Using gcov to collect coverage data for kernels compiled with GCC 10.1
    causes random malfunctions and kernel crashes.  This is the result of a
    changed GCOV_COUNTERS value in GCC 10.1 that causes a mismatch between
    the layout of the gcov_info structure created by GCC profiling code and
    the related structure used by the kernel.

    Fix this by updating the in-kernel GCOV_COUNTERS value.  Also re-enable
    config GCOV_KERNEL for use with GCC 10.

    Reported-by: Colin Ian King <colin.king@canonical.com>
    Reported-by: Leon Romanovsky <leonro@nvidia.com>
    Signed-off-by: Peter Oberparleiter <oberpar@linux.ibm.com>
    Tested-by: Leon Romanovsky <leonro@nvidia.com>
    Tested-and-Acked-by: Colin Ian King <colin.king@canonical.com>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 10cb6c2e406d46421f3bd6c7621083e6bca59aea
Author: Rob Clark <robdclark@chromium.org>
Date:   Mon Aug 17 09:23:09 2020 -0700

    drm/msm/gpu: make ringbuffer readonly

    [ Upstream commit 352c83fb39cae3eff95a8e1ed23006291abb6196 ]

    The GPU has no business writing into the ringbuffer, let's make it
    readonly to the GPU.

    Fixes: 7198e6b03155 ("drm/msm: add a3xx gpu support")
    Signed-off-by: Rob Clark <robdclark@chromium.org>
    Reviewed-by: Jordan Crouse <jcrouse@codeaurora.org>
    Signed-off-by: Rob Clark <robdclark@chromium.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a6915129ae9769a58415c6dcdf6db9650d170cd5
Author: Utkarsh Patel <utkarsh.h.patel@intel.com>
Date:   Mon Sep 7 17:21:52 2020 +0300

    usb: typec: intel_pmc_mux: Do not configure SBU and HSL Orientation in Alternate modes

    commit 7c6bbdf086ac7f1374bcf1ef0994b15109ecaf48 upstream.

    According to the PMC Type C Subsystem (TCSS) Mux programming guide rev
    0.7, bits 4 and 5 are reserved in Alternate modes.
    SBU Orientation and HSL Orientation needs to be configured only during
```

```
    initial cable detection in USB connect flow based on device property of
    "sbu-orientation" and "hsl-orientation".
    Configuring these reserved bits in the Alternate modes may result in delay
    in display link training or some unexpected behaviour.
    So do not configure them while issuing Alternate Mode requests.

    Fixes: ff4a30d5e243 ("usb: typec: mux: intel_pmc_mux: Support for static SBU/HSL
orientation")
    Signed-off-by: Utkarsh Patel <utkarsh.h.patel@intel.com>
    Cc: stable <stable@vger.kernel.org>
    Signed-off-by: Heikki Krogerus <heikki.krogerus@linux.intel.com>
    Link: https://lore.kernel.org/r/20200907142152.35678-3-heikki.krogerus@linux.intel.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 41e3571c02ec070c4428ee3d7f932167e3d5948b
Author: Utkarsh Patel <utkarsh.h.patel@intel.com>
Date:   Mon Sep 7 17:21:51 2020 +0300

    usb: typec: intel_pmc_mux: Do not configure Altmode HPD High

    commit 294955fd43dbf1e8f3a84cffa4797c6f22badc31 upstream.

    According to the PMC Type C Subsystem (TCSS) Mux programming guide rev
    0.7, bit 14 is reserved in Alternate mode.
    In DP Alternate Mode state, if the HPD_STATE (bit 7) field in the
    status update command VDO is set to HPD_HIGH, HPD is configured via
    separate HPD mode request after configuring DP Alternate mode request.
    Configuring reserved bit may show unexpected behaviour.
    So do not configure them while issuing the Alternate Mode request.

    Fixes: 7990be48ef4d ("usb: typec: mux: intel: Handle alt mode HPD_HIGH")
    Cc: stable@vger.kernel.org
    Signed-off-by: Utkarsh Patel <utkarsh.h.patel@intel.com>
    Signed-off-by: Heikki Krogerus <heikki.krogerus@linux.intel.com>
    Link: https://lore.kernel.org/r/20200907142152.35678-2-heikki.krogerus@linux.intel.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit efec148aa4eb407b9e4d97adf473a3e847a43452
Author: Madhusudanarao Amara <madhusudanarao.amara@intel.com>
Date:   Wed Aug 26 00:08:11 2020 +0530

    usb: typec: intel_pmc_mux: Un-register the USB role switch

    commit 290a405ce318d036666c4155d5899eb8cd6e0d97 upstream.

    Added missing code for un-register USB role switch in the remove and
    error path.

    Cc: Stable <stable@vger.kernel.org> # v5.8
    Reviewed-by: Heikki Krogerus <heikki.krogerus@linux.intel.com>
    Fixes: 6701adfa9693b ("usb: typec: driver for Intel PMC mux control")
    Signed-off-by: Madhusudanarao Amara <madhusudanarao.amara@intel.com>
    Link: https://lore.kernel.org/r/20200825183811.7262-1-madhusudanarao.amara@intel.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit fd23e24737de62965fadac69df695df00bac9afa
Author: Heikki Krogerus <heikki.krogerus@linux.intel.com>
Date:   Fri Sep 4 14:09:18 2020 +0300

    usb: typec: ucsi: acpi: Check the _DEP dependencies

    commit 1f3546ff3f0a1000971daef58406954bad3f7061 upstream.

    Failing probe with -EPROBE_DEFER until all dependencies
    listed in the _DEP (Operation Region Dependencies) object
```

```
    have been met.

    This will fix an issue where on some platforms UCSI ACPI
    driver fails to probe because the address space handler for
    the operation region that the UCSI ACPI interface uses has
    not been loaded yet.

    Fixes: 8243edf44152 ("usb: typec: ucsi: Add ACPI driver")
    Cc: stable@vger.kernel.org
    Signed-off-by: Heikki Krogerus <heikki.krogerus@linux.intel.com>
    Link: https://lore.kernel.org/r/20200904110918.51546-1-heikki.krogerus@linux.intel.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 5f98bc578de970a4c669adf715c1070c6c81a24a
Author: Mathias Nyman <mathias.nyman@linux.intel.com>
Date:   Tue Sep 1 11:25:28 2020 +0300

    usb: Fix out of sync data toggle if a configured device is reconfigured

    commit cfd54fa83a5068b61b7eb28d3c117d8354c74c7a upstream.

    Userspace drivers that use a SetConfiguration() request to "lightweight"
    reset an already configured usb device might cause data toggles to get out
    of sync between the device and host, and the device becomes unusable.

    The xHCI host requires endpoints to be dropped and added back to reset the
    toggle. If USB core notices the new configuration is the same as the
    current active configuration it will avoid these extra steps by calling
    usb_reset_configuration() instead of usb_set_configuration().

    A SetConfiguration() request will reset the device side data toggles.
    Make sure usb_reset_configuration() function also drops and adds back the
    endpoints to ensure data toggles are in sync.

    To avoid code duplication split the current usb_disable_device() function
    and reuse the endpoint specific part.

    Cc: stable <stable@vger.kernel.org>
    Tested-by: Martin Thierer <mthierer@gmail.com>
    Signed-off-by: Mathias Nyman <mathias.nyman@linux.intel.com>
    Link: https://lore.kernel.org/r/20200901082528.12557-1-mathias.nyman@linux.intel.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a55d0aa3a917f56999e84f91d950bdb7b4d0be8d
Author: Aleksander Morgado <aleksander@aleksander.es>
Date:   Sat Aug 29 11:05:39 2020 +0200

    USB: serial: option: add support for SIM7070/SIM7080/SIM7090 modules

    commit 1ac698790819b83f39fd7ea4f6cdabee9bdd7b38 upstream.

    These modules have 2 different USB layouts:

    The default layout with PID 0x9205 (AT+CUSBSELNV=1) exposes 4 TTYs and
    an ECM interface:

      T:  Bus=02 Lev=01 Prnt=01 Port=02 Cnt=01 Dev#=  6 Spd=480 MxCh= 0
      D:  Ver= 2.00 Cls=ef(misc ) Sub=02 Prot=01 MxPS=64 #Cfgs=  1
      P:  Vendor=1e0e ProdID=9205 Rev=00.00
      S:  Manufacturer=SimTech, Incorporated
      S:  Product=SimTech SIM7080
      S:  SerialNumber=1234567890ABCDEF
      C:  #Ifs= 6 Cfg#= 1 Atr=e0 MxPwr=500mA
      I:  If#=0x0 Alt= 0 #EPs= 2 Cls=ff(vend.) Sub=ff Prot=ff Driver=option
      I:  If#=0x1 Alt= 0 #EPs= 2 Cls=ff(vend.) Sub=ff Prot=ff Driver=option
```

```
        I:  If#=0x2 Alt= 0 #EPs= 2 Cls=ff(vend.) Sub=ff Prot=ff Driver=option
        I:  If#=0x3 Alt= 0 #EPs= 3 Cls=ff(vend.) Sub=ff Prot=ff Driver=option
        I:  If#=0x4 Alt= 0 #EPs= 1 Cls=02(commc) Sub=06 Prot=00 Driver=cdc_ether
        I:  If#=0x5 Alt= 1 #EPs= 2 Cls=0a(data ) Sub=00 Prot=00 Driver=cdc_ether

      The purpose of each TTY is as follows:
       * ttyUSB0: DIAG/QCDM port.
       * ttyUSB1: GNSS data.
       * ttyUSB2: AT-capable port (control).
       * ttyUSB3: AT-capable port (data).

      In the secondary layout with PID=0x9206 (AT+CUSBSELNV=86) the module
      exposes 6 TTY ports:

        T:  Bus=02 Lev=01 Prnt=01 Port=02 Cnt=01 Dev#=  8 Spd=480 MxCh= 0
        D:  Ver= 2.00 Cls=02(commc) Sub=00 Prot=00 MxPS=64 #Cfgs=  1
        P:  Vendor=1e0e ProdID=9206 Rev=00.00
        S:  Manufacturer=SimTech, Incorporated
        S:  Product=SimTech SIM7080
        S:  SerialNumber=1234567890ABCDEF
        C:  #Ifs= 6 Cfg#= 1 Atr=e0 MxPwr=500mA
        I:  If#=0x0 Alt= 0 #EPs= 2 Cls=ff(vend.) Sub=ff Prot=ff Driver=option
        I:  If#=0x1 Alt= 0 #EPs= 2 Cls=ff(vend.) Sub=ff Prot=ff Driver=option
        I:  If#=0x2 Alt= 0 #EPs= 2 Cls=ff(vend.) Sub=ff Prot=ff Driver=option
        I:  If#=0x3 Alt= 0 #EPs= 2 Cls=ff(vend.) Sub=ff Prot=ff Driver=option
        I:  If#=0x4 Alt= 0 #EPs= 2 Cls=ff(vend.) Sub=ff Prot=ff Driver=option
        I:  If#=0x5 Alt= 0 #EPs= 3 Cls=ff(vend.) Sub=ff Prot=ff Driver=option

      The purpose of each TTY is as follows:
       * ttyUSB0: DIAG/QCDM port.
       * ttyUSB1: GNSS data.
       * ttyUSB2: AT-capable port (control).
       * ttyUSB3: QFLOG interface.
       * ttyUSB4: DAM interface.
       * ttyUSB5: AT-capable port (data).

      Signed-off-by: Aleksander Morgado <aleksander@aleksander.es>
      Cc: stable@vger.kernel.org
      Signed-off-by: Johan Hovold <johan@kernel.org>
      Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

  commit 3c04f365d1f9415db6f5f13b1c2514d31b577d39
  Author: Bjørn Mork <bjorn@mork.no>
  Date:   Sat Aug 29 15:42:50 2020 +0200

      USB: serial: option: support dynamic Quectel USB compositions

      commit 2bb70f0a4b238323e4e2f392fc3ddeb5b7208c9e upstream.

      The USB composition, defining the set of exported functions, is dynamic
      in newer Quectel modems.  Default functions can be disabled and
      alternative functions can be enabled instead.  The alternatives
      includes class functions using interface pairs, which should be
      handled by the respective class drivers.

      Active interfaces are numbered consecutively, so static
      blacklisting based on interface numbers will fail when the
      composition changes.  An example of such an error, where the
      option driver has bound to the CDC ECM data interface,
      preventing cdc_ether from handling this function:

       T: Bus=01 Lev=01 Prnt=01 Port=00 Cnt=01 Dev#= 2 Spd=480 MxCh= 0
       D: Ver= 2.00 Cls=ef(misc ) Sub=02 Prot=01 MxPS=64 #Cfgs= 1
       P: Vendor=2c7c ProdID=0125 Rev= 3.18
       S: Manufacturer=Quectel
```

```
 S: Product=EC25-AF
 C:* #Ifs= 6 Cfg#= 1 Atr=a0 MxPwr=500mA
 A: FirstIf#= 4 IfCount= 2 Cls=02(comm.) Sub=06 Prot=00
 I:* If#= 0 Alt= 0 #EPs= 2 Cls=ff(vend.) Sub=ff Prot=ff Driver=option
 E: Ad=81(I) Atr=02(Bulk) MxPS= 512 Ivl=0ms
 E: Ad=01(O) Atr=02(Bulk) MxPS= 512 Ivl=0ms
 I:* If#= 1 Alt= 0 #EPs= 3 Cls=ff(vend.) Sub=00 Prot=00 Driver=option
 E: Ad=83(I) Atr=03(Int.) MxPS= 10 Ivl=32ms
 E: Ad=82(I) Atr=02(Bulk) MxPS= 512 Ivl=0ms
 E: Ad=02(O) Atr=02(Bulk) MxPS= 512 Ivl=0ms
 I:* If#= 2 Alt= 0 #EPs= 3 Cls=ff(vend.) Sub=00 Prot=00 Driver=option
 E: Ad=85(I) Atr=03(Int.) MxPS= 10 Ivl=32ms
 E: Ad=84(I) Atr=02(Bulk) MxPS= 512 Ivl=0ms
 E: Ad=03(O) Atr=02(Bulk) MxPS= 512 Ivl=0ms
 I:* If#= 3 Alt= 0 #EPs= 3 Cls=ff(vend.) Sub=00 Prot=00 Driver=option
 E: Ad=87(I) Atr=03(Int.) MxPS= 10 Ivl=32ms
 E: Ad=86(I) Atr=02(Bulk) MxPS= 512 Ivl=0ms
 E: Ad=04(O) Atr=02(Bulk) MxPS= 512 Ivl=0ms
 I:* If#= 4 Alt= 0 #EPs= 1 Cls=02(comm.) Sub=06 Prot=00 Driver=(none)
 E: Ad=89(I) Atr=03(Int.) MxPS= 16 Ivl=32ms
 I:* If#= 5 Alt= 0 #EPs= 0 Cls=0a(data ) Sub=00 Prot=00 Driver=option
 I: If#= 5 Alt= 1 #EPs= 2 Cls=0a(data ) Sub=00 Prot=00 Driver=option
 E: Ad=88(I) Atr=02(Bulk) MxPS= 512 Ivl=0ms
 E: Ad=05(O) Atr=02(Bulk) MxPS= 512 Ivl=0ms

Another device with the same id gets correct drivers, since the
interface of the network function happens to be blacklisted by option:

 T: Bus=01 Lev=02 Prnt=02 Port=01 Cnt=01 Dev#= 3 Spd=480 MxCh= 0
 D: Ver= 2.00 Cls=ef(misc ) Sub=02 Prot=01 MxPS=64 #Cfgs= 1
 P: Vendor=2c7c ProdID=0125 Rev= 3.18
 S: Manufacturer=Android
 S: Product=Android
 C:* #Ifs= 5 Cfg#= 1 Atr=a0 MxPwr=500mA
 I:* If#= 0 Alt= 0 #EPs= 2 Cls=ff(vend.) Sub=ff Prot=ff Driver=option
 E: Ad=81(I) Atr=02(Bulk) MxPS= 512 Ivl=0ms
 E: Ad=01(O) Atr=02(Bulk) MxPS= 512 Ivl=0ms
 I:* If#= 1 Alt= 0 #EPs= 3 Cls=ff(vend.) Sub=00 Prot=00 Driver=option
 E: Ad=83(I) Atr=03(Int.) MxPS= 10 Ivl=32ms
 E: Ad=82(I) Atr=02(Bulk) MxPS= 512 Ivl=0ms
 E: Ad=02(O) Atr=02(Bulk) MxPS= 512 Ivl=0ms
 I:* If#= 2 Alt= 0 #EPs= 3 Cls=ff(vend.) Sub=00 Prot=00 Driver=option
 E: Ad=85(I) Atr=03(Int.) MxPS= 10 Ivl=32ms
 E: Ad=84(I) Atr=02(Bulk) MxPS= 512 Ivl=0ms
 E: Ad=03(O) Atr=02(Bulk) MxPS= 512 Ivl=0ms
 I:* If#= 3 Alt= 0 #EPs= 3 Cls=ff(vend.) Sub=00 Prot=00 Driver=option
 E: Ad=87(I) Atr=03(Int.) MxPS= 10 Ivl=32ms
 E: Ad=86(I) Atr=02(Bulk) MxPS= 512 Ivl=0ms
 E: Ad=04(O) Atr=02(Bulk) MxPS= 512 Ivl=0ms
 I:* If#= 4 Alt= 0 #EPs= 3 Cls=ff(vend.) Sub=ff Prot=ff Driver=qmi_wwan
 E: Ad=89(I) Atr=03(Int.) MxPS= 8 Ivl=32ms
 E: Ad=88(I) Atr=02(Bulk) MxPS= 512 Ivl=0ms
 E: Ad=05(O) Atr=02(Bulk) MxPS= 512 Ivl=0ms

Change rules for EC21, EC25, BG96 and EG95 to match vendor specific
serial functions only, to prevent binding to class functions. Require
2 endpoints on ff/ff/ff functions, avoiding the 3 endpoint QMI/RMNET
network functions.

Cc: AceLan Kao <acelan.kao@canonical.com>
Cc: Sebastian Sjoholm <ssjoholm@mac.com>
Cc: Dan Williams <dcbw@redhat.com>
Cc: stable@vger.kernel.org
Signed-off-by: Bjørn Mork <bjorn@mork.no>
Signed-off-by: Johan Hovold <johan@kernel.org>
```

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

commit 64de97011bc9c8e9187dcc64d17f5e23de937bfc
Author: Patrick Riphagen <patrick.riphagen@xsens.com>
Date:   Thu Aug 6 13:55:47 2020 +0200

    USB: serial: ftdi_sio: add IDs for Xsens Mti USB converter

    commit 6ccc48e0eb2f3a5f3bd39954a21317e5f8874726 upstream.

    The device added has an FTDI chip inside.
    The device is used to connect Xsens USB Motion Trackers.

    Cc: stable@vger.kernel.org
    Signed-off-by: Patrick Riphagen <patrick.riphagen@xsens.com>
    Signed-off-by: Johan Hovold <johan@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3356a2fe9ba76a76134f32bd4e7d91ac3223b6ee
Author: Zeng Tao <prime.zeng@hisilicon.com>
Date:   Fri Sep 4 14:37:44 2020 +0800

    usb: core: fix slab-out-of-bounds Read in read_descriptors

    commit a18cd6c9b6bc73dc17e8b7e9bd07decaa8833c97 upstream.

    The USB device descriptor may get changed between two consecutive
    enumerations on the same device for some reason, such as DFU or
    malicius device.
    In that case, we may access the changing descriptor if we don't take
    the device lock here.

    The issue is reported:
    https://syzkaller.appspot.com/bug?id=901a0d9e6519ef8dc7acab25344bd287dd3c7be9

    Cc: stable <stable@vger.kernel.org>
    Cc: Alan Stern <stern@rowland.harvard.edu>
    Reported-by: syzbot+256e56ddde8b8957eabd@syzkaller.appspotmail.com
    Fixes: 217a9081d8e6 ("USB: add all configs to the "descriptors" attribute")
    Signed-off-by: Zeng Tao <prime.zeng@hisilicon.com>
    Link: https://lore.kernel.org/r/1599201467-11000-1-git-send-email-
prime.zeng@hisilicon.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 4ee2ea67a64c3ed676cd5de65aa86e389dea1b59
Author: Sivaprakash Murugesan <sivaprak@codeaurora.org>
Date:   Wed Jul 29 21:00:03 2020 +0530

    phy: qcom-qmp: Use correct values for ipq8074 PCIe Gen2 PHY init

    commit afd55e6d1bd35b4b36847869011447a83a81c8e0 upstream.

    There were some problem in ipq8074 Gen2 PCIe phy init sequence.

    1. Few register values were wrongly updated in the phy init sequence.
    2. The register QSERDES_RX_SIGDET_CNTRL is a RX tuning parameter
       register which is added in serdes table causing the wrong register
       was getting updated.
    3. Clocks and resets were not added in the phy init.

    Fix these to make Gen2 PCIe port on ipq8074 devices to work.

    Fixes: eef243d04b2b6 ("phy: qcom-qmp: Add support for IPQ8074")
    Cc: stable@vger.kernel.org
    Co-developed-by: Selvam Sathappan Periakaruppan <speriaka@codeaurora.org>
```

```
    Signed-off-by: Selvam Sathappan Periakaruppan <speriaka@codeaurora.org>
    Signed-off-by: Sivaprakash Murugesan <sivaprak@codeaurora.org>
    Link: https://lore.kernel.org/r/1596036607-11877-4-git-send-email-
sivaprak@codeaurora.org
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 97364c2546d51d72fda0bbfb9042dbe461cee5e2
Author: Vaibhav Agarwal <vaibhav.sr@gmail.com>
Date:   Fri Aug 14 18:03:15 2020 +0530

    staging: greybus: audio: fix uninitialized value issue

    commit 1dffeb8b8b4c261c45416d53c75ea51e6ece1770 upstream.

    The current implementation for gbcodec_mixer_dapm_ctl_put() uses
    uninitialized gbvalue for comparison with updated value. This was found
    using static analysis with coverity.

    Uninitialized scalar variable (UNINIT)
    11. uninit_use: Using uninitialized value
    gbvalue.value.integer_value[0].
    460        if (gbvalue.value.integer_value[0] != val) {

    This patch fixes the issue with fetching the gbvalue before using it for
        comparision.

    Fixes: 6339d2322c47 ("greybus: audio: Add topology parser for GB codec")
    Reported-by: Colin Ian King <colin.king@canonical.com>
    Signed-off-by: Vaibhav Agarwal <vaibhav.sr@gmail.com>
    Cc: stable <stable@vger.kernel.org>
    Link:
https://lore.kernel.org/r/bc4f29eb502ccf93cd2ffd98db0e319fa7d0f247.1597408126.git.vaibhav.sr
@gmail.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b56b1e392e7e7eba99c2bf6fe488ad7d1a53019c
Author: Tetsuo Handa <penguin-kernel@i-love.sakura.ne.jp>
Date:   Mon Aug 31 19:37:00 2020 +0900

    video: fbdev: fix OOB read in vga_8planes_imageblit()

    commit bd018a6a75cebb511bb55a0e7690024be975fe93 upstream.

    syzbot is reporting OOB read at vga_8planes_imageblit() [1], for
    "cdat[y] >> 4" can become a negative value due to "const char *cdat".

    [1] https://syzkaller.appspot.com/bug?id=0d7a0da1557dcd1989e00cb3692b26d4173b4132

    Reported-by: syzbot <syzbot+69fbd3e01470f169c8c4@syzkaller.appspotmail.com>
    Signed-off-by: Tetsuo Handa <penguin-kernel@I-love.SAKURA.ne.jp>
    Cc: stable <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/90b55ec3-d5b0-3307-9f7c-7ff5c5fd6ad3@i-love.sakura.ne.jp
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b2d5d5ea7c564a68440a171322fa4e6cb20aaa40
Author: Chris Healy <cphealy@gmail.com>
Date:   Fri Aug 21 14:21:02 2020 -0700

    ARM: dts: vfxxx: Add syscon compatible with OCOTP

    commit 2a6838d54128952ace6f0ca166dd8706abe46649 upstream.

    Add syscon compatibility with Vybrid OCOTP node. This is required to
    access the UID.
```

```
    Fixes: fa8d20c8dbb77 ("ARM: dts: vfxxx: Add node corresponding to OCOTP")
    Cc: stable@vger.kernel.org
    Reviewed-by: Fabio Estevam <festevam@gmail.com>
    Reviewed-by: Stefan Agner <stefan@agner.ch>
    Signed-off-by: Chris Healy <cphealy@gmail.com>
    Signed-off-by: Shawn Guo <shawnguo@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 737bfb9d464b2cdc0e10f256e4c8b19066bdae4c
Author: Robin Gong <yibin.gong@nxp.com>
Date:   Tue Sep 1 18:21:49 2020 +0800

    arm64: dts: imx8mp: correct sdma1 clk setting

    commit 66138621f2473e29625dfa6bb229872203b71b90 upstream.

    Correct sdma1 ahb clk, otherwise wrong 1:1 clk ratio will be chosed so
    that sdma1 function broken. sdma1 should use 1:2 clk, while sdma2/3 use
    1:1.

    Fixes: 6d9b8d20431f ("arm64: dts: freescale: Add i.MX8MP dtsi support")
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Robin Gong <yibin.gong@nxp.com>
    Signed-off-by: Shawn Guo <shawnguo@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit f17c2fe6ea89a007e72aed27d672d9e3c417d2b7
Author: Kees Cook <keescook@chromium.org>
Date:   Wed Sep 9 15:53:54 2020 -0700

    test_firmware: Test platform fw loading on non-EFI systems

    commit baaabecfc80fad255f866563b53b8c7a3eec176e upstream.

    On non-EFI systems, it wasn't possible to test the platform firmware
    loader because it will have never set "checked_fw" during __init.
    Instead, allow the test code to override this check. Additionally split
    the declarations into a private symbol namespace so there is greater
    enforcement of the symbol visibility.

    Fixes: 548193cba2a7 ("test_firmware: add support for firmware_request_platform")
    Cc: stable@vger.kernel.org
    Signed-off-by: Kees Cook <keescook@chromium.org>
    Acked-by: Ard Biesheuvel <ardb@kernel.org>
    Link: https://lore.kernel.org/r/20200909225354.3118328-1-keescook@chromium.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 46230cc3ff269a09ea86df8b1ea6d90db6758cf8
Author: Vladis Dronov <vdronov@redhat.com>
Date:   Tue Aug 11 17:01:29 2020 +0200

    debugfs: Fix module state check condition

    commit e3b9fc7eec55e6fdc8beeed18f2ed207086341e2 upstream.

    The '#ifdef MODULE' check in the original commit does not work as intended.
    The code under the check is not built at all if CONFIG_DEBUG_FS=y. Fix this
    by using a correct check.

    Fixes: 275678e7a9be ("debugfs: Check module state before warning in
{full/open}_proxy_open()")
    Signed-off-by: Vladis Dronov <vdronov@redhat.com>
    Cc: stable <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20200811150129.53343-1-vdronov@redhat.com
```

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

commit bf4c17b74472e5c9e9793c054af71dff62650be5
Author: Amjad Ouled-Ameur <aouledameur@baylibre.com>
Date:   Thu Aug 27 16:48:10 2020 +0200

    Revert "usb: dwc3: meson-g12a: fix shared reset control use"

    commit a6498d51821edf9615b42b968fb419a40197a982 upstream.

    This reverts commit 7a410953d1fb4dbe91ffcfdee9cbbf889d19b0d7.

    This commit breaks USB on meson-gxl-s905x-libretech-cc. Reverting
    the change solves the issue.

    In fact, according to the reset framework code, consumers must not use
    reset_control_(de)assert() on shared reset lines when reset_control_reset
    has been used, and vice-versa.

    Moreover, with this commit, usb is not guaranted to be reset since the
    reset is likely to be initially deasserted.

    Reverting the commit will bring back the suspend warning mentioned in the
    commit description. Nevertheless, a warning is much less critical than
    breaking dwc3-meson-g12a USB completely. We will address the warning
    issue in another way as a 2nd step.

    Fixes: 7a410953d1fb ("usb: dwc3: meson-g12a: fix shared reset control use")
    Cc: stable <stable@vger.kernel.org>
    Signed-off-by: Amjad Ouled-Ameur <aouledameur@baylibre.com>
    Reported-by: Jerome Brunet <jbrunet@baylibre.com>
    Acked-by: Neil Armstrong <narmstrong@baylibre.com>
    Acked-by: Philipp Zabel <p.zabel@pengutronix.de>
    Link: https://lore.kernel.org/r/20200827144810.26657-1-aouledameur@baylibre.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 68c125324b5e1d1d22805653735442923d896a1d
Author: Rustam Kovhaev <rkovhaev@gmail.com>
Date:   Mon Sep 7 11:55:35 2020 -0700

    KVM: fix memory leak in kvm_io_bus_unregister_dev()

    commit f65886606c2d3b562716de030706dfe1bea4ed5e upstream.

    when kmalloc() fails in kvm_io_bus_unregister_dev(), before removing
    the bus, we should iterate over all other devices linked to it and call
    kvm_iodevice_destructor() for them

    Fixes: 90db10434b16 ("KVM: kvm_io_bus_unregister_dev() should never fail")
    Cc: stable@vger.kernel.org
    Reported-and-tested-by: syzbot+f196caa45793d6374707@syzkaller.appspotmail.com
    Link: https://syzkaller.appspot.com/bug?extid=f196caa45793d6374707
    Signed-off-by: Rustam Kovhaev <rkovhaev@gmail.com>
    Reviewed-by: Vitaly Kuznetsov <vkuznets@redhat.com>
    Message-Id: <20200907185535.233114-1-rkovhaev@gmail.com>
    Signed-off-by: Paolo Bonzini <pbonzini@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a5ba8c6f24b178dcc0f2d0a55fa995dd8f82d292
Author: Lai Jiangshan <laijs@linux.alibaba.com>
Date:   Wed Sep 2 21:54:21 2020 +0800

    kvm x86/mmu: use KVM_REQ_MMU_SYNC to sync when needed

    commit f6f6195b888c28a0b59ceb0562daff92a2be86c3 upstream.
```

```
    When kvm_mmu_get_page() gets a page with unsynced children, the spt
    pagetable is unsynchronized with the guest pagetable. But the
    guest might not issue a "flush" operation on it when the pagetable
    entry is changed from zero or other cases. The hypervisor has the
    responsibility to synchronize the pagetables.

    KVM behaved as above for many years, But commit 8c8560b83390
    ("KVM: x86/mmu: Use KVM_REQ_TLB_FLUSH_CURRENT for MMU specific flushes")
    inadvertently included a line of code to change it without giving any
    reason in the changelog. It is clear that the commit's intention was to
    change KVM_REQ_TLB_FLUSH -> KVM_REQ_TLB_FLUSH_CURRENT, so we don't
    needlessly flush other contexts; however, one of the hunks changed
    a nearby KVM_REQ_MMU_SYNC instead.  This patch changes it back.

    Link: https://lore.kernel.org/lkml/20200320212833.3507-26-
sean.j.christopherson@intel.com/
    Cc: Sean Christopherson <sean.j.christopherson@intel.com>
    Cc: Vitaly Kuznetsov <vkuznets@redhat.com>
    Signed-off-by: Lai Jiangshan <laijs@linux.alibaba.com>
    Message-Id: <20200902135421.31158-1-jiangshanlai@gmail.com>
    fixes: 8c8560b83390 ("KVM: x86/mmu: Use KVM_REQ_TLB_FLUSH_CURRENT for MMU specific
flushes")
    Cc: stable@vger.kernel.org
    Signed-off-by: Paolo Bonzini <pbonzini@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1e1f9fd4a85224e1633770ebd0d1d76c9fa35596
Author: Marc Zyngier <maz@kernel.org>
Date:   Wed Sep 2 11:18:29 2020 +0100

    KVM: arm64: Do not try to map PUDs when they are folded into PMD

    commit 3fb884ffe921c99483a84b0175f3c03f048e9069 upstream.

    For the obscure cases where PMD and PUD are the same size
    (64kB pages with 42bit VA, for example, which results in only
    two levels of page tables), we can't map anything as a PUD,
    because there is... erm... no PUD to speak of. Everything is
    either a PMD or a PTE.

    So let's only try and map a PUD when its size is different from
    that of a PMD.

    Cc: stable@vger.kernel.org
    Fixes: b8e0ba7c8bea ("KVM: arm64: Add support for creating PUD hugepages at stage 2")
    Reported-by: Gavin Shan <gshan@redhat.com>
    Reported-by: Eric Auger <eric.auger@redhat.com>
    Reviewed-by: Alexandru Elisei <alexandru.elisei@arm.com>
    Reviewed-by: Gavin Shan <gshan@redhat.com>
    Tested-by: Gavin Shan <gshan@redhat.com>
    Tested-by: Eric Auger <eric.auger@redhat.com>
    Tested-by: Alexandru Elisei <alexandru.elisei@arm.com>
    Signed-off-by: Marc Zyngier <maz@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 209c934c9708073cdc315e8a4dc22dff9d49e691
Author: Wanpeng Li <wanpengli@tencent.com>
Date:   Wed Aug 19 16:55:27 2020 +0800

    KVM: VMX: Don't freeze guest when event delivery causes an APIC-access exit

    commit 99b82a1437cb31340dbb2c437a2923b9814a7b15 upstream.

    According to SDM 27.2.4, Event delivery causes an APIC-access VM exit.
```

```
    Don't report internal error and freeze guest when event delivery causes
    an APIC-access exit, it is handleable and the event will be re-injected
    during the next vmentry.

    Signed-off-by: Wanpeng Li <wanpengli@tencent.com>
    Message-Id: <1597827327-25055-2-git-send-email-wanpengli@tencent.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Paolo Bonzini <pbonzini@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 20782abbbdfe922496a28f9cc0c3c0030f7dfb8f
Author: Linus Torvalds <torvalds@linux-foundation.org>
Date:   Wed Sep 9 14:53:50 2020 -0700

    vgacon: remove software scrollback support

    commit 973c096f6a85e5b5f2a295126ba6928d9a6afd45 upstream.

    Yunhai Zhang recently fixed a VGA software scrollback bug in commit
    ebfdfeeae8c0 ("vgacon: Fix for missing check in scrollback handling"),
    but that then made people look more closely at some of this code, and
    there were more problems on the vgacon side, but also the fbcon software
    scrollback.

    We don't really have anybody who maintains this code - probably because
    nobody actually _uses_ it any more.  Sure, people still use both VGA and
    the framebuffer consoles, but they are no longer the main user
    interfaces to the kernel, and haven't been for decades, so these kinds
    of extra features end up bitrotting and not really being used.

    So rather than try to maintain a likely unused set of code, I'll just
    aggressively remove it, and see if anybody even notices.  Maybe there
    are people who haven't jumped on the whole GUI badnwagon yet, and think
    it's just a fad.  And maybe those people use the scrollback code.

    If that turns out to be the case, we can resurrect this again, once
    we've found the sucker^Wmaintainer for it who actually uses it.

    Reported-by: NopNop Nop <nopitydays@gmail.com>
    Tested-by: Willy Tarreau <w@1wt.eu>
    Cc: 张云海 <zhangyunhai@nsfocus.com>
    Acked-by: Andy Lutomirski <luto@amacapital.net>
    Acked-by: Willy Tarreau <w@1wt.eu>
    Reviewed-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ffa74c8e58b8f42b2d95b29443befba2e28fb260
Author: Linus Torvalds <torvalds@linux-foundation.org>
Date:   Tue Sep 8 10:56:27 2020 -0700

    fbcon: remove now unusued 'softback_lines' cursor() argument

    commit 06a0df4d1b8b13b551668e47b11fd7629033b7df upstream.

    Since the softscroll code got removed, this argument is always zero and
    makes no sense any more.

    Tested-by: Yuan Ming <yuanmingbuaa@gmail.com>
    Tested-by: Willy Tarreau <w@1wt.eu>
    Reviewed-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 245a228891e3627e47921db1ec1b6612f118158b
```

```
Author: Linus Torvalds <torvalds@linux-foundation.org>
Date:   Mon Sep 7 11:45:27 2020 -0700

    fbcon: remove soft scrollback code

    commit 50145474f6ef4a9c19205b173da6264a644c7489 upstream.

    This (and the VGA soft scrollback) turns out to have various nasty small
    special cases that nobody really is willing to fight.  The soft
    scrollback code was really useful a few decades ago when you typically
    used the console interactively as the main way to interact with the
    machine, but that just isn't the case any more.

    So it's not worth dragging along.

    Tested-by: Yuan Ming <yuanmingbuaa@gmail.com>
    Tested-by: Willy Tarreau <w@1wt.eu>
    Acked-by: Bartlomiej Zolnierkiewicz <b.zolnierkie@samsung.com>
    Acked-by: Daniel Vetter <daniel.vetter@ffwll.ch>
    Reviewed-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 20a3fda3dc88aeb566c9be9633510d3f69b72baf
Author: Mark Bloch <markb@mellanox.com>
Date:   Mon Aug 24 14:02:29 2020 +0300

    RDMA/mlx4: Read pkey table length instead of hardcoded value

    commit ec78b3bd66bc9a015505df0ef0eb153d9e64b03b upstream.

    If the pkey_table is not available (which is the case when RoCE is not
    supported), the cited commit caused a regression where mlx4_devices
    without RoCE are not created.

    Fix this by returning a pkey table length of zero in procedure
    eth_link_query_port() if the pkey-table length reported by the device is
    zero.

    Link: https://lore.kernel.org/r/20200824110229.1094376-1-leon@kernel.org
    Cc: <stable@vger.kernel.org>
    Fixes: 1901b91f9982 ("IB/core: Fix potential NULL pointer dereference in pkey cache")
    Fixes: fa417f7b520e ("IB/mlx4: Add support for IBoE")
    Signed-off-by: Mark Bloch <markb@mellanox.com>
    Reviewed-by: Maor Gottlieb <maorg@nvidia.com>
    Signed-off-by: Leon Romanovsky <leonro@nvidia.com>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ef5c5dc2c83b64871b49bdad145aefe5e85a54d2
Author: Yi Zhang <yi.zhang@redhat.com>
Date:   Thu Aug 20 23:36:46 2020 +0800

    RDMA/rxe: Fix the parent sysfs read when the interface has 15 chars

    commit 60b1af64eb35074a4f2d41cc1e503a7671e68963 upstream.

    'parent' sysfs reads will yield '\0' bytes when the interface name has 15
    chars, and there will no "\n" output.

    To reproduce, create one interface with 15 chars:

     [root@test ~]# ip a s enp0s29u1u7u3c2
     2: enp0s29u1u7u3c2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state
UNKNOWN group default qlen 1000
```

```
          link/ether 02:21:28:57:47:17 brd ff:ff:ff:ff:ff:ff
          inet6 fe80::ac41:338f:5bcd:c222/64 scope link noprefixroute
             valid_lft forever preferred_lft forever
       [root@test ~]# modprobe rdma_rxe
       [root@test ~]# echo enp0s29u1u7u3c2 > /sys/module/rdma_rxe/parameters/add
       [root@test ~]# cat /sys/class/infiniband/rxe0/parent
       enp0s29u1u7u3c2[root@test ~]#
       [root@test ~]# f="/sys/class/infiniband/rxe0/parent"
       [root@test ~]# echo "$(<"$f")"
       -bash: warning: command substitution: ignored null byte in input
       enp0s29u1u7u3c2
```

    Use scnprintf and PAGE_SIZE to fill the sysfs output buffer.

    Cc: stable@vger.kernel.org
    Fixes: 8700e3e7c485 ("Soft RoCE driver")
    Link: https://lore.kernel.org/r/20200820153646.31316-1-yi.zhang@redhat.com
    Suggested-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Yi Zhang <yi.zhang@redhat.com>
    Reviewed-by: Bart Van Assche <bvanassche@acm.org>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 81f5de838b82144fb336a71b697d28ceefc5aead
Author: Ilya Dryomov <idryomov@gmail.com>
Date: Thu Sep 3 13:24:11 2020 +0200

    rbd: require global CAP_SYS_ADMIN for mapping and unmapping

    commit f44d04e696feaf13d192d942c4f14ad2e117065a upstream.

    It turns out that currently we rely only on sysfs attribute
    permissions:

```
      $ ll /sys/bus/rbd/{add*,remove*}
      --w------- 1 root root 4096 Sep  3 20:37 /sys/bus/rbd/add
      --w------- 1 root root 4096 Sep  3 20:37 /sys/bus/rbd/add_single_major
      --w------- 1 root root 4096 Sep  3 20:37 /sys/bus/rbd/remove
      --w------- 1 root root 4096 Sep  3 20:38 /sys/bus/rbd/remove_single_major
```

    This means that images can be mapped and unmapped (i.e. block devices
    can be created and deleted) by a UID 0 process even after it drops all
    privileges or by any process with CAP_DAC_OVERRIDE in its user namespace
    as long as UID 0 is mapped into that user namespace.

    Be consistent with other virtual block devices (loop, nbd, dm, md, etc)
    and require CAP_SYS_ADMIN in the initial user namespace for mapping and
    unmapping, and also for dumping the configuration string and refreshing
    the image header.

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

commit 49947f869fdf94c011e9e9402d998a88663d2d0f
Author: James Smart <james.smart@broadcom.com>
Date: Fri Aug 28 12:01:50 2020 -0700

    nvme: Revert: Fix controller creation races with teardown flow

    commit b63de8400a6e1001b5732286cf6f5ec27799b7b4 upstream.

    The indicated patch introduced a barrier in the sysfs_delete attribute
    for the controller that rejects the request if the controller isn't

```
    created. "Created" is defined as at least 1 call to nvme_start_ctrl().

    This is problematic in error-injection testing.  If an error occurs on
    the initial attempt to create an association and the controller enters
    reconnect(s) attempts, the admin cannot delete the controller until
    either there is a successful association created or ctrl_loss_tmo
    times out.

    Where this issue is particularly hurtful is when the "admin" is the
    nvme-cli, it is performing a connection to a discovery controller, and
    it is initiated via auto-connect scripts.  With the FC transport, if the
    first connection attempt fails, the controller enters a normal reconnect
    state but returns control to the cli thread that created the controller.
    In this scenario, the cli attempts to read the discovery log via ioctl,
    which fails, causing the cli to see it as an empty log and then proceeds
    to delete the discovery controller. The delete is rejected and the
    controller is left live. If the discovery controller reconnect then
    succeeds, there is no action to delete it, and it sits live doing nothing.

    Cc: <stable@vger.kernel.org> # v5.7+
    Fixes: ce1518139e69 ("nvme: Fix controller creation races with teardown flow")
    Signed-off-by: James Smart <james.smart@broadcom.com>
    CC: Israel Rukshin <israelr@mellanox.com>
    CC: Max Gurtovoy <maxg@mellanox.com>
    CC: Christoph Hellwig <hch@lst.de>
    CC: Keith Busch <kbusch@kernel.org>
    CC: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c10bc9123c25713137297d58ae1b170599a79b45
Author: Chris Packham <chris.packham@alliedtelesis.co.nz>
Date:   Thu Sep 3 13:20:29 2020 +1200

    mmc: sdhci-of-esdhc: Don't walk device-tree on every interrupt

    commit 060522d89705f9d961ef1762dc1468645dd21fbd upstream.

    Commit b214fe592ab7 ("mmc: sdhci-of-esdhc: add erratum eSDHC7 support")
    added code to check for a specific compatible string in the device-tree
    on every esdhc interrupat. Instead of doing this record the quirk in
    struct sdhci_esdhc and lookup the struct in esdhc_irq.

    Signed-off-by: Chris Packham <chris.packham@alliedtelesis.co.nz>
    Link: https://lore.kernel.org/r/20200903012029.25673-1-chris.packham@alliedtelesis.co.nz
    Fixes: b214fe592ab7 ("mmc: sdhci-of-esdhc: add erratum eSDHC7 support")
    Cc: stable@vger.kernel.org
    Signed-off-by: Ulf Hansson <ulf.hansson@linaro.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit fe7c9fa20fc246a86c5aca1cd68c74732902ba7a
Author: Adrian Hunter <adrian.hunter@intel.com>
Date:   Thu Sep 3 11:20:07 2020 +0300

    mmc: sdio: Use mmc_pre_req() / mmc_post_req()

    commit f0c393e2104e48c8a881719a8bd37996f71b0aee upstream.

    SDHCI changed from using a tasklet to finish requests, to using an IRQ
    thread i.e. commit c07a48c2651965 ("mmc: sdhci: Remove finish_tasklet").
    Because this increased the latency to complete requests, a preparatory
    change was made to complete the request from the IRQ handler if
    possible i.e. commit 19d2f695f4e827 ("mmc: sdhci: Call mmc_request_done()
    from IRQ handler if possible").  That alleviated the situation for MMC
    block devices because the MMC block driver makes use of mmc_pre_req()
```

```
    and mmc_post_req() so that successful requests are completed in the IRQ
    handler and any DMA unmapping is handled separately in mmc_post_req().
    However SDIO was still affected, and an example has been reported with
    up to 20% degradation in performance.

    Looking at SDIO I/O helper functions, sdio_io_rw_ext_helper() appeared
    to be a possible candidate for making use of asynchronous requests
    within its I/O loops, but analysis revealed that these loops almost
    never iterate more than once, so the complexity of the change would not
    be warrented.

    Instead, mmc_pre_req() and mmc_post_req() are added before and after I/O
    submission (mmc_wait_for_req) in mmc_io_rw_extended().  This still has
    the potential benefit of reducing the duration of interrupt handlers, as
    well as addressing the latency issue for SDHCI.  It also seems a more
    reasonable solution than forcing drivers to do everything in the IRQ
    handler.

    Reported-by: Dmitry Osipenko <digetx@gmail.com>
    Fixes: c07a48c2651965 ("mmc: sdhci: Remove finish_tasklet")
    Signed-off-by: Adrian Hunter <adrian.hunter@intel.com>
    Tested-by: Dmitry Osipenko <digetx@gmail.com>
    Cc: stable@vger.kernel.org
    Link: https://lore.kernel.org/r/20200903082007.18715-1-adrian.hunter@intel.com
    Signed-off-by: Ulf Hansson <ulf.hansson@linaro.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3f4f13a91d87ea641e6502b12738860de72e49bf
Author: Jordan Crouse <jcrouse@codeaurora.org>
Date:   Thu Sep 3 20:03:13 2020 -0600

    drm/msm: Disable the RPTR shadow

    commit f6828e0c4045f03f9cf2df6c2a768102641183f4 upstream.

    Disable the RPTR shadow across all targets. It will be selectively
    re-enabled later for targets that need it.

    Cc: stable@vger.kernel.org
    Signed-off-by: Jordan Crouse <jcrouse@codeaurora.org>
    Signed-off-by: Rob Clark <robdclark@chromium.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 16f3920c6508e915a6a0b4768b338d652f3ed182
Author: Jordan Crouse <jcrouse@codeaurora.org>
Date:   Thu Sep 3 20:03:12 2020 -0600

    drm/msm: Disable preemption on all 5xx targets

    commit 7b3f3948c8b7053d771acc9f79810cc410f5e2e0 upstream.

    Temporarily disable preemption on a5xx targets pending some improvements
    to protect the RPTR shadow from being corrupted.

    Cc: stable@vger.kernel.org
    Signed-off-by: Jordan Crouse <jcrouse@codeaurora.org>
    Signed-off-by: Rob Clark <robdclark@chromium.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c657e400e7353ef387fdd611a6bcff2f92d67409
Author: Jordan Crouse <jcrouse@codeaurora.org>
Date:   Thu Sep 3 20:03:10 2020 -0600

    drm/msm: Split the a5xx preemption record
```

```
    commit 34221545d2069dc947131f42392fd4cebabe1b39 upstream.

    The main a5xx preemption record can be marked as privileged to
    protect it from user access but the counters storage needs to be
    remain unprivileged. Split the buffers and mark the critical memory
    as privileged.

    Cc: stable@vger.kernel.org
    Signed-off-by: Jordan Crouse <jcrouse@codeaurora.org>
    Signed-off-by: Rob Clark <robdclark@chromium.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 8e6a2ad4658bbde7648b4ec3daf46db2dd7dfa8b
Author: Linus Walleij <linus.walleij@linaro.org>
Date:   Thu Aug 20 22:31:44 2020 +0200

    drm/tve200: Stabilize enable/disable

    commit f71800228dc74711c3df43854ce7089562a3bc2d upstream.

    The TVE200 will occasionally print a bunch of lost interrupts
    and similar dmesg messages, sometimes during boot and sometimes
    after disabling and coming back to enablement. This is probably
    because the hardware is left in an unknown state by the boot
    loader that displays a logo.

    This can be fixed by bringing the controller into a known state
    by resetting the controller while enabling it. We retry reset 5
    times like the vendor driver does. We also put the controller
    into reset before de-clocking it and clear all interrupts before
    enabling the vblank IRQ.

    This makes the video enable/disable/enable cycle rock solid
    on the D-Link DIR-685. Tested extensively.

    Signed-off-by: Linus Walleij <linus.walleij@linaro.org>
    Acked-by: Daniel Vetter <daniel.vetter@ffwll.ch>
    Cc: stable@vger.kernel.org
    Link: https://patchwork.freedesktop.org/patch/msgid/20200820203144.271081-1-
linus.walleij@linaro.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 79e1d5d7e3d1ef645f82e09de3d9adf81c8f97e6
Author: Hou Pu <houpu@bytedance.com>
Date:   Wed Jul 29 09:03:43 2020 -0400

    scsi: target: iscsi: Fix hang in iscsit_access_np() when getting tpg->np_login_sem

    commit ed43ffea78dcc97db3f561da834f1a49c8961e33 upstream.

    The iSCSI target login thread might get stuck with the following stack:

    cat /proc/`pidof iscsi_np`/stack
    [<0>] down_interruptible+0x42/0x50
    [<0>] iscsit_access_np+0xe3/0x167
    [<0>] iscsi_target_locate_portal+0x695/0x8ac
    [<0>] __iscsi_target_login_thread+0x855/0xb82
    [<0>] iscsi_target_login_thread+0x2f/0x5a
    [<0>] kthread+0xfa/0x130
    [<0>] ret_from_fork+0x1f/0x30

    This can be reproduced via the following steps:

    1. Initiator A tries to log in to iqn1-tpg1 on port 3260. After finishing
       PDU exchange in the login thread and before the negotiation is finished
```

```
        the the network link goes down. At this point A has not finished login
        and tpg->np_login_sem is held.

     2. Initiator B tries to log in to iqn2-tpg1 on port 3260. After finishing
        PDU exchange in the login thread the target expects to process remaining
        login PDUs in workqueue context.

     3. Initiator A' tries to log in to iqn1-tpg1 on port 3260 from a new
        socket. A' will wait for tpg->np_login_sem with np->np_login_timer
        loaded to wait for at most 15 seconds. The lock is held by A so A'
        eventually times out.

     4. Before A' got timeout initiator B gets negotiation failed and calls
        iscsi_target_login_drop()->iscsi_target_login_sess_out().  The
        np->np_login_timer is canceled and initiator A' will hang forever.
        Because A' is now in the login thread, no new login requests can be
        serviced.

     Fix this by moving iscsi_stop_login_thread_timer() out of
     iscsi_target_login_sess_out(). Also remove iscsi_np parameter from
     iscsi_target_login_sess_out().

     Link: https://lore.kernel.org/r/20200729130343.24976-1-houpu@bytedance.com
     Cc: stable@vger.kernel.org
     Reviewed-by: Mike Christie <michael.christie@oracle.com>
     Signed-off-by: Hou Pu <houpu@bytedance.com>
     Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
     Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b03e704b29a1b47f91866b28408ad1241ae8d807
Author: James Smart <james.smart@broadcom.com>
Date:   Fri Aug 28 10:53:29 2020 -0700

     scsi: lpfc: Fix setting IRQ affinity with an empty CPU mask

     commit 7ac836ebcb1509845fe7d66314f469f8e709da93 upstream.

     Some systems are reporting the following log message during driver unload
     or system shutdown:

       ics_rtas_set_affinity: No online cpus in the mask

     A prior commit introduced the writing of an empty affinity mask in calls to
     irq_set_affinity_hint() when disabling interrupts or when there are no
     remaining online CPUs to service an eq interrupt. At least some ppc64
     systems are checking whether affinity masks are empty or not.

     Do not call irq_set_affinity_hint() with an empty CPU mask.

     Fixes: dcaa21367938 ("scsi: lpfc: Change default IRQ model on AMD architectures")
     Link: https://lore.kernel.org/r/20200828175332.130300-2-james.smart@broadcom.com
     Cc: <stable@vger.kernel.org> # v5.5+
     Co-developed-by: Dick Kennedy <dick.kennedy@broadcom.com>
     Signed-off-by: Dick Kennedy <dick.kennedy@broadcom.com>
     Signed-off-by: James Smart <james.smart@broadcom.com>
     Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
     Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 731435aa8105e7691d4532039d30d82379d6e046
Author: Varun Prakash <varun@chelsio.com>
Date:   Tue Aug 25 18:05:10 2020 +0530

     scsi: target: iscsi: Fix data digest calculation

     commit 5528d03183fe5243416c706f64b1faa518b05130 upstream.
```

```
    Current code does not consider 'page_off' in data digest calculation. To
    fix this, add a local variable 'first_sg' and set first_sg.offset to
    sg->offset + page_off.

    Link: https://lore.kernel.org/r/1598358910-3052-1-git-send-email-varun@chelsio.com
    Fixes: e48354ce078c ("iscsi-target: Add iSCSI fabric support for target v4.1")
    Cc: <stable@vger.kernel.org>
    Reviewed-by: Mike Christie <michael.christie@oralce.com>
    Signed-off-by: Varun Prakash <varun@chelsio.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 87f0ffddae01d9587f58cfa70830c6470be8b7b0
Author: Vadym Kochan <vadym.kochan@plvision.eu>
Date:   Mon Aug 31 04:55:39 2020 +0300

    misc: eeprom: at24: register nvmem only after eeprom is ready to use

    commit 45df80d7605c25055a85fbc5a8446c81c6c0ca24 upstream.

    During nvmem_register() the nvmem core sends notifications when:

        - cell added
        - nvmem added

    and during these notifications some callback func may access the nvmem
    device, which will fail in case of at24 eeprom because regulator and pm
    are enabled after nvmem_register().

    Fixes: cd5676db0574 ("misc: eeprom: at24: support pm_runtime control")
    Fixes: b20eb4c1f026 ("eeprom: at24: drop unnecessary label")
    Cc: stable@vger.kernel.org
    Signed-off-by: Vadym Kochan <vadym.kochan@plvision.eu>
    Signed-off-by: Bartosz Golaszewski <bgolaszewski@baylibre.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 94abb06dc753434c0e1d230a269de05f441fc217
Author: Dmitry Osipenko <digetx@gmail.com>
Date:   Mon Aug 31 23:43:35 2020 +0300

    regulator: core: Fix slab-out-of-bounds in regulator_unlock_recursive()

    commit 0a7416f94707c60b9f66b01c0a505b7e41375f3a upstream.

    The recent commit 7d8196641ee1 ("regulator: Remove pointer table
    overallocation") changed the size of coupled_rdevs and now KASAN is able
    to detect slab-out-of-bounds problem in regulator_unlock_recursive(),
    which is a legit problem caused by a typo in the code. The recursive
    unlock function uses n_coupled value of a parent regulator for unlocking
    supply regulator, while supply's n_coupled should be used. In practice
    problem may only affect platforms that use coupled regulators.

    Cc: stable@vger.kernel.org # 5.0+
    Fixes: f8702f9e4aa7 ("regulator: core: Use ww_mutex for regulators locking")
    Signed-off-by: Dmitry Osipenko <digetx@gmail.com>
    Link: https://lore.kernel.org/r/20200831204335.19489-1-digetx@gmail.com
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e6724a64b9425c9dc45a4be993fb61ea4d0f2065
Author: Michał Mirosław <mirq-linux@rere.qmqm.pl>
Date:   Wed Aug 12 03:31:36 2020 +0200

    regulator: plug of_node leak in regulator_register()'s error path
```

```
    commit d3c731564e09b6c2ebefcd1344743a91a237d6dc upstream.

    By calling device_initialize() earlier and noting that kfree(NULL) is
    ok, we can save a bit of code in error handling and plug of_node leak.
    Fixed commit already did part of the work.

    Fixes: 9177514ce349 ("regulator: fix memory leak on error path of regulator_register()")
    Signed-off-by: Michał Mirosław <mirq-linux@rere.qmqm.pl>
    Reviewed-by: Vladimir Zapolskiy <vz@mleia.com>
    Acked-by: Vladimir Zapolskiy <vz@mleia.com>
    Cc: stable@vger.kernel.org
    Link:
https://lore.kernel.org/r/f5035b1b4d40745e66bacd571bbbb5e4644d21a1.1597195321.git.mirq-
linux@rere.qmqm.pl
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 6828c2aae0b725f01bf2a4b1e4a8d4ec71098ce0
Author: Michał Mirosław <mirq-linux@rere.qmqm.pl>
Date:   Wed Aug 12 03:31:36 2020 +0200

    regulator: push allocation in set_consumer_device_supply() out of lock

    commit 5c06540165d443c6455123eb48e7f1a9b618ab34 upstream.

    Pull regulator_list_mutex into set_consumer_device_supply() and keep
    allocations outside of it. Fourth of the fs_reclaim deadlock case.

    Fixes: 45389c47526d ("regulator: core: Add early supply resolution for regulators")
    Signed-off-by: Michał Mirosław <mirq-linux@rere.qmqm.pl>
    Cc: stable@vger.kernel.org
    Link:
https://lore.kernel.org/r/f0380bdb3d60aeefa9693c4e234d2dcda7e56747.1597195321.git.mirq-
linux@rere.qmqm.pl
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 557bc4f1f78ef56703b9edd2f646e663ab7b0e29
Author: Michał Mirosław <mirq-linux@rere.qmqm.pl>
Date:   Wed Aug 12 03:31:35 2020 +0200

    regulator: push allocations in create_regulator() outside of lock

    commit 87fe29b61f9522a3d7b60a4580851f548558186f upstream.

    Move all allocations outside of the regulator_lock()ed section.

    ======================================================
    WARNING: possible circular locking dependency detected
    5.7.13+ #535 Not tainted
    ------------------------------------------------------
    f2fs_discard-179:7/702 is trying to acquire lock:
    c0e5d920 (regulator_list_mutex){+.+.}-{3:3}, at: regulator_lock_dependent+0x54/0x2c0

    but task is already holding lock:
    cb95b080 (&dcc->cmd_lock){+.+.}-{3:3}, at: __issue_discard_cmd+0xec/0x5f8

    which lock already depends on the new lock.

    the existing dependency chain (in reverse order) is:

    [...]

    -> #3 (fs_reclaim){+.+.}-{0:0}:
```

```
             fs_reclaim_acquire.part.11+0x40/0x50
             fs_reclaim_acquire+0x24/0x28
             __kmalloc_track_caller+0x54/0x218
             kstrdup+0x40/0x5c
             create_regulator+0xf4/0x368
             regulator_resolve_supply+0x1a0/0x200
             regulator_register+0x9c8/0x163c

     [...]

     other info that might help us debug this:

     Chain exists of:
       regulator_list_mutex --> &sit_i->sentry_lock --> &dcc->cmd_lock

     [...]

     Fixes: f8702f9e4aa7 ("regulator: core: Use ww_mutex for regulators locking")
     Signed-off-by: Michał Mirosław <mirq-linux@rere.qmqm.pl>
     Cc: stable@vger.kernel.org
     Link:
  https://lore.kernel.org/r/6eebc99b2474f4ffaa0405b15178ece0e7e4f608.1597195321.git.mirq-
  linux@rere.qmqm.pl
     Signed-off-by: Mark Brown <broonie@kernel.org>
     Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

  commit 96ccd5a5bc0668730c34803145442c9daf45d014
  Author: Michał Mirosław <mirq-linux@rere.qmqm.pl>
  Date:   Wed Aug 12 03:31:34 2020 +0200

     regulator: push allocation in regulator_init_coupling() outside of lock

     commit 73a32129f8ccb556704a26b422f54e048bf14bd0 upstream.

     Allocating memory with regulator_list_mutex held makes lockdep unhappy
     when memory pressure makes the system do fs_reclaim on eg. eMMC using
     a regulator. Push the lock inside regulator_init_coupling() after the
     allocation.

     ======================================================
     WARNING: possible circular locking dependency detected
     5.7.13+ #533 Not tainted
     ------------------------------------------------------
     kswapd0/383 is trying to acquire lock:
     cca78ca4 (&sbi->write_io[i][j].io_rwsem){++++}-{3:3}, at:
  __submit_merged_write_cond+0x104/0x154
     but task is already holding lock:
     c0e38518 (fs_reclaim){+.+.}-{0:0}, at: __fs_reclaim_acquire+0x0/0x50
     which lock already depends on the new lock.
     the existing dependency chain (in reverse order) is:
     -> #2 (fs_reclaim){+.+.}-{0:0}:
            fs_reclaim_acquire.part.11+0x40/0x50
            fs_reclaim_acquire+0x24/0x28
            __kmalloc+0x54/0x218
            regulator_register+0x860/0x1584
            dummy_regulator_probe+0x60/0xa8
     [...]
     other info that might help us debug this:

     Chain exists of:
       &sbi->write_io[i][j].io_rwsem --> regulator_list_mutex --> fs_reclaim

     Possible unsafe locking scenario:

            CPU0                    CPU1
```

```
           ----                    ----
      lock(fs_reclaim);
                                   lock(regulator_list_mutex);
                                   lock(fs_reclaim);
      lock(&sbi->write_io[i][j].io_rwsem);
     *** DEADLOCK ***

    1 lock held by kswapd0/383:
     #0: c0e38518 (fs_reclaim){+.+.}-{0:0}, at: __fs_reclaim_acquire+0x0/0x50
    [...]

    Fixes: d8ca7d184b33 ("regulator: core: Introduce API for regulators coupling
customization")
    Signed-off-by: Michał Mirosław <mirq-linux@rere.qmqm.pl>
    Reviewed-by: Dmitry Osipenko <digetx@gmail.com>
    Cc: stable@vger.kernel.org
    Link:
https://lore.kernel.org/r/1a889cf7f61c6429c9e6b34ddcdde99be77a26b6.1597195321.git.mirq-
linux@rere.qmqm.pl
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 47052502f04084426b3465ea1f874c8a9cc9861e
Author: Andy Shevchenko <andriy.shevchenko@linux.intel.com>
Date:   Mon Aug 3 11:27:06 2020 +0300

    kobject: Restore old behaviour of kobject_del(NULL)

    commit 40b8b826a6998639dd1c26f0e127f18371e1058d upstream.

    The commit 079ad2fb4bf9 ("kobject: Avoid premature parent object freeing in
    kobject_cleanup()") inadvertently dropped a possibility to call kobject_del()
    with NULL pointer. Restore the old behaviour.

    Fixes: 079ad2fb4bf9 ("kobject: Avoid premature parent object freeing in
kobject_cleanup()")
    Cc: stable <stable@vger.kernel.org>
    Reported-by: Qu Wenruo <quwenruo.btrfs@gmx.com>
    Cc: Heikki Krogerus <heikki.krogerus@linux.intel.com>
    Signed-off-by: Andy Shevchenko <andriy.shevchenko@linux.intel.com>
    Reviewed-by: Qu Wenruo <wqu@suse.com>
    Link: https://lore.kernel.org/r/20200803082706.65347-1-andriy.shevchenko@linux.intel.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a36133cc9cf243e5b89201ca3a36b9d755f60a25
Author: Nikunj A. Dadhania <nikunj.dadhania@linux.intel.com>
Date:   Tue Jul 21 17:05:23 2020 +0530

    thunderbolt: Disable ports that are not implemented

    commit 8824d19b45867be75d375385414c4f06719a11a4 upstream.

    Commit 4caf2511ec49 ("thunderbolt: Add trivial .shutdown") exposes a bug
    in the Thunderbolt driver, that frees an unallocated id, resulting in the
    following spinlock bad magic bug.

    [ 20.633803] BUG: spinlock bad magic on CPU#4, halt/3313
    [ 20.640030] lock: 0xffff92e6ad5c97e0, .magic: 00000000, .owner: <none>/-1, .owner_cpu:
0
    [ 20.672139] Call Trace:
    [ 20.675032] dump_stack+0x97/0xdb
    [ 20.678950] ? spin_bug+0xa5/0xb0
    [ 20.682865] do_raw_spin_lock+0x68/0x98
    [ 20.687397] _raw_spin_lock_irqsave+0x3f/0x5d
    [ 20.692535] ida_destroy+0x4f/0x124
```

```
     [ 20.696657] tb_switch_release+0x6d/0xfd
     [ 20.701295] device_release+0x2c/0x7d
     [ 20.705622] kobject_put+0x8e/0xac
     [ 20.709637] tb_stop+0x55/0x66
     [ 20.713243] tb_domain_remove+0x36/0x62
     [ 20.717774] nhi_remove+0x4d/0x58

     Fix the issue by disabling ports that are enabled as per the EEPROM, but
     not implemented. While at it, update the kernel doc for the disabled
     field, to reflect this.

     Cc: stable@vger.kernel.org
     Fixes: 4caf2511ec49 ("thunderbolt: Add trivial .shutdown")
     Reported-by: Srikanth Nandamuri <srikanth.nandamuri@intel.com>
     Signed-off-by: Nikunj A. Dadhania <nikunj.dadhania@linux.intel.com>
     Signed-off-by: Mika Westerberg <mika.westerberg@linux.intel.com>
     Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 171169d8254b7e6b0e299973460fa49744685900
Author: Filipe Manana <fdmanana@suse.com>
Date:   Mon Sep 14 09:01:04 2020 +0100

     btrfs: fix wrong address when faulting in pages in the search ioctl

     commit 1c78544eaa4660096aeb6a57ec82b42cdb3bfe5a upstream.

     When faulting in the pages for the user supplied buffer for the search
     ioctl, we are passing only the base address of the buffer to the function
     fault_in_pages_writeable(). This means that after the first iteration of
     the while loop that searches for leaves, when we have a non-zero offset,
     stored in 'sk_offset', we try to fault in a wrong page range.

     So fix this by adding the offset in 'sk_offset' to the base address of the
     user supplied buffer when calling fault_in_pages_writeable().

     Several users have reported that the applications compsize and bees have
     started to operate incorrectly since commit a48b73eca4ceb9 ("btrfs: fix
     potential deadlock in the search ioctl") was added to stable trees, and
     these applications make heavy use of the search ioctls. This fixes their
     issues.

     Link: https://lore.kernel.org/linux-btrfs/632b888d-a3c3-b085-cdf5-
f9bb61017d92@lechevalier.se/
     Link: https://github.com/kilobyte/compsize/issues/34
     Fixes: a48b73eca4ceb9 ("btrfs: fix potential deadlock in the search ioctl")
     CC: stable@vger.kernel.org # 4.4+
     Tested-by: A L <mail@lechevalier.se>
     Reviewed-by: Josef Bacik <josef@toxicpanda.com>
     Signed-off-by: Filipe Manana <fdmanana@suse.com>
     Reviewed-by: David Sterba <dsterba@suse.com>
     Signed-off-by: David Sterba <dsterba@suse.com>
     Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ac049e1bd9eb26319e76e62948b72c9252d14814
Author: Josef Bacik <josef@toxicpanda.com>
Date:   Thu Sep 3 14:29:50 2020 -0400

     btrfs: free data reloc tree on failed mount

     commit 9e3aa8054453d23d9f477f0cdae70a6a1ea6ec8a upstream.

     While testing a weird problem with -o degraded, I noticed I was getting
     leaked root errors

       BTRFS warning (device loop0): writable mount is not allowed due to too many missing
```

```
  devices
        BTRFS error (device loop0): open_ctree failed
        BTRFS error (device loop0): leaked root -9-0 refcount 1

      This is the DATA_RELOC root, which gets read before the other fs roots,
      but is included in the fs roots radix tree.  Handle this by adding a
      btrfs_drop_and_free_fs_root() on the data reloc root if it exists.  This
      is ok to do here if we fail further up because we will only drop the ref
      if we delete the root from the radix tree, and all other cleanup won't
      be duplicated.

      CC: stable@vger.kernel.org # 5.8+
      Reviewed-by: Nikolay Borisov <nborisov@suse.com>
      Signed-off-by: Josef Bacik <josef@toxicpanda.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 6368e27c5edba7e13e0f15d825d0b6308971648b
  Author: Josef Bacik <josef@toxicpanda.com>
  Date:   Mon Aug 31 10:52:42 2020 -0400

      btrfs: fix lockdep splat in add_missing_dev

      commit fccc0007b8dc952c6bc0805cdf842eb8ea06a639 upstream.

      Nikolay reported a lockdep splat in generic/476 that I could reproduce
      with btrfs/187.

        ======================================================
        WARNING: possible circular locking dependency detected
        5.9.0-rc2+ #1 Tainted: G        W
        ------------------------------------------------------
        kswapd0/100 is trying to acquire lock:
        ffff9e8ef38b6268 (&delayed_node->mutex){+.+.}-{3:3}, at:
  __btrfs_release_delayed_node.part.0+0x3f/0x330

        but task is already holding lock:
        ffffffffa9d74700 (fs_reclaim){+.+.}-{0:0}, at: __fs_reclaim_acquire+0x5/0x30

        which lock already depends on the new lock.

        the existing dependency chain (in reverse order) is:

        -> #2 (fs_reclaim){+.+.}-{0:0}:
               fs_reclaim_acquire+0x65/0x80
               slab_pre_alloc_hook.constprop.0+0x20/0x200
               kmem_cache_alloc_trace+0x3a/0x1a0
               btrfs_alloc_device+0x43/0x210
               add_missing_dev+0x20/0x90
               read_one_chunk+0x301/0x430
               btrfs_read_sys_array+0x17b/0x1b0
               open_ctree+0xa62/0x1896
               btrfs_mount_root.cold+0x12/0xea
               legacy_get_tree+0x30/0x50
               vfs_get_tree+0x28/0xc0
               vfs_kern_mount.part.0+0x71/0xb0
               btrfs_mount+0x10d/0x379
               legacy_get_tree+0x30/0x50
               vfs_get_tree+0x28/0xc0
               path_mount+0x434/0xc00
               __x64_sys_mount+0xe3/0x120
               do_syscall_64+0x33/0x40
               entry_SYSCALL_64_after_hwframe+0x44/0xa9
```

```
       -> #1 (&fs_info->chunk_mutex){+.+.}-{3:3}:
              __mutex_lock+0x7e/0x7e0
              btrfs_chunk_alloc+0x125/0x3a0
              find_free_extent+0xdf6/0x1210
              btrfs_reserve_extent+0xb3/0x1b0
              btrfs_alloc_tree_block+0xb0/0x310
              alloc_tree_block_no_bg_flush+0x4a/0x60
              __btrfs_cow_block+0x11a/0x530
              btrfs_cow_block+0x104/0x220
              btrfs_search_slot+0x52e/0x9d0
              btrfs_lookup_inode+0x2a/0x8f
              __btrfs_update_delayed_inode+0x80/0x240
              btrfs_commit_inode_delayed_inode+0x119/0x120
              btrfs_evict_inode+0x357/0x500
              evict+0xcf/0x1f0
              vfs_rmdir.part.0+0x149/0x160
              do_rmdir+0x136/0x1a0
              do_syscall_64+0x33/0x40
              entry_SYSCALL_64_after_hwframe+0x44/0xa9

       -> #0 (&delayed_node->mutex){+.+.}-{3:3}:
              __lock_acquire+0x1184/0x1fa0
              lock_acquire+0xa4/0x3d0
              __mutex_lock+0x7e/0x7e0
              __btrfs_release_delayed_node.part.0+0x3f/0x330
              btrfs_evict_inode+0x24c/0x500
              evict+0xcf/0x1f0
              dispose_list+0x48/0x70
              prune_icache_sb+0x44/0x50
              super_cache_scan+0x161/0x1e0
              do_shrink_slab+0x178/0x3c0
              shrink_slab+0x17c/0x290
              shrink_node+0x2b2/0x6d0
              balance_pgdat+0x30a/0x670
              kswapd+0x213/0x4c0
              kthread+0x138/0x160
              ret_from_fork+0x1f/0x30

       other info that might help us debug this:

       Chain exists of:
         &delayed_node->mutex --> &fs_info->chunk_mutex --> fs_reclaim

        Possible unsafe locking scenario:

              CPU0                    CPU1
              ----                    ----
         lock(fs_reclaim);
                                      lock(&fs_info->chunk_mutex);
                                      lock(fs_reclaim);
         lock(&delayed_node->mutex);

        *** DEADLOCK ***

       3 locks held by kswapd0/100:
        #0: ffffffffa9d74700 (fs_reclaim){+.+.}-{0:0}, at: __fs_reclaim_acquire+0x5/0x30
        #1: ffffffffa9d65c50 (shrinker_rwsem){++++}-{3:3}, at: shrink_slab+0x115/0x290
        #2: ffff9e8e9da260e0 (&type->s_umount_key#48){++++}-{3:3}, at:
 super_cache_scan+0x38/0x1e0

       stack backtrace:
       CPU: 1 PID: 100 Comm: kswapd0 Tainted: G        W         5.9.0-rc2+ #1
       Hardware name: QEMU Standard PC (Q35 + ICH9, 2009), BIOS 1.13.0-2.fc32 04/01/2014
       Call Trace:
        dump_stack+0x92/0xc8
```

```
        check_noncircular+0x12d/0x150
        __lock_acquire+0x1184/0x1fa0
        lock_acquire+0xa4/0x3d0
        ? __btrfs_release_delayed_node.part.0+0x3f/0x330
        __mutex_lock+0x7e/0x7e0
        ? __btrfs_release_delayed_node.part.0+0x3f/0x330
        ? __btrfs_release_delayed_node.part.0+0x3f/0x330
        ? lock_acquire+0xa4/0x3d0
        ? btrfs_evict_inode+0x11e/0x500
        ? find_held_lock+0x2b/0x80
        __btrfs_release_delayed_node.part.0+0x3f/0x330
        btrfs_evict_inode+0x24c/0x500
        evict+0xcf/0x1f0
        dispose_list+0x48/0x70
        prune_icache_sb+0x44/0x50
        super_cache_scan+0x161/0x1e0
        do_shrink_slab+0x178/0x3c0
        shrink_slab+0x17c/0x290
        shrink_node+0x2b2/0x6d0
        balance_pgdat+0x30a/0x670
        kswapd+0x213/0x4c0
        ? _raw_spin_unlock_irqrestore+0x46/0x60
        ? add_wait_queue_exclusive+0x70/0x70
        ? balance_pgdat+0x670/0x670
        kthread+0x138/0x160
        ? kthread_create_worker_on_cpu+0x40/0x40
        ret_from_fork+0x1f/0x30

    This is because we are holding the chunk_mutex when we call
    btrfs_alloc_device, which does a GFP_KERNEL allocation.  We don't want
    to switch that to a GFP_NOFS lock because this is the only place where
    it matters.  So instead use memalloc_nofs_save() around the allocation
    in order to avoid the lockdep splat.

    Reported-by: Nikolay Borisov <nborisov@suse.com>
    CC: stable@vger.kernel.org # 4.4+
    Reviewed-by: Anand Jain <anand.jain@oracle.com>
    Signed-off-by: Josef Bacik <josef@toxicpanda.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 731cef38b55e788657132d921509493ce1e309fb
Author: Qu Wenruo <wqu@suse.com>
Date:   Wed Aug 26 17:26:43 2020 +0800

    btrfs: require only sector size alignment for parent eb bytenr

    commit ea57788eb76dc81f6003245427356a1dcd0ac524 upstream.

    [BUG]
    A completely sane converted fs will cause kernel warning at balance
    time:

      [ 1557.188633] BTRFS info (device sda7): relocating block group 8162107392 flags data
      [ 1563.358078] BTRFS info (device sda7): found 11722 extents
      [ 1563.358277] BTRFS info (device sda7): leaf 7989321728 gen 95 total ptrs 213 free
space 3458 owner 2
      [ 1563.358280]        item 0 key (7984947200 169 0) itemoff 16250 itemsize 33
      [ 1563.358281]                extent refs 1 gen 90 flags 2
      [ 1563.358282]                ref#0: tree block backref root 4
      [ 1563.358285]        item 1 key (7985602560 169 0) itemoff 16217 itemsize 33
      [ 1563.358286]                extent refs 1 gen 93 flags 258
      [ 1563.358287]                ref#0: shared block backref parent 7985602560
      [ 1563.358288]                        (parent 7985602560 is NOT ALIGNED to nodesize
```

```
 16384)
      [ 1563.358290]        item 2 key (7985635328 169 0) itemoff 16184 itemsize 33
      ...
      [ 1563.358995] BTRFS error (device sda7): eb 7989321728 invalid extent inline ref type
 182
      [ 1563.358996] ------------[ cut here ]------------
      [ 1563.359005] WARNING: CPU: 14 PID: 2930 at 0xffffffff9f231766

    Then with transaction abort, and obviously failed to balance the fs.

    [CAUSE]
    That mentioned inline ref type 182 is completely sane, it's
    BTRFS_SHARED_BLOCK_REF_KEY, it's some extra check making kernel to
    believe it's invalid.

    Commit 64ecdb647ddb ("Btrfs: add one more sanity check for shared ref
    type") introduced extra checks for backref type.

    One of the requirement is, parent bytenr must be aligned to node size,
    which is not correct.

    One example is like this:

    0       1G  1G+4K               2G 2G+4K
            |   |////////////////////|//|  <- A chunk starts at 1G+4K
                |   |       <- A tree block get reserved at bytenr 1G+4K

    Then we have a valid tree block at bytenr 1G+4K, but not aligned to
    nodesize (16K).

    Such chunk is not ideal, but current kernel can handle it pretty well.
    We may warn about such tree block in the future, but should not reject
    them.

    [FIX]
    Change the alignment requirement from node size alignment to sector size
    alignment.

    Also, to make our lives a little easier, also output @iref when
    btrfs_get_extent_inline_ref_type() failed, so we can locate the item
    easier.

    Bugzilla: https://bugzilla.kernel.org/show_bug.cgi?id=205475
    Fixes: 64ecdb647ddb ("Btrfs: add one more sanity check for shared ref type")
    CC: stable@vger.kernel.org # 4.14+
    Reviewed-by: Josef Bacik <josef@toxicpanda.com>
    Signed-off-by: Qu Wenruo <wqu@suse.com>
    [ update comments and messages ]
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

 commit 75ee09f3b745726be256394a08a301c00d8d0320
 Author: Rustam Kovhaev <rkovhaev@gmail.com>
 Date:   Tue Aug 4 07:56:14 2020 -0700

    staging: wlan-ng: fix out of bounds read in prism2sta_probe_usb()

    commit fea22e159d51c766ba70473f473a0ec914cc7e92 upstream.

    let's use usb_find_common_endpoints() to discover endpoints, it does all
    necessary checks for type and xfer direction

    remove memset() in hfa384x_create(), because we now assign endpoints in
    prism2sta_probe_usb() and because create_wlan() uses kzalloc() to
    allocate hfa384x struct before calling hfa384x_create()
```

```
    Fixes: faaff9765664 ("staging: wlan-ng: properly check endpoint types")
    Reported-and-tested-by: syzbot+22794221ab96b0bab53a@syzkaller.appspotmail.com
    Link: https://syzkaller.appspot.com/bug?extid=22794221ab96b0bab53a
    Signed-off-by: Rustam Kovhaev <rkovhaev@gmail.com>
    Cc: stable <stable@vger.kernel.org>
    Link: https://lore.kernel.org/r/20200804145614.104320-1-rkovhaev@gmail.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ac0698332870b48f76f1802a8994e705b17a1c74
Author: Jonathan Cameron <Jonathan.Cameron@huawei.com>
Date:   Wed Jul 22 16:50:38 2020 +0100

    iio:accel:mma8452: Fix timestamp alignment and prevent data leak.

    commit 89226a296d816727405d3fea684ef69e7d388bd8 upstream.

    One of a class of bugs pointed out by Lars in a recent review.
    iio_push_to_buffers_with_timestamp assumes the buffer used is aligned
    to the size of the timestamp (8 bytes).  This is not guaranteed in
    this driver which uses a 16 byte u8 array on the stack.  As Lars also noted
    this anti pattern can involve a leak of data to userspace and that
    indeed can happen here.  We close both issues by moving to
    a suitable structure in the iio_priv() data with alignment
    ensured by use of an explicit c structure.  This data is allocated
    with kzalloc so no data can leak appart from previous readings.

    The additional forcing of the 8 byte alignment of the timestamp
    is not strictly necessary but makes the code less fragile by
    making this explicit.

    Fixes: c7eeea93ac60 ("iio: Add Freescale MMA8452Q 3-axis accelerometer driver")
    Reported-by: Lars-Peter Clausen <lars@metafoo.de>
    Cc: Peter Meerwald <pmeerw@pmeerw.net>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Reviewed-by: Andy Shevchenko <andy.shevchenko@gmail.com>
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e8e7cba15d6118002c418090d3d5a44bdb21add3
Author: Jonathan Cameron <Jonathan.Cameron@huawei.com>
Date:   Wed Jul 22 16:50:40 2020 +0100

    iio:accel:mma7455: Fix timestamp alignment and prevent data leak.

    commit 7e5ac1f2206eda414f90c698fe1820dee873394d upstream.

    One of a class of bugs pointed out by Lars in a recent review.
    iio_push_to_buffers_with_timestamp assumes the buffer used is aligned
    to the size of the timestamp (8 bytes).  This is not guaranteed in
    this driver which uses a 16 byte u8 array on the stack   As Lars also noted
    this anti pattern can involve a leak of data to userspace and that
    indeed can happen here.  We close both issues by moving to
    a suitable structure in the iio_priv() data with alignment
    ensured by use of an explicit c structure.  This data is allocated
    with kzalloc so no data can leak appart from previous readings.

    The force alignment of ts is not strictly necessary in this particularly
    case but does make the code less fragile.

    Fixes: a84ef0d181d9 ("iio: accel: add Freescale MMA7455L/MMA7456L 3-axis accelerometer
driver")
    Reported-by: Lars-Peter Clausen <lars@metafoo.de>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Cc: <Stable@vger.kernel.org>
```

```
    Reviewed-by: Andy Shevchenko <andy.shevchenko@gmail.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 87efc85d53d7136504270320151953795cd847c1
Author: Jonathan Cameron <Jonathan.Cameron@huawei.com>
Date:   Wed Jul 22 16:50:37 2020 +0100

    iio: accel: kxsd9: Fix alignment of local buffer.

    commit 95ad67577de4ea08eb8e441394e698aa4addcc0b upstream.

    iio_push_to_buffers_with_timestamp assumes 8 byte alignment which
    is not guaranteed by an array of smaller elements.

    Note that whilst in this particular case the alignment forcing
    of the ts element is not strictly necessary it acts as good
    documentation.  Doing this where not necessary should cut
    down on the number of cut and paste introduced errors elsewhere.

    Fixes: 0427a106a98a ("iio: accel: kxsd9: Add triggered buffer handling")
    Reported-by: Lars-Peter Clausen <lars@metafoo.de>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Reviewed-by: Andy Shevchenko <andy.shevchenko@gmail.com>
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 58f091e131d1c01d7edeff24aa06e15fa8acd9b9
Author: Jonathan Cameron <Jonathan.Cameron@huawei.com>
Date:   Wed Jul 22 16:50:43 2020 +0100

    iio:chemical:ccs811: Fix timestamp alignment and prevent data leak.

    commit eb1a148ef41d8ae8d9201efc3f1b145976290331 upstream.

    One of a class of bugs pointed out by Lars in a recent review.
    iio_push_to_buffers_with_timestamp assumes the buffer used is aligned
    to the size of the timestamp (8 bytes).  This is not guaranteed in
    this driver which uses an array of smaller elements on the stack.
    As Lars also noted this anti pattern can involve a leak of data to
    userspace and that indeed can happen here.  We close both issues by
    moving to a suitable structure in the iio_priv() data with alignment
    explicitly requested.  This data is allocated with kzalloc so no
    data can leak appart from previous readings.

    The explicit alignment of ts is necessary to ensure consistent
    padding for x86_32 in which the ts would otherwise be 4 byte aligned.

    Fixes: 283d26917ad6 ("iio: chemical: ccs811: Add triggered buffer support")
    Reported-by: Lars-Peter Clausen <lars@metafoo.de>
    Cc: Narcisa Ana Maria Vasile <narcisaanamaria12@gmail.com>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Reviewed-by: Andy Shevchenko <andy.shevchenko@gmail.com>
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit be0bc3cf9e607961a11f4ee3a733388a5e4dd148
Author: Jonathan Cameron <Jonathan.Cameron@huawei.com>
Date:   Wed Jul 22 16:50:45 2020 +0100

    iio:light:max44000 Fix timestamp alignment and prevent data leak.

    commit 523628852a5f5f34a15252b2634d0498d3cfb347 upstream.

    One of a class of bugs pointed out by Lars in a recent review.
    iio_push_to_buffers_with_timestamp assumes the buffer used is aligned
```

```
      to the size of the timestamp (8 bytes).  This is not guaranteed in
      this driver which uses a 16 byte array of smaller elements on the stack.
      As Lars also noted this anti pattern can involve a leak of data to
      userspace and that indeed can happen here.  We close both issues by
      moving to a suitable structure in the iio_priv().
      This data is allocated with kzalloc so no data can leak appart
      from previous readings.

      It is necessary to force the alignment of ts to avoid the padding
      on x86_32 being different from 64 bit platorms (it alows for
      4 bytes aligned 8 byte types.

      Fixes: 06ad7ea10e2b ("max44000: Initial triggered buffer support")
      Reported-by: Lars-Peter Clausen <lars@metafoo.de>
      Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
      Reviewed-by: Andy Shevchenko <andy.shevchenko@gmail.com>
      Cc: <Stable@vger.kernel.org>
      Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

  commit 6ae35c1825e85cea3deb4e04d4e690aeda0cd0d7
  Author: Jonathan Cameron <Jonathan.Cameron@huawei.com>
  Date:   Wed Jul 22 16:50:49 2020 +0100

      iio:magnetometer:ak8975 Fix alignment and data leak issues.

      commit 02ad21cefbac4d89ac443866f25b90449527737b upstream.

      One of a class of bugs pointed out by Lars in a recent review.
      iio_push_to_buffers_with_timestamp assumes the buffer used is aligned
      to the size of the timestamp (8 bytes).  This is not guaranteed in
      this driver which uses an array of smaller elements on the stack.
      As Lars also noted this anti pattern can involve a leak of data to
      userspace and that indeed can happen here.  We close both issues by
      moving to a suitable structure in the iio_priv() data.

      This data is allocated with kzalloc so no data can leak apart from
      previous readings.

      The explicit alignment of ts is not necessary in this case as by
      coincidence the padding will end up the same, however I consider
      it to make the code less fragile and have included it.

      Fixes: bc11ca4a0b84 ("iio:magnetometer:ak8975: triggered buffer support")
      Reported-by: Lars-Peter Clausen <lars@metafoo.de>
      Cc: Gregor Boirie <gregor.boirie@parrot.com>
      Cc: Linus Walleij <linus.walleij@linaro.org>
      Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
      Reviewed-by: Andy Shevchenko <andy.shevchenko@gmail.com>
      Cc: <Stable@vger.kernel.org>
      Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

  commit 1f2b6037d4ffc699b7d55211acbe7f4cab5060e5
  Author: Jonathan Cameron <Jonathan.Cameron@huawei.com>
  Date:   Wed Jul 22 16:50:56 2020 +0100

      iio:adc:ti-adc081c Fix alignment and data leak issues

      commit 54f82df2ba86e2a8e9cbf4036d192366e3905c89 upstream.

      One of a class of bugs pointed out by Lars in a recent review.
      iio_push_to_buffers_with_timestamp assumes the buffer used is aligned
      to the size of the timestamp (8 bytes).  This is not guaranteed in
      this driver which uses an array of smaller elements on the stack.
      As Lars also noted this anti pattern can involve a leak of data to
      userspace and that indeed can happen here.  We close both issues by
```

```
    moving to a suitable structure in the iio_priv().

    This data is allocated with kzalloc so no data can leak apart
    from previous readings.

    The eplicit alignment of ts is necessary to ensure correct padding
    on x86_32 where s64 is only aligned to 4 bytes.

    Fixes: 08e05d1fce5c ("ti-adc081c: Initial triggered buffer support")
    Reported-by: Lars-Peter Clausen <lars@metafoo.de>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Reviewed-by: Andy Shevchenko <andy.shevchenko@gmail.com>
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 456fd60c5803d224a24fde05952715c23a1c004b
Author: Jonathan Cameron <Jonathan.Cameron@huawei.com>
Date:   Wed Jul 22 16:51:03 2020 +0100

    iio:adc:max1118 Fix alignment of timestamp and data leak issues

    commit db8f06d97ec284dc018e2e4890d2e5035fde8630 upstream.

    One of a class of bugs pointed out by Lars in a recent review.
    iio_push_to_buffers_with_timestamp assumes the buffer used is aligned
    to the size of the timestamp (8 bytes).  This is not guaranteed in
    this driver which uses an array of smaller elements on the stack.
    As Lars also noted this anti pattern can involve a leak of data to
    userspace and that indeed can happen here.  We close both issues by
    moving to a suitable structure in the iio_priv() data.

    This data is allocated with kzalloc so no data can leak apart
    from previous readings.

    The explicit alignment of ts is necessary to ensure correct padding
    on architectures where s64 is only 4 bytes aligned such as x86_32.

    Fixes: a9e9c7153e96 ("iio: adc: add max1117/max1118/max1119 ADC driver")
    Reported-by: Lars-Peter Clausen <lars@metafoo.de>
    Cc: Akinobu Mita <akinobu.mita@gmail.com>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Reviewed-by: Andy Shevchenko <andy.shevchenko@gmail.com>
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 743c6bc208d9db2112fed57249d4d8ed69b05468
Author: Jonathan Cameron <Jonathan.Cameron@huawei.com>
Date:   Wed Jul 22 16:51:02 2020 +0100

    iio:adc:ina2xx Fix timestamp alignment issue.

    commit f8cd222feb82ecd82dcf610fcc15186f55f9c2b5 upstream.

    One of a class of bugs pointed out by Lars in a recent review.
    iio_push_to_buffers_with_timestamp assumes the buffer used is aligned
    to the size of the timestamp (8 bytes).  This is not guaranteed in
    this driver which uses a 32 byte array of smaller elements on the stack.
    As Lars also noted this anti pattern can involve a leak of data to
    userspace and that indeed can happen here.  We close both issues by
    moving to a suitable structure in the iio_priv() data with alignment
    explicitly requested.  This data is allocated with kzalloc so no
    data can leak apart from previous readings. The explicit alignment
    isn't technically needed here, but it reduced fragility and avoids
    cut and paste into drivers where it will be needed.
```

```
    If we want this in older stables will need manual backport due to
    driver reworks.

    Fixes: c43a102e67db ("iio: ina2xx: add support for TI INA2xx Power Monitors")
    Reported-by: Lars-Peter Clausen <lars@metafoo.de>
    Cc: Stefan Brüns <stefan.bruens@rwth-aachen.de>
    Cc: Marc Titinger <mtitinger@baylibre.com>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Reviewed-by: Andy Shevchenko <andy.shevchenko@gmail.com>
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 98bd6a0d4237c994806afd171adf9a82431a3580
Author: Jonathan Cameron <Jonathan.Cameron@huawei.com>
Date:   Wed Jul 22 16:50:57 2020 +0100

    iio:adc:ti-adc084s021 Fix alignment and data leak issues.

    commit a661b571e3682705cb402a5cd1e970586a3ec00f upstream.

    One of a class of bugs pointed out by Lars in a recent review.
    iio_push_to_buffers_with_timestamp assumes the buffer used is aligned
    to the size of the timestamp (8 bytes).  This is not guaranteed in
    this driver which uses an array of smaller elements on the stack.
    As Lars also noted this anti pattern can involve a leak of data to
    userspace and that indeed can happen here.  We close both issues by
    moving to a suitable structure in the iio_priv().

    This data is allocated with kzalloc so no data can leak apart from
    previous readings.

    The force alignment of ts is not strictly necessary in this case
    but reduces the fragility of the code.

    Fixes: 3691e5a69449 ("iio: adc: add driver for the ti-adc084s021 chip")
    Reported-by: Lars-Peter Clausen <lars@metafoo.de>
    Cc: Mårten Lindahl <martenli@axis.com>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Reviewed-by: Andy Shevchenko <andy.shevchenko@gmail.com>
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3df5a17f9c362c4aa754b70daf9e84ab9918115d
Author: Jonathan Cameron <Jonathan.Cameron@huawei.com>
Date:   Wed Jul 22 16:50:39 2020 +0100

    iio:accel:bmc150-accel: Fix timestamp alignment and prevent data leak.

    commit a6f86f724394de3629da63fe5e1b7a4ab3396efe upstream.

    One of a class of bugs pointed out by Lars in a recent review.
    iio_push_to_buffers_with_timestamp assumes the buffer used is aligned
    to the size of the timestamp (8 bytes).  This is not guaranteed in
    this driver which uses a 16 byte array of smaller elements on the stack.
    As Lars also noted this anti pattern can involve a leak of data to
    userspace and that indeed can happen here.  We close both issues by moving
    to a suitable structure in the iio_priv() data with alignment
    ensured by use of an explicit c structure.  This data is allocated
    with kzalloc so no data can leak appart from previous readings.

    Fixes tag is beyond some major refactoring so likely manual backporting
    would be needed to get that far back.

    Whilst the force alignment of the ts is not strictly necessary, it
    does make the code less fragile.
```

```
    Fixes: 3bbec9773389 ("iio: bmc150_accel: add support for hardware fifo")
    Reported-by: Lars-Peter Clausen <lars@metafoo.de>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Acked-by: Srinivas Pandruvada <srinivas.pandruvada@linux.intel.com>
    Reviewed-by: Andy Shevchenko <andy.shevchenko@gmail.com>
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 35304bc22976fabfc1dd8d28552cd10ba2a11032
Author: Jonathan Cameron <Jonathan.Cameron@huawei.com>
Date:   Wed Jul 22 16:50:42 2020 +0100

    iio:proximity:mb1232: Fix timestamp alignment and prevent data leak.

    commit f60e8bb84282b8e633956cfe74b4f0d64ca73cec upstream.

    One of a class of bugs pointed out by Lars in a recent review.
    iio_push_to_buffers_with_timestamp assumes the buffer used is aligned
    to the size of the timestamp (8 bytes).  This is not guaranteed in
    this driver which uses a 16 byte s16 array on the stack   As Lars also noted
    this anti pattern can involve a leak of data to userspace and that
    indeed can happen here.  We close both issues by moving to
    a suitable structure in the iio_priv() data with alignment
    ensured by use of an explicit c structure.  This data is allocated
    with kzalloc so no data can leak appart from previous readings.

    In this case the forced alignment of the ts is necessary to ensure
    correct padding on x86_32 where the s64 would only be 4 byte aligned.

    Fixes: 16b05261537e ("mb1232.c: add distance iio sensor with i2c")
    Reported-by: Lars-Peter Clausen <lars@metafoo.de>
    Cc: Andreas Klinger <ak@it-klinger.de>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Cc: <Stable@vger.kernel.org>
    Reviewed-by: Andy Shevchenko <andy.shevchenko@gmail.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c56598930b1ce5afc404455f367af58c260a64b7
Author: Jonathan Cameron <Jonathan.Cameron@huawei.com>
Date:   Wed Jul 22 16:50:48 2020 +0100

    iio:light:ltr501 Fix timestamp alignment issue.

    commit 2684d5003490df5398aeafe2592ba9d4a4653998 upstream.

    One of a class of bugs pointed out by Lars in a recent review.
    iio_push_to_buffers_with_timestamp assumes the buffer used is aligned
    to the size of the timestamp (8 bytes).  This is not guaranteed in
    this driver which uses an array of smaller elements on the stack.
    Here we use a structure on the stack.  The driver already did an
    explicit memset so no data leak was possible.

    Forced alignment of ts is not strictly necessary but probably makes
    the code slightly less fragile.

    Note there has been some rework in this driver of the years, so no
    way this will apply cleanly all the way back.

    Fixes: 2690be905123 ("iio: Add Lite-On ltr501 ambient light / proximity sensor driver")
    Reported-by: Lars-Peter Clausen <lars@metafoo.de>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Reviewed-by: Andy Shevchenko <andy.shevchenko@gmail.com>
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit 678f5426826944e94b5500f874e4ca200fe77052
Author: Gwendal Grignou <gwendal@chromium.org>
Date:   Tue Jul 28 13:48:25 2020 -0700

    iio: cros_ec: Set Gyroscope default frequency to 25Hz

    commit 336306790b2bbf7ce837625fa3b24ba724d05838 upstream.

    BMI160 Minimium gyroscope frequency in normal mode is 25Hz.
    When older EC firmware do not report their sensors frequencies,
    use 25Hz as the minimum for gyroscope to be sure it works on BMI160.

    Fixes: ae7b02ad2f32d ("iio: common: cros_ec_sensors: Expose cros_ec_sensors frequency
range via iio sysfs")
    Signed-off-by: Gwendal Grignou <gwendal@chromium.org>
    Reviewed-by: Enric Balletbo i Serra <enric.balletbo@collabora.com>
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 79d1a06c5940fafd0957d1029e9cc07009489d13
Author: Maxim Kochetkov <fido_max@inbox.ru>
Date:   Mon Aug 3 08:04:05 2020 +0300

    iio: adc: ti-ads1015: fix conversion when CONFIG_PM is not set

    commit e71e6dbe96ac80ac2aebe71a6a942e7bd60e7596 upstream.

    To stop conversion ads1015_set_power_state() function call unimplemented
    function __pm_runtime_suspend() from pm_runtime_put_autosuspend()
    if CONFIG_PM is not set.
    In case of CONFIG_PM is not set: __pm_runtime_suspend() returns -ENOSYS,
    so ads1015_read_raw() failed because ads1015_set_power_state() returns an
    error.

    If CONFIG_PM is disabled, there is no need to start/stop conversion.
    Fix it by adding return 0 function variant if CONFIG_PM is not set.

    Signed-off-by: Maxim Kochetkov <fido_max@inbox.ru>
    Fixes: ecc24e72f437 ("iio: adc: Add TI ADS1015 ADC driver support")
    Tested-by: Maxim Kiselev <bigunclemax@gmail.com>
    Reviewed-by: Andy Shevchenko <andriy.shevchenko@linux.intel.com>
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit bd9b60be76ef6e9c9c8cae2e61f1886a2f8515cd
Author: Angelo Compagnucci <angelo.compagnucci@gmail.com>
Date:   Tue Sep 1 11:32:18 2020 +0200

    iio: adc: mcp3422: fix locking on error path

    [ Upstream commit a139ffa40f0c24b753838b8ef3dcf6ad10eb7854 ]

    Reading from the chip should be unlocked on error path else the lock
    could never being released.

    Fixes: 07914c84ba30 ("iio: adc: Add driver for Microchip MCP3422/3/4 high resolution
ADC")
    Fixes: 3f1093d83d71 ("iio: adc: mcp3422: fix locking scope")
    Acked-by: Jonathan Cameron <jonathan.cameron@huawei.com>
    Signed-off-by: Angelo Compagnucci <angelo.compagnucci@gmail.com>
    Link: https://lore.kernel.org/r/20200901093218.1500845-1-angelo.compagnucci@gmail.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

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

commit 5deb6f82653d8b738865129a1c35379d21f3907c
Author: Angelo Compagnucci <angelo.compagnucci@gmail.com>
Date:   Wed Aug 19 09:55:25 2020 +0200

    iio: adc: mcp3422: fix locking scope

    commit 3f1093d83d7164e4705e4232ccf76da54adfda85 upstream.

    Locking should be held for the entire reading sequence involving setting
    the channel, waiting for the channel switch and reading from the
    channel.
    If not, reading from a channel can result mixing with the reading from
    another channel.

    Fixes: 07914c84ba30 ("iio: adc: Add driver for Microchip MCP3422/3/4 high resolution
ADC")
    Signed-off-by: Angelo Compagnucci <angelo.compagnucci@gmail.com>
    Link: https://lore.kernel.org/r/20200819075525.1395248-1-angelo.compagnucci@gmail.com
    Cc: <Stable@vger.kernel.org>
    Signed-off-by: Jonathan Cameron <Jonathan.Cameron@huawei.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit d9f376187cb1269850808d3af0c417eb489d48c8
Author: Leon Romanovsky <leon@kernel.org>
Date:   Fri Sep 4 18:58:08 2020 +0300

    gcov: Disable gcov build with GCC 10

    [ Upstream commit cfc905f158eaa099d6258031614d11869e7ef71c ]

    GCOV built with GCC 10 doesn't initialize n_function variable.  This
    produces different kernel panics as was seen by Colin in Ubuntu and me
    in FC 32.

    As a workaround, let's disable GCOV build for broken GCC 10 version.

    Link: https://bugs.launchpad.net/ubuntu/+source/linux/+bug/1891288
    Link: https://lore.kernel.org/lkml/20200827133932.3338519-1-leon@kernel.org
    Link: https://lore.kernel.org/lkml/CAHk-=whbijeSdSvx-
Xcr0DPMj0BiwhJ+uiNnDSVZcr_h_kg7UA@mail.gmail.com/
    Cc: Colin Ian King <colin.king@canonical.com>
    Signed-off-by: Leon Romanovsky <leonro@nvidia.com>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 790303eb397d54deb6d78a9b53521882e37c4c85
Author: Joerg Roedel <jroedel@suse.de>
Date:   Mon Aug 24 12:54:15 2020 +0200

    iommu/amd: Do not use IOMMUv2 functionality when SME is active

    [ Upstream commit 2822e582501b65707089b097e773e6fd70774841 ]

    When memory encryption is active the device is likely not in a direct
    mapped domain. Forbid using IOMMUv2 functionality for now until finer
    grained checks for this have been implemented.

    Signed-off-by: Joerg Roedel <jroedel@suse.de>
    Link: https://lore.kernel.org/r/20200824105415.21000-3-joro@8bytes.org
    Signed-off-by: Joerg Roedel <jroedel@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5aa21abcedd295a4e3338073f09f2c4849f11e39
```

```
Author: Joerg Roedel <jroedel@suse.de>
Date:   Mon Aug 24 12:54:14 2020 +0200

    iommu/amd: Do not force direct mapping when SME is active

    [ Upstream commit 7cad554887f1c5fd77e57e6bf4be38370c2160cb ]

    Do not force devices supporting IOMMUv2 to be direct mapped when memory
    encryption is active. This might cause them to be unusable because their
    DMA mask does not include the encryption bit.

    Signed-off-by: Joerg Roedel <jroedel@suse.de>
    Link: https://lore.kernel.org/r/20200824105415.21000-2-joro@8bytes.org
    Signed-off-by: Joerg Roedel <jroedel@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4898d0e064b9aa5d5697e4b5844384ed8ccddae0
Author: Sandeep Raghuraman <sandy.8925@gmail.com>
Date:   Thu Aug 27 18:43:37 2020 +0530

    drm/amdgpu: Fix bug in reporting voltage for CIK

    [ Upstream commit d98299885c9ea140c1108545186593deba36c4ac ]

    On my R9 390, the voltage was reported as a constant 1000 mV.
    This was due to a bug in smu7_hwmgr.c, in the smu7_read_sensor()
    function, where some magic constants were used in a condition,
    to determine whether the voltage should be read from PLANE2_VID
    or PLANE1_VID. The VDDC mask was incorrectly used, instead of
    the VDDGFX mask.

    This patch changes the code to use the correct defined constants
    (and apply the correct bitshift), thus resulting in correct voltage reporting.

    Signed-off-by: Sandeep Raghuraman <sandy.8925@gmail.com>
    Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e688f4fbb37eb15d363ed1ac94aab79f65d137f5
Author: Xie He <xie.he.0141@gmail.com>
Date:   Wed Sep 2 05:07:06 2020 -0700

    drivers/net/wan/hdlc: Change the default of hard_header_len to 0

    [ Upstream commit 2b7bcd967a0f5b7ac9bb0c37b92de36e073dd119 ]

    Change the default value of hard_header_len in hdlc.c from 16 to 0.

    Currently there are 6 HDLC protocol drivers, among them:

    hdlc_raw_eth, hdlc_cisco, hdlc_ppp, hdlc_x25 set hard_header_len when
    attaching the protocol, overriding the default. So this patch does not
    affect them.

    hdlc_raw and hdlc_fr don't set hard_header_len when attaching the
    protocol. So this patch will change the hard_header_len of the HDLC
    device for them from 16 to 0.

    This is the correct change because both hdlc_raw and hdlc_fr don't have
    header_ops, and the code in net/packet/af_packet.c expects the value of
    hard_header_len to be consistent with header_ops.

    In net/packet/af_packet.c, in the packet_snd function,
    for AF_PACKET/DGRAM sockets it would reserve a headroom of
    hard_header_len and call dev_hard_header to fill in that headroom,
```

```
    and for AF_PACKET/RAW sockets, it does not reserve the headroom and
    does not call dev_hard_header, but checks if the user has provided a
    header of length hard_header_len (in function dev_validate_header).

    Cc: Krzysztof Halasa <khc@pm.waw.pl>
    Cc: Martin Schiller <ms@dev.tdt.de>
    Signed-off-by: Xie He <xie.he.0141@gmail.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3b4c12838a4f0c32cc8fea8fd6de83236bdf81fe
Author: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com>
Date:   Wed Sep 2 18:42:50 2020 +0300

    ALSA: hda: use consistent HDAudio spelling in comments/docs

    [ Upstream commit b79de57b4378a93115307be6962d05b099eb0f37 ]

    We use HDaudio and HDAudio, pick one to make searches easier.
    No functionality change

    Also fix timestamping typo in documentation.

    Reported-by: Guennadi Liakhovetski <guennadi.liakhovetski@linux.intel.com>
    Signed-off-by: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com>
    Reviewed-by: Bard Liao <yung-chuan.liao@linux.intel.com>
    Reviewed-by: Guennadi Liakhovetski <guennadi.liakhovetski@linux.intel.com>
    Signed-off-by: Kai Vehmanen <kai.vehmanen@linux.intel.com>
    Link: https://lore.kernel.org/r/20200902154250.1440585-1-kai.vehmanen@linux.intel.com
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dd8ffea695c814dd064216e4a79676c8e9a72314
Author: Rander Wang <rander.wang@intel.com>
Date:   Wed Sep 2 18:42:18 2020 +0300

    ALSA: hda: fix a runtime pm issue in SOF when integrated GPU is disabled

    [ Upstream commit 13774d81f38538c5fa2924bdcdfa509155480fa6 ]

    In snd_hdac_device_init pm_runtime_set_active is called to
    increase child_count in parent device. But when it is failed
    to build connection with GPU for one case that integrated
    graphic gpu is disabled, snd_hdac_ext_bus_device_exit will be
    invoked to clean up a HD-audio extended codec base device. At
    this time the child_count of parent is not decreased, which
    makes parent device can't get suspended.

    This patch calls pm_runtime_set_suspended to decrease child_count
    in parent device in snd_hdac_device_exit to match with
    snd_hdac_device_init. pm_runtime_set_suspended can make sure that
    it will not decrease child_count if the device is already suspended.

    Signed-off-by: Rander Wang <rander.wang@intel.com>
    Reviewed-by: Ranjani Sridharan <ranjani.sridharan@linux.intel.com>
    Reviewed-by: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com>
    Reviewed-by: Bard Liao <yung-chuan.liao@linux.intel.com>
    Reviewed-by: Guennadi Liakhovetski <guennadi.liakhovetski@linux.intel.com>
    Signed-off-by: Kai Vehmanen <kai.vehmanen@linux.intel.com>
    Link: https://lore.kernel.org/r/20200902154218.1440441-1-kai.vehmanen@linux.intel.com
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ac9f804a8e5ab66802152c680e284b119be423d1
Author: Rander Wang <rander.wang@intel.com>
```

```
Date:   Wed Sep 2 18:42:07 2020 +0300

    ALSA: hda: hdmi - add Rocketlake support

    [ Upstream commit f804a324a41a880c1ab43cc5145d8b3e5790430d ]

    Add Rocketlake HDMI codec support. Rocketlake shares
    the pin-to-port mapping table with Tigerlake.

    Signed-off-by: Rander Wang <rander.wang@intel.com>
    Reviewed-by: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com>
    Reviewed-by: Ranjani Sridharan <ranjani.sridharan@linux.intel.com>
    Signed-off-by: Kai Vehmanen <kai.vehmanen@linux.intel.com>
    Link: https://lore.kernel.org/r/20200902154207.1440393-1-kai.vehmanen@linux.intel.com
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5e091ceb5084a9ee4c387b9b112ebf54e3c7cef8
Author: Jessica Yu <jeyu@kernel.org>
Date:   Tue Sep 1 18:00:16 2020 +0200

    arm64/module: set trampoline section flags regardless of CONFIG_DYNAMIC_FTRACE

    [ Upstream commit e0328feda79d9681b3e3245e6e180295550c8ee9 ]

    In the arm64 module linker script, the section .text.ftrace_trampoline
    is specified unconditionally regardless of whether CONFIG_DYNAMIC_FTRACE
    is enabled (this is simply due to the limitation that module linker
    scripts are not preprocessed like the vmlinux one).

    Normally, for .plt and .text.ftrace_trampoline, the section flags
    present in the module binary wouldn't matter since module_frob_arch_sections()
    would assign them manually anyway. However, the arm64 module loader only
    sets the section flags for .text.ftrace_trampoline when CONFIG_DYNAMIC_FTRACE=y.
    That's only become problematic recently due to a recent change in
    binutils-2.35, where the .text.ftrace_trampoline section (along with the
    .plt section) is now marked writable and executable (WAX).

    We no longer allow writable and executable sections to be loaded due to
    commit 5c3a7db0c7ec ("module: Harden STRICT_MODULE_RWX"), so this is
    causing all modules linked with binutils-2.35 to be rejected under arm64.
    Drop the IS_ENABLED(CONFIG_DYNAMIC_FTRACE) check in module_frob_arch_sections()
    so that the section flags for .text.ftrace_trampoline get properly set to
    SHF_EXECINSTR|SHF_ALLOC, without SHF_WRITE.

    Signed-off-by: Jessica Yu <jeyu@kernel.org>
    Acked-by: Will Deacon <will@kernel.org>
    Acked-by: Ard Biesheuvel <ardb@kernel.org>
    Link: http://lore.kernel.org/r/20200831094651.GA16385@linux-8ccs
    Link: https://lore.kernel.org/r/20200901160016.3646-1-jeyu@kernel.org
    Signed-off-by: Catalin Marinas <catalin.marinas@arm.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d1df83979d57b8bd1e01c43dc44c2fe61c5b6d26
Author: Francisco Jerez <currojerez@riseup.net>
Date:   Mon Aug 31 20:02:50 2020 -0700

    cpufreq: intel_pstate: Fix intel_pstate_get_hwp_max() for turbo disabled

    [ Upstream commit eacc9c5a927e474c173a5d53dd7fb8e306511768 ]

    This fixes the behavior of the scaling_max_freq and scaling_min_freq
    sysfs files in systems which had turbo disabled by the BIOS.

    Caleb noticed that the HWP is programmed to operate in the wrong
```

```
    P-state range on his system when the CPUFREQ policy min/max frequency
    is set via sysfs.  This seems to be because in his system
    intel_pstate_get_hwp_max() is returning the maximum turbo P-state even
    though turbo was disabled by the BIOS, which causes intel_pstate to
    scale kHz frequencies incorrectly e.g. setting the maximum turbo
    frequency whenever the maximum guaranteed frequency is requested via
    sysfs.

    Tested-by: Caleb Callaway <caleb.callaway@intel.com>
    Signed-off-by: Francisco Jerez <currojerez@riseup.net>
    Acked-by: Srinivas Pandruvada <srinivas.pandruvada@linux.intel.com>
    [ rjw: Minor subject edits ]
    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4b65cda38fc88c2796da2b19007f8f62447c8d85
Author: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
Date:   Thu Aug 20 17:40:02 2020 +0200

    cpufreq: intel_pstate: Refuse to turn off with HWP enabled

    [ Upstream commit 43298db3009f06fe5c69e1ca8b6cfc2565772fa1 ]

    After commit f6ebbcf08f37 ("cpufreq: intel_pstate: Implement passive
    mode with HWP enabled") it is possible to change the driver status
    to "off" via sysfs with HWP enabled, which effectively causes the
    driver to unregister itself, but HWP remains active and it forces the
    minimum performance, so even if another cpufreq driver is loaded,
    it will not be able to control the CPU frequency.

    For this reason, make the driver refuse to change the status to
    "off" with HWP enabled.

    Signed-off-by: Rafael J. Wysocki <rafael.j.wysocki@intel.com>
    Acked-by: Srinivas Pandruvada <srinivas.pandruvada@linux.intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5412a6f03aca512618921a9dbf02310424d0048f
Author: Evgeniy Didin <Evgeniy.Didin@synopsys.com>
Date:   Tue Jul 7 18:38:58 2020 +0300

    ARC: [plat-hsdk]: Switch ethernet phy-mode to rgmii-id

    [ Upstream commit 26907eb605fbc3ba9dbf888f21d9d8d04471271d ]

    HSDK board has Micrel KSZ9031, recent commit
    bcf3440c6dd ("net: phy: micrel: add phy-mode support for the KSZ9031 PHY")
    caused a breakdown of Ethernet.
    Using 'phy-mode = "rgmii"' is not correct because accodring RGMII
    specification it is necessary to have delay on RX (PHY to MAX)
    which is not generated in case of "rgmii".
    Using "rgmii-id" adds necessary delay and solves the issue.

    Also adding name of PHY placed on HSDK board.

    Signed-off-by: Evgeniy Didin <Evgeniy.Didin@synopsys.com>
    Cc: Eugeniy Paltsev <Eugeniy.Paltsev@synopsys.com>
    Cc: Alexey Brodkin <abrodkin@synopsys.com>
    Signed-off-by: Vineet Gupta <vgupta@synopsys.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit eb525e6a54990db54a39f0591505f80b67cc3c65
Author: Dinghao Liu <dinghao.liu@zju.edu.cn>
Date:   Mon Aug 31 17:06:43 2020 +0800
```

```
    HID: elan: Fix memleak in elan_input_configured

    [ Upstream commit b7429ea53d6c0936a0f10a5d64164f0aea440143 ]

    When input_mt_init_slots() fails, input should be freed
    to prevent memleak. When input_register_device() fails,
    we should call input_mt_destroy_slots() to free memory
    allocated by input_mt_init_slots().

    Signed-off-by: Dinghao Liu <dinghao.liu@zju.edu.cn>
    Signed-off-by: Jiri Kosina <jkosina@suse.cz>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dac76fe22da5bd38e03b0c25e5ba388ae1c5a545
Author: Xie He <xie.he.0141@gmail.com>
Date:   Fri Aug 28 00:07:52 2020 -0700

    drivers/net/wan/hdlc_cisco: Add hard_header_len

    [ Upstream commit 1a545ebe380bf4c1433e3c136e35a77764fda5ad ]

    This driver didn't set hard_header_len. This patch sets hard_header_len
    for it according to its header_ops->create function.

    This driver's header_ops->create function (cisco_hard_header) creates
    a header of (struct hdlc_header), so hard_header_len should be set to
    sizeof(struct hdlc_header).

    Cc: Martin Schiller <ms@dev.tdt.de>
    Signed-off-by: Xie He <xie.he.0141@gmail.com>
    Acked-by: Krzysztof Halasa <khc@pm.waw.pl>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 80b7267b31270653a79c78ca4ca62966d62867e3
Author: Nicholas Miell <nmiell@gmail.com>
Date:   Fri Aug 28 21:14:29 2020 -0700

    HID: microsoft: Add rumble support for the 8bitdo SN30 Pro+ controller

    [ Upstream commit 724a419ea28f7514a391e80040230f69cf626707 ]

    When operating in XInput mode, the 8bitdo SN30 Pro+ requires the same
    quirk as the official Xbox One Bluetooth controllers for rumble to
    function.

    Other controllers like the N30 Pro 2, SF30 Pro, SN30 Pro, etc. probably
    also need this quirk, but I do not have the hardware to test.

    Signed-off-by: Nicholas Miell <nmiell@gmail.com>
    Signed-off-by: Jiri Kosina <jkosina@suse.cz>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 38ce1ae8e2c8f318bba99f04e9bda47ff4085930
Author: Nirenjan Krishnan <nirenjan@gmail.com>
Date:   Sun Aug 30 17:48:59 2020 -0700

    HID: quirks: Set INCREMENT_USAGE_ON_DUPLICATE for all Saitek X52 devices

    [ Upstream commit 77df710ba633dfb6c65c65cf99ea9e084a1c9933 ]

    The Saitek X52 family of joysticks has a pair of axes that were
    originally (by the Windows driver) used as mouse pointer controls. The
    corresponding usage page is the Game Controls page, which is not
    recognized by the generic HID driver, and therefore, both axes get
```

```
    mapped to ABS_MISC. The quirk makes the second axis get mapped to
    ABS_MISC+1, and therefore made available separately.

    One Saitek X52 device is already fixed. This patch fixes the other two
    known devices with VID/PID 06a3:0255 and 06a3:0762.

    Signed-off-by: Nirenjan Krishnan <nirenjan@gmail.com>
    Signed-off-by: Jiri Kosina <jkosina@suse.cz>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d3a6cfff8e6a9b456d6285956012baa3037c5cc5
Author: Tong Zhang <ztong0001@gmail.com>
Date:   Fri Aug 28 10:17:08 2020 -0400

    nvme-pci: cancel nvme device request before disabling

    [ Upstream commit 7ad92f656bddff4cf8f641e0e3b1acd4eb9644cb ]

    This patch addresses an irq free warning and null pointer dereference
    error problem when nvme devices got timeout error during initialization.
    This problem happens when nvme_timeout() function is called while
    nvme_reset_work() is still in execution. This patch fixed the problem by
    setting flag of the problematic request to NVME_REQ_CANCELLED before
    calling nvme_dev_disable() to make sure __nvme_submit_sync_cmd() returns
    an error code and let nvme_submit_sync_cmd() fail gracefully.
    The following is console output.

    [   62.472097] nvme nvme0: I/O 13 QID 0 timeout, disable controller
    [   62.488796] nvme nvme0: could not set timestamp (881)
    [   62.494888] ------------[ cut here ]------------
    [   62.495142] Trying to free already-free IRQ 11
    [   62.495366] WARNING: CPU: 0 PID: 7 at kernel/irq/manage.c:1751 free_irq+0x1f7/0x370
    [   62.495742] Modules linked in:
    [   62.495902] CPU: 0 PID: 7 Comm: kworker/u4:0 Not tainted 5.8.0+ #8
    [   62.496206] Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS rel-1.13.0-
48-gd9c812dda519-p4
    [   62.496772] Workqueue: nvme-reset-wq nvme_reset_work
    [   62.497019] RIP: 0010:free_irq+0x1f7/0x370
    [   62.497223] Code: e8 ce 49 11 00 48 83 c4 08 4c 89 e0 5b 5d 41 5c 41 5d 41 5e 41 5f
c3 44 89 f6 48 c70
    [   62.498133] RSP: 0000:ffffa96800043d40 EFLAGS: 00010086
    [   62.498391] RAX: 0000000000000000 RBX: ffff9b87fc458400 RCX: 0000000000000000
    [   62.498741] RDX: 0000000000000001 RSI: 0000000000000096 RDI: ffffffff9693d72c
    [   62.499091] RBP: ffff9b87fd4c8f60 R08: ffffa96800043bfd R09: 0000000000000163
    [   62.499440] R10: ffffa96800043bf8 R11: ffffa96800043bfd R12: ffff9b87fd4c8e00
    [   62.499790] R13: ffff9b87fd4c8ea4 R14: 000000000000000b R15: ffff9b87fd76b000
    [   62.500140] FS:  0000000000000000(0000) GS:ffff9b87fdc00000(0000)
knlGS:0000000000000000
    [   62.500534] CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    [   62.500816] CR2: 0000000000000000 CR3: 000000003aa0a000 CR4: 00000000000006f0
    [   62.501165] DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000
    [   62.501515] DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400
    [   62.501864] Call Trace:
    [   62.501993]  pci_free_irq+0x13/0x20
    [   62.502167]  nvme_reset_work+0x5d0/0x12a0
    [   62.502369]  ? update_load_avg+0x59/0x580
    [   62.502569]  ? ttwu_queue_wakelist+0xa8/0xc0
    [   62.502780]  ? try_to_wake_up+0x1a2/0x450
    [   62.502979]  process_one_work+0x1d2/0x390
    [   62.503179]  worker_thread+0x45/0x3b0
    [   62.503361]  ? process_one_work+0x390/0x390
    [   62.503568]  kthread+0xf9/0x130
    [   62.503726]  ? kthread_park+0x80/0x80
    [   62.503911]  ret_from_fork+0x22/0x30
    [   62.504090] ---[ end trace de9ed4a70f8d71e2 ]---
```

```
    [  123.912275] nvme nvme0: I/O 12 QID 0 timeout, disable controller
    [  123.914670] nvme nvme0: 1/0/0 default/read/poll queues
    [  123.916310] BUG: kernel NULL pointer dereference, address: 0000000000000000
    [  123.917469] #PF: supervisor write access in kernel mode
    [  123.917725] #PF: error_code(0x0002) - not-present page
    [  123.917976] PGD 0 P4D 0
    [  123.918109] Oops: 0002 [#1] SMP PTI
    [  123.918283] CPU: 0 PID: 7 Comm: kworker/u4:0 Tainted: G        W         5.8.0+ #8
    [  123.918650] Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS rel-1.13.0-
48-gd9c812dda519-p4
    [  123.919219] Workqueue: nvme-reset-wq nvme_reset_work
    [  123.919469] RIP: 0010:__blk_mq_alloc_map_and_request+0x21/0x80
    [  123.919757] Code: 66 0f 1f 84 00 00 00 00 00 41 55 41 54 55 48 63 ee 53 48 8b 47 68
89 ee 48 89 fb 8b4
    [  123.920657] RSP: 0000:ffffa96800043d40 EFLAGS: 00010286
    [  123.920912] RAX: ffff9b87fc4fee40 RBX: ffff9b87fc8cb008 RCX: 0000000000000000
    [  123.921258] RDX: 0000000000000000 RSI: 0000000000000000 RDI: ffff9b87fc618000
    [  123.921602] RBP: 0000000000000000 R08: ffff9b87fdc2c4a0 R09: ffff9b87fc616000
    [  123.921949] R10: 0000000000000000 R11: ffff9b87fffd1500 R12: 0000000000000000
    [  123.922295] R13: 0000000000000000 R14: ffff9b87fc8cb200 R15: ffff9b87fc8cb000
    [  123.922641] FS:  0000000000000000(0000) GS:ffff9b87fdc00000(0000)
knlGS:0000000000000000
    [  123.923032] CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    [  123.923312] CR2: 0000000000000000 CR3: 000000003aa0a000 CR4: 00000000000006f0
    [  123.923660] DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000
    [  123.924007] DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400
    [  123.924353] Call Trace:
    [  123.924479]  blk_mq_alloc_tag_set+0x137/0x2a0
    [  123.924694]  nvme_reset_work+0xed6/0x12a0
    [  123.924898]  process_one_work+0x1d2/0x390
    [  123.925099]  worker_thread+0x45/0x3b0
    [  123.925280]  ? process_one_work+0x390/0x390
    [  123.925486]  kthread+0xf9/0x130
    [  123.925642]  ? kthread_park+0x80/0x80
    [  123.925825]  ret_from_fork+0x22/0x30
    [  123.926004] Modules linked in:
    [  123.926158] CR2: 0000000000000000
    [  123.926322] ---[ end trace de9ed4a70f8d71e3 ]---
    [  123.926549] RIP: 0010:__blk_mq_alloc_map_and_request+0x21/0x80
    [  123.926832] Code: 66 0f 1f 84 00 00 00 00 00 41 55 41 54 55 48 63 ee 53 48 8b 47 68
89 ee 48 89 fb 8b4
    [  123.927734] RSP: 0000:ffffa96800043d40 EFLAGS: 00010286
    [  123.927989] RAX: ffff9b87fc4fee40 RBX: ffff9b87fc8cb008 RCX: 0000000000000000
    [  123.928336] RDX: 0000000000000000 RSI: 0000000000000000 RDI: ffff9b87fc618000
    [  123.928679] RBP: 0000000000000000 R08: ffff9b87fdc2c4a0 R09: ffff9b87fc616000
    [  123.929025] R10: 0000000000000000 R11: ffff9b87fffd1500 R12: 0000000000000000
    [  123.929370] R13: 0000000000000000 R14: ffff9b87fc8cb200 R15: ffff9b87fc8cb000
    [  123.929715] FS:  0000000000000000(0000) GS:ffff9b87fdc00000(0000)
knlGS:0000000000000000
    [  123.930106] CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
    [  123.930384] CR2: 0000000000000000 CR3: 000000003aa0a000 CR4: 00000000000006f0
    [  123.930731] DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000
    [  123.931077] DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400

    Co-developed-by: Keith Busch <kbusch@kernel.org>
    Signed-off-by: Tong Zhang <ztong0001@gmail.com>
    Reviewed-by: Keith Busch <kbusch@kernel.org>
    Signed-off-by: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e625888ac2aa6c59ec8564785ab0a0bd90a96085
Author: Sagi Grimberg <sagi@grimberg.me>
Date:   Thu Jul 30 13:42:42 2020 -0700

    nvme-rdma: fix reset hang if controller died in the middle of a reset
```

```
    [ Upstream commit 2362acb6785611eda795bfc12e1ea6b202ecf62c ]

    If the controller becomes unresponsive in the middle of a reset, we
    will hang because we are waiting for the freeze to complete, but that
    cannot happen since we have commands that are inflight holding the
    q_usage_counter, and we can't blindly fail requests that times out.

    So give a timeout and if we cannot wait for queue freeze before
    unfreezing, fail and have the error handling take care how to
    proceed (either schedule a reconnect of remove the controller).

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

commit ca46ff644ab0324eb0de939155a08122f97523b0
Author: Sagi Grimberg <sagi@grimberg.me>
Date:   Wed Jul 29 02:36:03 2020 -0700

    nvme-rdma: fix timeout handler

    [ Upstream commit 0475a8dcbcee92a5d22e40c9c6353829fc6294b8 ]

    When a request times out in a LIVE state, we simply trigger error
    recovery and let the error recovery handle the request cancellation,
    however when a request times out in a non LIVE state, we make sure to
    complete it immediately as it might block controller setup or teardown
    and prevent forward progress.

    However tearing down the entire set of I/O and admin queues causes
    freeze/unfreeze imbalance (q->mq_freeze_depth) because and is really
    an overkill to what we actually need, which is to just fence controller
    teardown that may be running, stop the queue, and cancel the request if
    it is not already completed.

    Now that we have the controller teardown_lock, we can safely serialize
    request cancellation. This addresses a hang caused by calling extra
    queue freeze on controller namespaces, causing unfreeze to not complete
    correctly.

    Reviewed-by: Christoph Hellwig <hch@lst.de>
    Reviewed-by: James Smart <james.smart@broadcom.com>
    Signed-off-by: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3341cac774c144e61a69b59677628671e384d6fc
Author: Sagi Grimberg <sagi@grimberg.me>
Date:   Wed Aug 5 18:13:58 2020 -0700

    nvme-rdma: serialize controller teardown sequences

    [ Upstream commit 5110f40241d08334375eb0495f174b1d2c07657e ]

    In the timeout handler we may need to complete a request because the
    request that timed out may be an I/O that is a part of a serial sequence
    of controller teardown or initialization. In order to complete the
    request, we need to fence any other context that may compete with us
    and complete the request that is timing out.

    In this case, we could have a potential double completion in case
    a hard-irq or a different competing context triggered error recovery
    and is running inflight request cancellation concurrently with the
    timeout handler.
```

```
    Protect using a ctrl teardown_lock to serialize contexts that may
    complete a cancelled request due to error recovery or a reset.

    Reviewed-by: Christoph Hellwig <hch@lst.de>
    Reviewed-by: James Smart <james.smart@broadcom.com>
    Signed-off-by: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ed67604770cd486482da46c8f5c7d5c613d4becb
Author: Sagi Grimberg <sagi@grimberg.me>
Date:   Thu Jul 30 13:25:34 2020 -0700

    nvme-tcp: fix reset hang if controller died in the middle of a reset

    [ Upstream commit e5c01f4f7f623e768e868bcc08d8e7ceb03b75d0 ]

    If the controller becomes unresponsive in the middle of a reset, we will
    hang because we are waiting for the freeze to complete, but that cannot
    happen since we have commands that are inflight holding the
    q_usage_counter, and we can't blindly fail requests that times out.

    So give a timeout and if we cannot wait for queue freeze before
    unfreezing, fail and have the error handling take care how to proceed
    (either schedule a reconnect of remove the controller).

    Signed-off-by: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 18aca936e2fffd06a330dc5d88ec31fbed22901f
Author: Sagi Grimberg <sagi@grimberg.me>
Date:   Tue Jul 28 13:16:36 2020 -0700

    nvme-tcp: fix timeout handler

    [ Upstream commit 236187c4ed195161dfa4237c7beffbba0c5ae45b ]

    When a request times out in a LIVE state, we simply trigger error
    recovery and let the error recovery handle the request cancellation,
    however when a request times out in a non LIVE state, we make sure to
    complete it immediately as it might block controller setup or teardown
    and prevent forward progress.

    However tearing down the entire set of I/O and admin queues causes
    freeze/unfreeze imbalance (q->mq_freeze_depth) because and is really
    an overkill to what we actually need, which is to just fence controller
    teardown that may be running, stop the queue, and cancel the request if
    it is not already completed.

    Now that we have the controller teardown_lock, we can safely serialize
    request cancellation. This addresses a hang caused by calling extra
    queue freeze on controller namespaces, causing unfreeze to not complete
    correctly.

    Signed-off-by: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c0e1cf0f6ac2aef75110465dfa8de8341cdbf2b5
Author: Sagi Grimberg <sagi@grimberg.me>
Date:   Wed Aug 5 18:13:48 2020 -0700

    nvme-tcp: serialize controller teardown sequences

    [ Upstream commit d4d61470ae48838f49e668503e840e1520b97162 ]

    In the timeout handler we may need to complete a request because the
```

```
    request that timed out may be an I/O that is a part of a serial sequence
    of controller teardown or initialization. In order to complete the
    request, we need to fence any other context that may compete with us
    and complete the request that is timing out.

    In this case, we could have a potential double completion in case
    a hard-irq or a different competing context triggered error recovery
    and is running inflight request cancellation concurrently with the
    timeout handler.

    Protect using a ctrl teardown_lock to serialize contexts that may
    complete a cancelled request due to error recovery or a reset.

    Signed-off-by: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c244c493f079334874f9bac97a9b06cb86d4e4f1
Author: Sagi Grimberg <sagi@grimberg.me>
Date:   Thu Jul 30 13:24:45 2020 -0700

    nvme: have nvme_wait_freeze_timeout return if it timed out

    [ Upstream commit 7cf0d7c0f3c3b0203aaf81c1bc884924d8fdb9bd ]

    Users can detect if the wait has completed or not and take appropriate
    actions based on this information (e.g. weather to continue
    initialization or rather fail and schedule another initialization
    attempt).

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

commit 6c63e56986d3bfd281bec617c5ad0149a0f50fd7
Author: Sagi Grimberg <sagi@grimberg.me>
Date:   Fri Aug 14 11:46:51 2020 -0700

    nvme-fabrics: don't check state NVME_CTRL_NEW for request acceptance

    [ Upstream commit d7144f5c4cf4de95fdc3422943cf51c06aeaf7a7 ]

    NVME_CTRL_NEW should never see any I/O, because in order to start
    initialization it has to transition to NVME_CTRL_CONNECTING and from
    there it will never return to this state.

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

commit 0a12c794513ccb18d2d2e1b5ef73d0536a403d25
Author: Ziye Yang <ziye.yang@intel.com>
Date:   Sat Aug 22 00:48:10 2020 +0800

    nvmet-tcp: Fix NULL dereference when a connect data comes in h2cdata pdu

    [ Upstream commit a6ce7d7b4adaebc27ee7e78e5ecc378a1cfc221d ]

    When handling commands without in-capsule data, we assign the ttag
    assuming we already have the queue commands array allocated (based
    on the queue size information in the connect data payload). However
    if the connect itself did not send the connect data in-capsule we
    have yet to allocate the queue commands,and we will assign a bogus
    ttag and suffer a NULL dereference when we receive the corresponding
    h2cdata pdu.
```

```
    Fix this by checking if we already allocated commands before
    dereferencing it when handling h2cdata, if we didn't, its for sure a
    connect and we should use the preallocated connect command.

    Signed-off-by: Ziye Yang <ziye.yang@intel.com>
    Signed-off-by: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 76101def01209fb491b9e03f73e67c95914e0fef
Author: Sean Young <sean@mess.org>
Date:   Thu Aug 13 11:08:49 2020 +0200

    media: gpio-ir-tx: spinlock is not needed to disable interrupts

    [ Upstream commit 1451b93223bbe3b4e9c91fca6b451d00667c5bf0 ]

    During bit-banging the IR on a gpio pin, we cannot be scheduled or have
    anything interrupt us, else the generated signal will be incorrect.
    Therefore, we need to disable interrupts on the local cpu. This also
    disables preemption.

    local_irq_disable() does exactly what we need and does not require a
    spinlock.

    Signed-off-by: Sean Young <sean@mess.org>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+huawei@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 92fb4a758e05a738fb4f4ffa2d60884be6d7d375
Author: Vineet Gupta <vgupta@synopsys.com>
Date:   Mon Aug 24 12:10:33 2020 -0700

    irqchip/eznps: Fix build error for !ARC700 builds

    [ Upstream commit 89d29997f103d08264b0685796b420d911658b96 ]

    eznps driver is supposed to be platform independent however it ends up
    including stuff from inside arch/arc headers leading to rand config
    build errors.

    The quick hack to fix this (proper fix is too much chrun for non active
    user-base) is to add following to nps platform agnostic header.
     - copy AUX_IENABLE from arch/arc header
     - move CTOP_AUX_IACK from arch/arc/plat-eznps/*/**

    Reported-by: kernel test robot <lkp@intel.com>
    Reported-by: Sebastian Andrzej Siewior <bigeasy@linutronix.de>
    Link: https://lkml.kernel.org/r/20200824095831.5lpkmkafelnvlpi2@linutronix.de
    Signed-off-by: Vineet Gupta <vgupta@synopsys.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 4422ef3886450d9a5e2a4b4a30f8f9be3639123b
Author: Vineet Gupta <vgupta@synopsys.com>
Date:   Fri Aug 7 21:29:28 2020 -0700

    ARC: show_regs: fix r12 printing and simplify

    [ Upstream commit e5c388b4b967037a0e00b60194b0dbcf94881a9b ]

    when working on ARC64, spotted an issue in ARCv2 reg file printing.
    print_reg_file() assumes contiguous reg-file whereas in ARCv2 they are
    not: r12 comes before r0-r11 due to hardware auto-save. Apparently this
    issue has been present since v2 port submission.

    To avoid bolting hacks for this discontinuity while looping through
```

```
    pt_regs, just ditching the loop and print pt_regs directly.

    Signed-off-by: Vineet Gupta <vgupta@synopsys.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 74e4b9ae64a2749bd95fed1e700d44c36e7aadeb
Author: Darrick J. Wong <darrick.wong@oracle.com>
Date:   Wed Aug 26 14:12:18 2020 -0700

    xfs: initialize the shortform attr header padding entry

    [ Upstream commit 125eac243806e021f33a1fdea3687eccbb9f7636 ]

    Don't leak kernel memory contents into the shortform attr fork.

    Signed-off-by: Darrick J. Wong <darrick.wong@oracle.com>
    Reviewed-by: Eric Sandeen <sandeen@redhat.com>
    Reviewed-by: Dave Chinner <dchinner@redhat.com>
    Reviewed-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 01199dce4d445fc620161abf771ea54a3d24930e
Author: Amar Singhal <asinghal@codeaurora.org>
Date:   Fri Jun 19 13:52:01 2020 -0700

    cfg80211: Adjust 6 GHz frequency to channel conversion

    [ Upstream commit 2d9b55508556ccee6410310fb9ea2482fd3328eb ]

    Adjust the 6 GHz frequency to channel conversion function,
    the other way around was previously handled.

    Signed-off-by: Amar Singhal <asinghal@codeaurora.org>
    Link: https://lore.kernel.org/r/1592599921-10607-1-git-send-email-
asinghal@codeaurora.org
    [rewrite commit message, hard-code channel 2]
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3a8fcf73cbf1027813828f29b24d8858f31a9755
Author: Felix Fietkau <nbd@nbd.name>
Date:   Sat Aug 8 19:25:42 2020 +0200

    mac80211: reduce packet loss event false positives

    [ Upstream commit 47df8e059b49a80c179fae39256bcd7096810934 ]

    When running a large number of packets per second with a high data rate
    and long A-MPDUs, the packet loss threshold can be reached very quickly
    when the link conditions change. This frequently shows up as spurious
    disconnects.
    Mitigate false positives by using a similar logic for regular stations
    as the one being used for TDLS, though with a more aggressive timeout.
    Packet loss events are only reported if no ACK was received for a second.

    Signed-off-by: Felix Fietkau <nbd@nbd.name>
    Link: https://lore.kernel.org/r/20200808172542.41628-1-nbd@nbd.name
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 55de6d85df5de6ff6d2d79937aa7694772a110aa
Author: Shay Bar <shay.bar@celeno.com>
Date:   Wed Aug 26 17:31:39 2020 +0300

    wireless: fix wrong 160/80+80 MHz setting
```

```
    [ Upstream commit 3579994476b65cb5e272ff0f720a1fd31322e53f ]

    Fix cfg80211_chandef_usable():
    consider IEEE80211_VHT_CAP_EXT_NSS_BW when verifying 160/80+80 MHz.

    Based on:
    "Table 9-272 — Setting of the Supported Channel Width Set subfield and Extended NSS BW
    Support subfield at a STA transmitting the VHT Capabilities Information field"
    From "Draft P802.11REVmd_D3.0.pdf"

    Signed-off-by: Aviad Brikman <aviad.brikman@celeno.com>
    Signed-off-by: Shay Bar <shay.bar@celeno.com>
    Link: https://lore.kernel.org/r/20200826143139.25976-1-shay.bar@celeno.com
    [reformat the code a bit and use u32_get_bits()]
    Signed-off-by: Johannes Berg <johannes.berg@intel.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3fd5156d7fbd0510d8baeeeffaa1a4b8b84e5c9f
Author: Xie He <xie.he.0141@gmail.com>
Date:   Tue Aug 25 20:03:53 2020 -0700

    drivers/net/wan/lapbether: Set network_header before transmitting

    [ Upstream commit 91244d108441013b7367b3b4dcc6869998676473 ]

    Set the skb's network_header before it is passed to the underlying
    Ethernet device for transmission.

    This patch fixes the following issue:

    When we use this driver with AF_PACKET sockets, there would be error
    messages of:
       protocol 0805 is buggy, dev (Ethernet interface name)
    printed in the system "dmesg" log.

    This is because skbs passed down to the Ethernet device for transmission
    don't have their network_header properly set, and the dev_queue_xmit_nit
    function in net/core/dev.c complains about this.

    Reason of setting the network_header to this place (at the end of the
    Ethernet header, and at the beginning of the Ethernet payload):

    Because when this driver receives an skb from the Ethernet device, the
    network_header is also set at this place.

    Cc: Martin Schiller <ms@dev.tdt.de>
    Signed-off-by: Xie He <xie.he.0141@gmail.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d910490078ac70ba4223af4c22d7404737d85425
Author: Brian Foster <bfoster@redhat.com>
Date:   Wed Aug 26 14:08:27 2020 -0700

    xfs: fix off-by-one in inode alloc block reservation calculation

    [ Upstream commit 657f101930bc6c5b41bd7d6c22565c4302a80d33 ]

    The inode chunk allocation transaction reserves inobt_maxlevels-1
    blocks to accommodate a full split of the inode btree. A full split
    requires an allocation for every existing level and a new root
    block, which means inobt_maxlevels is the worst case block
    requirement for a transaction that inserts to the inobt. This can
    lead to a transaction block reservation overrun when tmpfile
```

```
    creation allocates an inode chunk and expands the inobt to its
    maximum depth. This problem has been observed in conjunction with
    overlayfs, which makes frequent use of tmpfiles internally.

    The existing reservation code goes back as far as the Linux git repo
    history (v2.6.12). It was likely never observed as a problem because
    the traditional file/directory creation transactions also include
    worst case block reservation for directory modifications, which most
    likely is able to make up for a single block deficiency in the inode
    allocation portion of the calculation. tmpfile support is relatively
    more recent (v3.15), less heavily used, and only includes the inode
    allocation block reservation as tmpfiles aren't linked into the
    directory tree on creation.

    Fix up the inode alloc block reservation macro and a couple of the
    block allocator minleft parameters that enforce an allocation to
    leave enough free blocks in the AG for a full inobt split.

    Signed-off-by: Brian Foster <bfoster@redhat.com>
    Reviewed-by: Darrick J. Wong <darrick.wong@oracle.com>
    Signed-off-by: Darrick J. Wong <darrick.wong@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c63f823048a7cf67f51ea96518fb9df951642662
Author: Yi Li <yili@winhong.com>
Date:   Wed Aug 26 13:11:50 2020 +0800

    net: hns3: Fix for geneve tx checksum bug

    [ Upstream commit a156998fc92d3859c8e820f1583f6d0541d643c3 ]

    when skb->encapsulation is 0, skb->ip_summed is CHECKSUM_PARTIAL
    and it is udp packet, which has a dest port as the IANA assigned.
    the hardware is expected to do the checksum offload, but the
    hardware will not do the checksum offload when udp dest port is
    6081.

    This patch fixes it by doing the checksum in software.

    Reported-by: Li Bing <libing@winhong.com>
    Signed-off-by: Yi Li <yili@winhong.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 79a1d64700261bf1ebeb90c7da630b4373a9bb14
Author: Madhuparna Bhowmik <madhuparnabhowmik10@gmail.com>
Date:   Fri Aug 21 09:14:23 2020 +0530

    drivers/dma/dma-jz4780: Fix race condition between probe and irq handler

    [ Upstream commit 6d6018fc30bee67290dbed2fa51123f7c6f3d691 ]

    In probe, IRQ is requested before zchan->id is initialized which can be
    read in the irq handler. Hence, shift request irq after other initializations
    complete.

    Found by Linux Driver Verification project (linuxtesting.org).

    Signed-off-by: Madhuparna Bhowmik <madhuparnabhowmik10@gmail.com>
    Reviewed-by: Paul Cercueil <paul@crapouillou.net>
    Link: https://lore.kernel.org/r/20200821034423.12713-1-madhuparnabhowmik10@gmail.com
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7a08699028e409a08b57c02268108053bae320e3
```

```
Author: Mohan Kumar <mkumard@nvidia.com>
Date:   Tue Aug 25 10:54:15 2020 +0530

    ALSA: hda/tegra: Program WAKEEN register for Tegra

    [ Upstream commit 23d63a31d9f44d7daeac0d1fb65c6a73c70e5216 ]

    The WAKEEN bits are used to indicate which bits in the
    STATESTS register may cause wake event during the codec
    state change request. Configure the WAKEEN register for
    the Tegra to detect the wake events.

    Signed-off-by: Mohan Kumar <mkumard@nvidia.com>
    Acked-by: Sameer Pujar <spujar@nvidia.com>
    Link: https://lore.kernel.org/r/20200825052415.20626-3-mkumard@nvidia.com
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 92c93dacd804535fe4f75779e5329c8d8c5fe18c
Author: Mohan Kumar <mkumard@nvidia.com>
Date:   Tue Aug 25 10:54:14 2020 +0530

    ALSA: hda: Fix 2 channel swapping for Tegra

    [ Upstream commit 216116eae43963c662eb84729507bad95214ca6b ]

    The Tegra HDA codec HW implementation has an issue related to not
    swapping the 2 channel Audio Sample Packet(ASP) channel mapping.
    Whatever the FL and FR mapping specified the left channel always
    comes out of left speaker and right channel on right speaker. So
    add condition to disallow the swapping of FL,FR during the playback.

    Signed-off-by: Mohan Kumar <mkumard@nvidia.com>
    Acked-by: Sameer Pujar <spujar@nvidia.com>
    Link: https://lore.kernel.org/r/20200825052415.20626-2-mkumard@nvidia.com
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 3e7682611f075a9eff41b23b71e275ad2a3b5ec2
Author: Ye Bin <yebin10@huawei.com>
Date:   Mon Aug 24 11:34:36 2020 +0800

    scsi: qedf: Fix null ptr reference in qedf_stag_change_work

    [ Upstream commit f308a35f547cd7c1d8a901c12b3ac508e96df665 ]

    Link: https://lore.kernel.org/r/20200824033436.45570-1-yebin10@huawei.com
    Acked-by: Saurav Kashyap <skashyap@marvell.com>
    Signed-off-by: Ye Bin <yebin10@huawei.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6ee6851f8704badb3beef274e8c9626248763d10
Author: Dinghao Liu <dinghao.liu@zju.edu.cn>
Date:   Sun Aug 23 19:29:35 2020 +0800

    firestream: Fix memleak in fs_open

    [ Upstream commit 15ac5cdafb9202424206dc5bd376437a358963f9 ]

    When make_rate() fails, vcc should be freed just
    like other error paths in fs_open().

    Signed-off-by: Dinghao Liu <dinghao.liu@zju.edu.cn>
    Signed-off-by: David S. Miller <davem@davemloft.net>
```

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

commit 74bbd3069881c82b50b6c424813445552e33760b
Author: Dinghao Liu <dinghao.liu@zju.edu.cn>
Date:   Sun Aug 23 15:23:43 2020 +0800

    NFC: st95hf: Fix memleak in st95hf_in_send_cmd

    [ Upstream commit f97c04c316d8fea16dca449fdfbe101fbdfee6a2 ]

    When down_killable() fails, skb_resp should be freed
    just like when st95hf_spi_send() fails.

    Signed-off-by: Dinghao Liu <dinghao.liu@zju.edu.cn>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit ca9e7f16d783a12909d1ef6ce36fe5fd09b5c86d
Author: Xie He <xie.he.0141@gmail.com>
Date:   Fri Aug 21 14:26:59 2020 -0700

    drivers/net/wan/lapbether: Added needed_tailroom

    [ Upstream commit 1ee39c1448c4e0d480c5b390e2db1987561fb5c2 ]

    The underlying Ethernet device may request necessary tailroom to be
    allocated by setting needed_tailroom. This driver should also set
    needed_tailroom to request the tailroom needed by the underlying
    Ethernet device to be allocated.

    Cc: Willem de Bruijn <willemdebruijn.kernel@gmail.com>
    Cc: Martin Schiller <ms@dev.tdt.de>
    Signed-off-by: Xie He <xie.he.0141@gmail.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1f5a4b08aec3677e95c4c0f5a7e98b66e5969e45
Author: Stefano Brivio <sbrivio@redhat.com>
Date:   Wed Aug 19 23:59:15 2020 +0200

    netfilter: nft_set_rbtree: Detect partial overlap with start endpoint match

    [ Upstream commit 0726763043dc10dd4c12481f050b1a5ef8f15410 ]

    Getting creative with nft and omitting the interval_overlap()
    check from the set_overlap() function, without omitting
    set_overlap() altogether, led to the observation of a partial
    overlap that wasn't detected, and would actually result in
    replacement of the end element of an existing interval.

    This is due to the fact that we'll return -EEXIST on a matching,
    pre-existing start element, instead of -ENOTEMPTY, and the error
    is cleared by API if NLM_F_EXCL is not given. At this point, we
    can insert a matching start, and duplicate the end element as long
    as we don't end up into other intervals.

    For instance, inserting interval 0 - 2 with an existing 0 - 3
    interval would result in a single 0 - 2 interval, and a dangling
    '3' end element. This is because nft will proceed after inserting
    the '0' start element as no error is reported, and no further
    conflicting intervals are detected on insertion of the end element.

    This needs a different approach as it's a local condition that can
    be detected by looking for duplicate ends coming from left and
    right, separately. Track those and directly report -ENOTEMPTY on
```

```
    duplicated end elements for a matching start.

    Signed-off-by: Stefano Brivio <sbrivio@redhat.com>
    Signed-off-by: Pablo Neira Ayuso <pablo@netfilter.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a0a91436d588bc8da4f11502200299fa4197b8f5
Author: Florian Westphal <fw@strlen.de>
Date:   Tue Aug 18 16:15:58 2020 +0200

    netfilter: conntrack: allow sctp hearbeat after connection re-use

    [ Upstream commit cc5453a5b7e90c39f713091a7ebc53c1f87d1700 ]

    If an sctp connection gets re-used, heartbeats are flagged as invalid
    because their vtag doesn't match.

    Handle this in a similar way as TCP conntrack when it suspects that the
    endpoints and conntrack are out-of-sync.

    When a HEARTBEAT request fails its vtag validation, flag this in the
    conntrack state and accept the packet.

    When a HEARTBEAT_ACK is received with an invalid vtag in the reverse
    direction after we allowed such a HEARTBEAT through, assume we are
    out-of-sync and re-set the vtag info.

    v2: remove left-over snippet from an older incarnation that moved
        new_state/old_state assignments, thats not needed so keep that
        as-is.

    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 83cadef120e6f745e51df04aa40598405aa1d1c8
Author: Jiaxun Yang <jiaxun.yang@flygoat.com>
Date:   Sat Aug 8 20:32:27 2020 +0800

    MIPS: Loongson64: Do not override watch and ejtag feature

    [ Upstream commit 433c1ca0d441ee0b88fdd83c84ee6d6d43080dcd ]

    Do not override ejtag feature to 0 as Loongson 3A1000+ do have ejtag.
    For watch, as KVM emulated CPU doesn't have watch feature, we should
    not enable it unconditionally.

    Signed-off-by: Jiaxun Yang <jiaxun.yang@flygoat.com>
    Reviewed-by: Huacai Chen <chenhc@lemote.com>
    Signed-off-by: Thomas Bogendoerfer <tsbogend@alpha.franken.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 36120307971e4a25d4571a0cbd25e1ba7711eee7
Author: Hanjun Guo <guohanjun@huawei.com>
Date:   Wed Jul 22 17:54:21 2020 +0800

    dmaengine: acpi: Put the CSRT table after using it

    [ Upstream commit 7eb48dd094de5fe0e216b550e73aa85257903973 ]

    The acpi_get_table() should be coupled with acpi_put_table() if
    the mapped table is not used at runtime to release the table
    mapping, put the CSRT table buf after using it.

    Signed-off-by: Hanjun Guo <guohanjun@huawei.com>
```

```
    Link: https://lore.kernel.org/r/1595411661-15936-1-git-send-email-guohanjun@huawei.com
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a588a27d36f272917810b68403e54ccf449c1302
Author: Vineet Gupta <vgupta@synopsys.com>
Date:   Thu Jul 9 19:52:32 2020 -0700

    ARC: HSDK: wireup perf irq

    [ Upstream commit fe81d927b78c4f0557836661d32e41ebc957b024 ]

    Newer version of HSDK aka HSDK-4xD (with dual issue HS48x4 CPU) wired up
    the perf interrupt, so enable that in DT.
    This is OK for old HSDK where this irq is ignored because pct irq is not
    wired up in hardware.

    Signed-off-by: Vineet Gupta <vgupta@synopsys.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c0823ab23d47ad0ada835b00d1ca0193f8430881
Author: Vitaly Kuznetsov <vkuznets@redhat.com>
Date:   Fri Sep 11 11:31:47 2020 +0200

    KVM: x86: always allow writing '0' to MSR_KVM_ASYNC_PF_EN

    [ Upstream commit d831de177217cd494bfb99f2c849a0d40c2a7890 ]

    Even without in-kernel LAPIC we should allow writing '0' to
    MSR_KVM_ASYNC_PF_EN as we're not enabling the mechanism. In
    particular, QEMU with 'kernel-irqchip=off' fails to start
    a guest with

    qemu-system-x86_64: error: failed to set MSR 0x4b564d02 to 0x0

    Fixes: 9d3c447c72fb2 ("KVM: X86: Fix async pf caused null-ptr-deref")
    Reported-by: Dr. David Alan Gilbert <dgilbert@redhat.com>
    Signed-off-by: Vitaly Kuznetsov <vkuznets@redhat.com>
    Message-Id: <20200911093147.484565-1-vkuznets@redhat.com>
    [Actually commit the version proposed by Sean Christopherson. - Paolo]
    Signed-off-by: Paolo Bonzini <pbonzini@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 485d953c997e160404d83d7f4ff417257d047bab
Author: Chenyi Qiang <chenyi.qiang@intel.com>
Date:   Fri Aug 28 16:56:21 2020 +0800

    KVM: nVMX: Fix the update value of nested load IA32_PERF_GLOBAL_CTRL control

    [ Upstream commit c6b177a3beb9140dc0ba05b61c5142fcec5f2bf7 ]

    A minor fix for the update of VM_EXIT_LOAD_IA32_PERF_GLOBAL_CTRL field
    in exit_ctls_high.

    Fixes: 03a8871add95 ("KVM: nVMX: Expose load IA32_PERF_GLOBAL_CTRL
    VM-{Entry,Exit} control")
    Signed-off-by: Chenyi Qiang <chenyi.qiang@intel.com>
    Reviewed-by: Xiaoyao Li <xiaoyao.li@intel.com>
    Message-Id: <20200828085622.8365-5-chenyi.qiang@intel.com>
    Reviewed-by: Jim Mattson <jmattson@google.com>
    Signed-off-by: Paolo Bonzini <pbonzini@redhat.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2115ddfc96c05d0a3287c342a9bc13b6b062ff1a
Author: Florian Fainelli <f.fainelli@gmail.com>
```

```
Date:   Wed Aug 26 16:49:19 2020 -0700

    arm64: dts: ns2: Fixed QSPI compatible string

    [ Upstream commit 686e0a0c8c61e0e3f55321d0181fece3efd92777 ]

    The string was incorrectly defined before from least to most specific,
    swap the compatible strings accordingly.

    Fixes: ff73917d38a6 ("ARM64: dts: Add QSPI Device Tree node for NS2")
    Signed-off-by: Florian Fainelli <f.fainelli@gmail.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit be8d961d3d4e76a3daa01f16a346eba943163e20
Author: Florian Fainelli <f.fainelli@gmail.com>
Date:   Wed Aug 26 16:45:29 2020 -0700

    ARM: dts: BCM5301X: Fixed QSPI compatible string

    [ Upstream commit b793dab8d811e103665d6bddaaea1c25db3776eb ]

    The string was incorrectly defined before from least to most
    specific, swap the compatible strings accordingly.

    Fixes: 1c8f40650723 ("ARM: dts: BCM5301X: convert to iProc QSPI")
    Signed-off-by: Florian Fainelli <f.fainelli@gmail.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9fd9bb1bd6cc886ffbd711861d7a4f3aea1d64ce
Author: Florian Fainelli <f.fainelli@gmail.com>
Date:   Wed Aug 26 16:44:25 2020 -0700

    ARM: dts: NSP: Fixed QSPI compatible string

    [ Upstream commit d1ecc40a954fd0f5e3789b91fa80f15e82284e39 ]

    The string was incorrectly defined before from least to most
    specific, swap the compatible strings accordingly.

    Fixes: 329f98c1974e ("ARM: dts: NSP: Add QSPI nodes to NSPI and bcm958625k DTSes")
    Signed-off-by: Florian Fainelli <f.fainelli@gmail.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit dc4145dfdc6b980f908635ffd8eaafbe4e23e289
Author: Florian Fainelli <f.fainelli@gmail.com>
Date:   Wed Aug 26 16:43:41 2020 -0700

    ARM: dts: bcm: HR2: Fixed QSPI compatible string

    [ Upstream commit d663186293a818af97c648624bee6c7a59e8218b ]

    The string was incorrectly defined before from least to most specific,
    swap the compatible strings accordingly.

    Fixes: b9099ec754b5 ("ARM: dts: Add Broadcom Hurricane 2 DTS include file")
    Signed-off-by: Florian Fainelli <f.fainelli@gmail.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f360dbeb301f85276c957c942978d01ba4aa3af7
Author: Sagi Grimberg <sagi@grimberg.me>
Date:   Fri Sep 4 12:50:39 2020 -0700

    IB/isert: Fix unaligned immediate-data handling

    [ Upstream commit 0b089c1ef7047652b13b4cdfdb1e0e7dbdb8c9ab ]
```

```
    Currently we allocate rx buffers in a single contiguous buffers for
    headers (iser and iscsi) and data trailer. This means that most likely the
    data starting offset is aligned to 76 bytes (size of both headers).

    This worked fine for years, but at some point this broke, resulting in
    data corruptions in isert when a command comes with immediate data and the
    underlying backend device assumes 512 bytes buffer alignment.

    We assume a hard-requirement for all direct I/O buffers to be 512 bytes
    aligned. To fix this, we should avoid passing unaligned buffers for I/O.

    Instead, we allocate our recv buffers with some extra space such that we
    can have the data portion align to 512 byte boundary. This also means that
    we cannot reference headers or data using structure but rather
    accessors (as they may move based on alignment). Also, get rid of the
    wrong __packed annotation from iser_rx_desc as this has only harmful
    effects (not aligned to anything).

    This affects the rx descriptors for iscsi login and data plane.

    Fixes: 3d75ca0adef4 ("block: introduce multi-page bvec helpers")
    Link: https://lore.kernel.org/r/20200904195039.31687-1-sagi@grimberg.me
    Reported-by: Stephen Rust <srust@blockbridge.com>
    Tested-by: Doug Dumitru <doug@dumitru.com>
    Signed-off-by: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9f6550695d2587400d6a657a97ae24ecf491ba9e
Author: Md Haris Iqbal <haris.iqbal@cloud.ionos.com>
Date:   Mon Sep 7 15:52:16 2020 +0530

    RDMA/rtrs-srv: Set .release function for rtrs srv device during device init

    [ Upstream commit 39c2d639ca183a400ba3259fa0825714cbb09c53 ]

    The device .release function was not being set during the device
    initialization. This was leading to the below warning, in error cases when
    put_srv was called before device_add was called.

    Warning:

    Device '(null)' does not have a release() function, it is broken and must
    be fixed. See Documentation/kobject.txt.

    So, set the device .release function during device initialization in the
    __alloc_srv() function.

    Fixes: baa5b28b7a47 ("RDMA/rtrs-srv: Replace device_register with device_initialize and
device_add")
    Link: https://lore.kernel.org/r/20200907102216.104041-1-haris.iqbal@cloud.ionos.com
    Signed-off-by: Md Haris Iqbal <haris.iqbal@cloud.ionos.com>
    Reviewed-by: Leon Romanovsky <leonro@nvidia.com>
    Acked-by: Jack Wang <jinpu.wang@cloud.ionos.com>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 5bcd37b6bf21399cfed9b68c88b067928bbc404a
Author: Ritesh Harjani <riteshh@linux.ibm.com>
Date:   Wed Sep 9 08:44:25 2020 +0530

    block: Set same_page to false in __bio_try_merge_page if ret is false

    [ Upstream commit 2cd896a5e86fc326bda8614b96c0401dcc145868 ]
```

```
    If we hit the UINT_MAX limit of bio->bi_iter.bi_size and so we are anyway
    not merging this page in this bio, then it make sense to make same_page
    also as false before returning.

    Without this patch, we hit below WARNING in iomap.
    This mostly happens with very large memory system and / or after tweaking
    vm dirty threshold params to delay writeback of dirty data.

    WARNING: CPU: 18 PID: 5130 at fs/iomap/buffered-io.c:74 iomap_page_release+0x120/0x150
     CPU: 18 PID: 5130 Comm: fio Kdump: loaded Tainted: G        W         5.8.0-rc3 #6
     Call Trace:
      __remove_mapping+0x154/0x320 (unreliable)
      iomap_releasepage+0x80/0x180
      try_to_release_page+0x94/0xe0
      invalidate_inode_page+0xc8/0x110
      invalidate_mapping_pages+0x1dc/0x540
      generic_fadvise+0x3c8/0x450
      xfs_file_fadvise+0x2c/0xe0 [xfs]
      vfs_fadvise+0x3c/0x60
      ksys_fadvise64_64+0x68/0xe0
      sys_fadvise64+0x28/0x40
      system_call_exception+0xf8/0x1c0
      system_call_common+0xf0/0x278

    Fixes: cc90bc68422 ("block: fix "check bi_size overflow before merge"")
    Reported-by: Shivaprasad G Bhat <sbhat@linux.ibm.com>
    Suggested-by: Christoph Hellwig <hch@infradead.org>
    Signed-off-by: Anju T Sudhakar <anju@linux.vnet.ibm.com>
    Signed-off-by: Ritesh Harjani <riteshh@linux.ibm.com>
    Reviewed-by: Ming Lei <ming.lei@redhat.com>
    Reviewed-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit bc1d374adc82a03fb2113734ba9c16f638b887b2
Author: Dan Carpenter <dan.carpenter@oracle.com>
Date:   Wed Sep 9 12:43:04 2020 +0300

    spi: stm32: fix pm_runtime_get_sync() error checking

    [ Upstream commit c170a5a3b6944ad8e76547c4a1d9fe81c8f23ac8 ]

    The pm_runtime_get_sync() can return either 0 or 1 on success but this
    code treats 1 as a failure.

    Fixes: db96bf976a4f ("spi: stm32: fixes suspend/resume management")
    Signed-off-by: Dan Carpenter <dan.carpenter@oracle.com>
    Reviewed-by: Alain Volmat <alain.volmat@st.com>
    Link: https://lore.kernel.org/r/20200909094304.GA420136@mwanda
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e3364a81254ba8f30102002d598ecc1ee04c5b74
Author: Sagi Grimberg <sagi@grimberg.me>
Date:   Tue Sep 8 12:56:08 2020 -0700

    nvme-fabrics: allow to queue requests for live queues

    [ Upstream commit 73a5379937ec89b91e907bb315e2434ee9696a2c ]

    Right now we are failing requests based on the controller state (which
    is checked inline in nvmf_check_ready) however we should definitely
    accept requests if the queue is live.
```

```
    When entering controller reset, we transition the controller into
    NVME_CTRL_RESETTING, and then return BLK_STS_RESOURCE for non-mpath
    requests (have blk_noretry_request set).

    This is also the case for NVME_REQ_USER for the wrong reason. There
    shouldn't be any reason for us to reject this I/O in a controller reset.
    We do want to prevent passthru commands on the admin queue because we
    need the controller to fully initialize first before we let user passthru
    admin commands to be issued.

    In a non-mpath setup, this means that the requests will simply be
    requeued over and over forever not allowing the q_usage_counter to drop
    its final reference, causing controller reset to hang if running
    concurrently with heavy I/O.

    Fixes: 35897b920c8a ("nvme-fabrics: fix and refine state checks in __nvmf_check_ready")
    Reviewed-by: James Smart <james.smart@broadcom.com>
    Signed-off-by: Sagi Grimberg <sagi@grimberg.me>
    Signed-off-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e8555b6d1dd8347e513221265ca2acf6ecca8278
Author: Tycho Andersen <tycho@tycho.pizza>
Date:   Tue Sep 1 19:40:16 2020 -0600

    seccomp: don't leak memory when filter install races

    [ Upstream commit a566a9012acd7c9a4be7e30dc7acb7a811ec2260 ]

    In seccomp_set_mode_filter() with TSYNC | NEW_LISTENER, we first initialize
    the listener fd, then check to see if we can actually use it later in
    seccomp_may_assign_mode(), which can fail if anyone else in our thread
    group has installed a filter and caused some divergence. If we can't, we
    partially clean up the newly allocated file: we put the fd, put the file,
    but don't actually clean up the *memory* that was allocated at
    filter->notif. Let's clean that up too.

    To accomplish this, let's hoist the actual "detach a notifier from a
    filter" code to its own helper out of seccomp_notify_release(), so that in
    case anyone adds stuff to init_listener(), they only have to add the
    cleanup code in one spot. This does a bit of extra locking and such on the
    failure path when the filter is not attached, but it's a slow failure path
    anyway.

    Fixes: 51891498f2da ("seccomp: allow TSYNC and USER_NOTIF together")
    Reported-by: syzbot+3ad9614a12f80994c32e@syzkaller.appspotmail.com
    Signed-off-by: Tycho Andersen <tycho@tycho.pizza>
    Acked-by: Christian Brauner <christian.brauner@ubuntu.com>
    Link: https://lore.kernel.org/r/20200902014017.934315-1-tycho@tycho.pizza
    Signed-off-by: Kees Cook <keescook@chromium.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b730cc810f71f7b2126d390b63b981e744777c35
Author: Christoph Hellwig <hch@lst.de>
Date:   Tue Sep 8 16:15:06 2020 +0200

    block: restore a specific error code in bdev_del_partition

    [ Upstream commit 88ce2a530cc9865a894454b2e40eba5957a60e1a ]

    mdadm relies on the fact that deleting an invalid partition returns
    -ENXIO or -ENOTTY to detect if a block device is a partition or a
    whole device.

    Fixes: 08fc1ab6d748 ("block: fix locking in bdev_del_partition")
```

```
    Reported-by: kernel test robot <rong.a.chen@intel.com>
    Signed-off-by: Christoph Hellwig <hch@lst.de>
    Signed-off-by: Jens Axboe <axboe@kernel.dk>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 7ded6f75c325101cc230c72c650ec7bb734d490c
Author: Tali Perry <tali.perry1@gmail.com>
Date:   Mon Aug 31 00:31:21 2020 +0300

    i2c: npcm7xx: Fix timeout calculation

    [ Upstream commit 06be67266a0c9a6a1ffb330a4ab50c2f21612e2b ]

    timeout_usec value calculation was wrong, the calculated value
    was in msec instead of usec.

    Fixes: 56a1485b102e ("i2c: npcm7xx: Add Nuvoton NPCM I2C controller driver")
    Signed-off-by: Tali Perry <tali.perry1@gmail.com>
    Reviewed-by: Avi Fishman <avifishman70@gmail.com>
    Reviewed-by: Joel Stanley <joel@jms.id.au>
    Reviewed-by: Alex Qiu <xqiu@google.com>
    Signed-off-by: Wolfram Sang <wsa@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 70d1f2b7ec595622865a02af7b9d81762db0bdb0
Author: Filipe Manana <fdmanana@suse.com>
Date:   Fri Sep 4 17:22:57 2020 +0100

    btrfs: fix NULL pointer dereference after failure to create snapshot

    [ Upstream commit 2d892ccdc163a3d2e08c5ed1cea8b61bf7e4f531 ]

    When trying to get a new fs root for a snapshot during the transaction
    at transaction.c:create_pending_snapshot(), if btrfs_get_new_fs_root()
    fails we leave "pending->snap" pointing to an error pointer, and then
    later at ioctl.c:create_snapshot() we dereference that pointer, resulting
    in a crash:

      [12264.614689] BUG: kernel NULL pointer dereference, address: 00000000000007c4
      [12264.615650] #PF: supervisor write access in kernel mode
      [12264.616487] #PF: error_code(0x0002) - not-present page
      [12264.617436] PGD 0 P4D 0
      [12264.618328] Oops: 0002 [#1] PREEMPT SMP DEBUG_PAGEALLOC PTI
      [12264.619150] CPU: 0 PID: 2310635 Comm: fsstress Tainted: G        W         5.9.0-
rc3-btrfs-next-67 #1
      [12264.619960] Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS rel-1.13.0-
0-gf21b5a4aeb02-prebuilt.qemu.org 04/01/2014
      [12264.621769] RIP: 0010:btrfs_mksubvol+0x438/0x4a0 [btrfs]
      [12264.622528] Code: bc ef ff ff (...)
      [12264.624092] RSP: 0018:ffffaa6fc7277cd8 EFLAGS: 00010282
      [12264.624669] RAX: 00000000fffffff4 RBX: ffff9d3e8f151a60 RCX: 0000000000000000
      [12264.625249] RDX: 0000000000000001 RSI: ffffffff9d56c9be RDI: fffffffffffffff4
      [12264.625830] RBP: ffff9d3e8f151b48 R08: 0000000000000000 R09: 0000000000000000
      [12264.626413] R10: 0000000000000000 R11: 0000000000000000 R12: 00000000fffffff4
      [12264.626994] R13: ffff9d3ede380538 R14: ffff9d3ede380500 R15: ffff9d3f61b2eeb8
      [12264.627582] FS:  00007f140d5d8200(0000) GS:ffff9d3fb5e00000(0000)
knlGS:0000000000000000
      [12264.628176] CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
      [12264.628773] CR2: 00000000000007c4 CR3: 000000020f8e8004 CR4: 0000000003706f0
      [12264.629379] DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000
      [12264.629994] DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400
      [12264.630594] Call Trace:
      [12264.631227]  btrfs_mksnapshot+0x7b/0xb0 [btrfs]
      [12264.631840]  __btrfs_ioctl_snap_create+0x16f/0x1a0 [btrfs]
      [12264.632458]  btrfs_ioctl_snap_create_v2+0xb0/0xf0 [btrfs]
```

```
        [12264.633078]  btrfs_ioctl+0x1864/0x3130 [btrfs]
        [12264.633689]  ? do_sys_openat2+0x1a7/0x2d0
        [12264.634295]  ? kmem_cache_free+0x147/0x3a0
        [12264.634899]  ? __x64_sys_ioctl+0x83/0xb0
        [12264.635488]  __x64_sys_ioctl+0x83/0xb0
        [12264.636058]  do_syscall_64+0x33/0x80
        [12264.636616]  entry_SYSCALL_64_after_hwframe+0x44/0xa9

        (gdb) list *(btrfs_mksubvol+0x438)
        0x7c7b8 is in btrfs_mksubvol (fs/btrfs/ioctl.c:858).
        853             ret = 0;
        854             pending_snapshot->anon_dev = 0;
        855     fail:
        856             /* Prevent double freeing of anon_dev */
        857             if (ret && pending_snapshot->snap)
        858                     pending_snapshot->snap->anon_dev = 0;
        859             btrfs_put_root(pending_snapshot->snap);
        860             btrfs_subvolume_release_metadata(root, &pending_snapshot->block_rsv);
        861     free_pending:
        862             if (pending_snapshot->anon_dev)

    So fix this by setting "pending->snap" to NULL if we get an error from the
    call to btrfs_get_new_fs_root() at transaction.c:create_pending_snapshot().

    Fixes: 2dfb1e43f57dd3 ("btrfs: preallocate anon block device at first phase of snapshot
creation")
    Signed-off-by: Filipe Manana <fdmanana@suse.com>
    Reviewed-by: David Sterba <dsterba@suse.com>
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit b6894531a925660b6dc16639c38fb7ec8bc97a72
Author: Marek Vasut <marex@denx.de>
Date:   Sat Sep 5 17:19:13 2020 +0200

    spi: stm32: Rate-limit the 'Communication suspended' message

    [ Upstream commit ea8be08cc9358f811e4175ba7fa7fea23c5d393e ]

    The 'spi_stm32 44004000.spi: Communication suspended' message means that
    when using PIO, the kernel did not read the FIFO fast enough and so the
    SPI controller paused the transfer. Currently, this is printed on every
    single such event, so if the kernel is busy and the controller is pausing
    the transfers often, the kernel will be all the more busy scrolling this
    message into the log buffer every few milliseconds. That is not helpful.

    Instead, rate-limit the message and print it every once in a while. It is
    not possible to use the default dev_warn_ratelimited(), because that is
    still too verbose, as it prints 10 lines (DEFAULT_RATELIMIT_BURST) every
    5 seconds (DEFAULT_RATELIMIT_INTERVAL). The policy here is to print 1 line
    every 50 seconds (DEFAULT_RATELIMIT_INTERVAL * 10), because 1 line is more
    than enough and the cycles saved on printing are better left to the CPU to
    handle the SPI. However, dev_warn_once() is also not useful, as the user
    should be aware that this condition is possibly recurring or ongoing. Thus
    the custom rate-limit policy.

    Finally, turn the message from dev_warn() to dev_dbg(), since the system
    does not suffer any sort of malfunction if this message appears, it is
    just slowing down. This further reduces the printing into the log buffer
    and frees the CPU to do useful work.

    Fixes: dcbe0d84dfa5 ("spi: add driver for STM32 SPI controller")
    Signed-off-by: Marek Vasut <marex@denx.de>
    Cc: Alexandre Torgue <alexandre.torgue@st.com>
    Cc: Amelie Delaunay <amelie.delaunay@st.com>
```

```
    Cc: Antonio Borneo <borneo.antonio@gmail.com>
    Cc: Mark Brown <broonie@kernel.org>
    Link: https://lore.kernel.org/r/20200905151913.117775-1-marex@denx.de
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 0a409e1a47ffa142166cfc73581352cf7b51c509
Author: Douglas Anderson <dianders@chromium.org>
Date:   Thu Aug 27 07:58:41 2020 -0700

    mmc: sdhci-msm: Add retries when all tuning phases are found valid

    [ Upstream commit 9d5dcefb7b114d610aeb2371f6a6f119af316e43 ]

    As the comments in this patch say, if we tune and find all phases are
    valid it's _almost_ as bad as no phases being found valid.  Probably
    all phases are not really reliable but we didn't detect where the
    unreliable place is.  That means we'll essentially be guessing and
    hoping we get a good phase.

    This is not just a problem in theory.  It was causing real problems on
    a real board.  On that board, most often phase 10 is found as the only
    invalid phase, though sometimes 10 and 11 are invalid and sometimes
    just 11.  Some percentage of the time, however, all phases are found
    to be valid.  When this happens, the current logic will decide to use
    phase 11.  Since phase 11 is sometimes found to be invalid, this is a
    bad choice.  Sure enough, when phase 11 is picked we often get mmc
    errors later in boot.

    I have seen cases where all phases were found to be valid 3 times in a
    row, so increase the retry count to 10 just to be extra sure.

    Fixes: 415b5a75da43 ("mmc: sdhci-msm: Add platform_execute_tuning implementation")
    Signed-off-by: Douglas Anderson <dianders@chromium.org>
    Reviewed-by: Veerabhadrarao Badiganti <vbadigan@codeaurora.org>
    Acked-by: Adrian Hunter <adrian.hunter@intel.com>
    Link:
https://lore.kernel.org/r/20200827075809.1.If179abf5ecb67c963494db79c3bc4247d987419b@changei
d
    Signed-off-by: Ulf Hansson <ulf.hansson@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 02a5514209ae8a7920fb5bd0724c596a648d9703
Author: Raul E Rangel <rrangel@chromium.org>
Date:   Mon Aug 31 15:10:32 2020 -0600

    mmc: sdhci-acpi: Clear amd_sdhci_host on reset

    [ Upstream commit 2cf9bfe9be75ed3656bbf882fb70c3e3047866e4 ]

    The commit 61d7437ed1390 ("mmc: sdhci-acpi: Fix HS400 tuning for AMDI0040")
    broke resume for eMMC HS400. When the system suspends the eMMC controller
    is powered down. So, on resume we need to reinitialize the controller.
    Although, amd_sdhci_host was not getting cleared, so the DLL was never
    re-enabled on resume. This results in HS400 being non-functional.

    To fix the problem, this change clears the tuned_clock flag, clears the
    dll_enabled flag and disables the DLL on reset.

    Fixes: 61d7437ed1390 ("mmc: sdhci-acpi: Fix HS400 tuning for AMDI0040")
    Signed-off-by: Raul E Rangel <rrangel@chromium.org>
    Acked-by: Adrian Hunter <adrian.hunter@intel.com>
    Link:
https://lore.kernel.org/r/20200831150517.1.I93c78bfc6575771bb653c9d3fca5eb018a08417d@changei
d
```

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

commit e341aeac14f1957307d2f39b4dde35e41111e09e
Author: Fugang Duan <fugang.duan@nxp.com>
Date:   Thu Sep 3 18:05:21 2020 +0800

    ARM: dts: imx6sx: fix the pad QSPI1B_SCLK mux mode for uart3

    [ Upstream commit 3ee99f6a2379eca87ab11122b7e9abd68f3441e2 ]

    The pad QSPI1B_SCLK mux mode 0x1 is for function UART3_DTE_TX,
    correct the mux mode.

    Fixes: 743636f25f1d ("ARM: dts: imx: add pin function header for imx6sx")
    Signed-off-by: Fugang Duan <fugang.duan@nxp.com>
    Signed-off-by: Shawn Guo <shawnguo@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c01fd1453981e0b297a5c03de6f14d3aeff26ca7
Author: Alexandru Elisei <alexandru.elisei@arm.com>
Date:   Tue Sep 1 14:33:56 2020 +0100

    KVM: arm64: Update page shift if stage 2 block mapping not supported

    [ Upstream commit 7b75cd5128421c673153efb1236705696a1a9812 ]

    Commit 196f878a7ac2e (" KVM: arm/arm64: Signal SIGBUS when stage2 discovers
    hwpoison memory") modifies user_mem_abort() to send a SIGBUS signal when
    the fault IPA maps to a hwpoisoned page. Commit 1559b7583ff6 ("KVM:
    arm/arm64: Re-check VMA on detecting a poisoned page") changed
    kvm_send_hwpoison_signal() to use the page shift instead of the VMA because
    at that point the code had already released the mmap lock, which means
    userspace could have modified the VMA.

    If userspace uses hugetlbfs for the VM memory, user_mem_abort() tries to
    map the guest fault IPA using block mappings in stage 2. That is not always
    possible, if, for example, userspace uses dirty page logging for the VM.
    Update the page shift appropriately in those cases when we downgrade the
    stage 2 entry from a block mapping to a page.

    Fixes: 1559b7583ff6 ("KVM: arm/arm64: Re-check VMA on detecting a poisoned page")
    Signed-off-by: Alexandru Elisei <alexandru.elisei@arm.com>
    Signed-off-by: Marc Zyngier <maz@kernel.org>
    Reviewed-by: Gavin Shan <gshan@redhat.com>
    Link: https://lore.kernel.org/r/20200901133357.52640-2-alexandru.elisei@arm.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 95de64bc498ebc7ec4fc6a713831eaa30896777b
Author: Maxime Ripard <maxime@cerno.tech>
Date:   Tue Jul 28 15:48:10 2020 +0200

    drm/sun4i: backend: Disable alpha on the lowest plane on the A20

    [ Upstream commit 5e2e2600a3744491a8b49b92597c13b693692082 ]

    Unlike we previously thought, the per-pixel alpha is just as broken on the
    A20 as it is on the A10. Remove the quirk that says we can use it.

    Fixes: dcf496a6a608 ("drm/sun4i: sun4i: Introduce a quirk for lowest plane alpha
support")
    Signed-off-by: Maxime Ripard <maxime@cerno.tech>
    Reviewed-by: Chen-Yu Tsai <wens@csie.org>
    Cc: Paul Kocialkowski <paul.kocialkowski@bootlin.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20200728134810.883457-2-
```

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

commit 272e10fbf4d021e69ff90b1995fae7065d4cc404
Author: Maxime Ripard <maxime@cerno.tech>
Date:   Tue Jul 28 15:48:09 2020 +0200

    drm/sun4i: backend: Support alpha property on lowest plane

    [ Upstream commit e359c70462d2a82aae80274d027351d38792dde6 ]

    Unlike what we previously thought, only the per-pixel alpha is broken on
    the lowest plane and the per-plane alpha isn't. Remove the check on the
    alpha property being set on the lowest plane to reject a mode.

    Fixes: dcf496a6a608 ("drm/sun4i: sun4i: Introduce a quirk for lowest plane alpha
support")
    Signed-off-by: Maxime Ripard <maxime@cerno.tech>
    Reviewed-by: Chen-Yu Tsai <wens@csie.org>
    Cc: Paul Kocialkowski <paul.kocialkowski@bootlin.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20200728134810.883457-1-
maxime@cerno.tech
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit e856612eed7eafa5aab579c9ccd0cf203c0781c3
Author: Jernej Skrabec <jernej.skrabec@siol.net>
Date:   Wed Sep 2 00:03:05 2020 +0200

    drm/sun4i: Fix DE2 YVU handling

    [ Upstream commit 0ee9f600e69d901d31469359287b90bbe8e54553 ]

    Function sun8i_vi_layer_get_csc_mode() is supposed to return CSC mode
    but due to inproper return type (bool instead of u32) it returns just 0
    or 1. Colors are wrong for YVU formats because of that.

    Fixes: daab3d0e8e2b ("drm/sun4i: de2: csc_mode in de2 format struct is mostly
redundant")
    Reported-by: Roman Stratiienko <r.stratiienko@gmail.com>
    Signed-off-by: Jernej Skrabec <jernej.skrabec@siol.net>
    Tested-by: Roman Stratiienko <r.stratiienko@gmail.com>
    Signed-off-by: Maxime Ripard <maxime@cerno.tech>
    Link: https://patchwork.freedesktop.org/patch/msgid/20200901220305.6809-1-
jernej.skrabec@siol.net
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 412bbf8b9a22c8635773a12f08c897b7f91ab6fc
Author: Daniel Jordan <daniel.m.jordan@oracle.com>
Date:   Wed Sep 2 13:07:56 2020 -0400

    padata: fix possible padata_works_lock deadlock

    [ Upstream commit 1b0df11fde0f14a269a181b3b7f5122415bc5ed7 ]

    syzbot reports,

      WARNING: inconsistent lock state
      5.9.0-rc2-syzkaller #0 Not tainted
      --------------------------------
      inconsistent {IN-SOFTIRQ-W} -> {SOFTIRQ-ON-W} usage.
      syz-executor.0/26715 takes:
      (padata_works_lock){+.?.}-{2:2}, at: padata_do_parallel kernel/padata.c:220
      {IN-SOFTIRQ-W} state was registered at:
        spin_lock include/linux/spinlock.h:354 [inline]
        padata_do_parallel kernel/padata.c:220
```

```
          ...
          __do_softirq kernel/softirq.c:298
          ...
          sysvec_apic_timer_interrupt arch/x86/kernel/apic/apic.c:1091
          asm_sysvec_apic_timer_interrupt arch/x86/include/asm/idtentry.h:581

         Possible unsafe locking scenario:

               CPU0
               ----
          lock(padata_works_lock);
          <Interrupt>
            lock(padata_works_lock);
```

padata_do_parallel() takes padata_works_lock with softirqs enabled, so a
deadlock is possible if, on the same CPU, the lock is acquired in
process context and then softirq handling done in an interrupt leads to
the same path.

Fix by leaving softirqs disabled while do_parallel holds
padata_works_lock.

Reported-by: syzbot+f4b9f49e38e25eb4ef52@syzkaller.appspotmail.com
Fixes: 4611ce2246889 ("padata: allocate work structures for parallel jobs from a pool")
Signed-off-by: Daniel Jordan <daniel.m.jordan@oracle.com>
Cc: Herbert Xu <herbert@gondor.apana.org.au>
Cc: Steffen Klassert <steffen.klassert@secunet.com>
Cc: linux-crypto@vger.kernel.org
Cc: linux-kernel@vger.kernel.org
Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 03ad17724b7a4cc20664902bac1331803d159533
Author: Mike Tipton <mdtipton@codeaurora.org>
Date: Thu Sep 3 12:21:44 2020 -0700

    interconnect: qcom: Fix small BW votes being truncated to zero

    [Upstream commit 91e045b93db79a2ef66e045ad0d1f8f9d348e1f4]

    Small BW votes that translate to less than a single BCM unit are
    currently truncated to zero. Ensure that non-zero BW requests always
    result in at least a vote of 1 to BCM.

    Fixes: 976daac4a1c5 ("interconnect: qcom: Consolidate interconnect RPMh support")
    Signed-off-by: Mike Tipton <mdtipton@codeaurora.org>
    Link: https://lore.kernel.org/r/20200903192149.30385-2-mdtipton@codeaurora.org
    Signed-off-by: Georgi Djakov <georgi.djakov@linaro.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit fdbbeae2b076004b15aef80031864fa74b3191eb
Author: Josh Poimboeuf <jpoimboe@redhat.com>
Date: Fri Jul 17 13:29:48 2020 -0500

    Revert "kbuild: use -flive-patching when CONFIG_LIVEPATCH is enabled"

    [Upstream commit 318af7b80b6a6751520cf2b71edb8c45abb9d9d8]

    Use of the new -flive-patching flag was introduced with the following
    commit:

      43bd3a95c98e ("kbuild: use -flive-patching when CONFIG_LIVEPATCH is enabled")

    This flag has several drawbacks:

```
      - It disables some optimizations, so it can have a negative effect on
        performance.

      - According to the GCC documentation it's not compatible with LTO, which
        will become a compatibility issue as LTO support gets upstreamed in
        the kernel.

      - It was intended to be used for source-based patch generation tooling,
        as opposed to binary-based patch generation tooling (e.g.,
        kpatch-build).  It probably should have at least been behind a
        separate config option so as not to negatively affect other livepatch
        users.

      - Clang doesn't have the flag, so as far as I can tell, this method of
        generating patches is incompatible with Clang, which like LTO is
        becoming more mainstream.

      - It breaks GCC's implicit noreturn detection for local functions.  This
        is the cause of several "unreachable instruction" objtool warnings.

      - The broken noreturn detection is an obvious GCC regression, but we
        haven't yet gotten GCC developers to acknowledge that, which doesn't
        inspire confidence in their willingness to keep the feature working as
        optimizations are added or changed going forward.

      - While there *is* a distro which relies on this flag for their distro
        livepatch module builds, there's not a publicly documented way to
        create safe livepatch modules with it.  Its use seems to be based on
        tribal knowledge.  It serves no benefit to those who don't know how to
        use it.

        (In fact, I believe the current livepatch documentation and samples
        are misleading and dangerous, and should be corrected.  Or at least
        amended with a disclaimer.  But I don't feel qualified to make such
        changes.)

      Also, we have an idea for using objtool to detect function changes,
      which could potentially obsolete the need for this flag anyway.

      At this point the flag has no benefits for upstream which would
      counteract the above drawbacks.  Revert it until it becomes more ready.

      This reverts commit 43bd3a95c98e1a86b8b55d97f745c224ecff02b9.

      Fixes: 43bd3a95c98e ("kbuild: use -flive-patching when CONFIG_LIVEPATCH is enabled")
      Reported-by: Randy Dunlap <rdunlap@infradead.org>
      Signed-off-by: Josh Poimboeuf <jpoimboe@redhat.com>
      Acked-by: Miroslav Benes <mbenes@suse.cz>
      Signed-off-by: Petr Mladek <pmladek@suse.com>
      Link:
  https://lore.kernel.org/r/696262e997359666afa053fe7d1a9fb2bb373964.1595010490.git.jpoimboe@r
  edhat.com
      Signed-off-by: Sasha Levin <sashal@kernel.org>

  commit 61591cd38cdfc5f2102f95b91a102fd95d364f82
  Author: Tom Rix <trix@redhat.com>
  Date:   Wed Sep 2 13:26:50 2020 -0700

      soundwire: fix double free of dangling pointer

      [ Upstream commit 3fbbf2148a406b3e350fe91e6fdd78eb42ecad24 ]

      clang static analysis flags this problem

      stream.c:844:9: warning: Use of memory after
```

```
       it is freed
             kfree(bus->defer_msg.msg->buf);
                   ^~~~~~~~~~~~~~~~~~~~~~
```

    This happens in an error handler cleaning up memory
    allocated for elements in a list.

```
             list_for_each_entry(m_rt, &stream->master_list, stream_node) {
                     bus = m_rt->bus;

                     kfree(bus->defer_msg.msg->buf);
                     kfree(bus->defer_msg.msg);
             }
```

    And is triggered when the call to sdw_bank_switch() fails.
    There are a two problems.

    First, when sdw_bank_switch() fails, though it frees memory it
    does not clear bus's reference 'defer_msg.msg' to that memory.

    The second problem is the freeing msg->buf. In some cases
    msg will be NULL so this will dereference a null pointer.
    Need to check before freeing.

    Fixes: 99b8a5d608a6 ("soundwire: Add bank switch routine")
    Signed-off-by: Tom Rix <trix@redhat.com>
    Reviewed-by: Nick Desaulniers <ndesaulniers@google.com>
    Reviewed-by: Pierre-Louis Bossart <pierre-louis.bossart@linux.intel.com>
    Link: https://lore.kernel.org/r/20200902202650.14189-1-trix@redhat.com
    Signed-off-by: Vinod Koul <vkoul@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6d1f49ae494dd133e13bbf74e8ca9b6508ed99ed
Author: Tomas Henzl <thenzl@redhat.com>
Date: Tue Sep 1 16:50:26 2020 +0200

    scsi: mpt3sas: Don't call disable_irq from IRQ poll handler

    [Upstream commit b614d55b970d08bcac5b0bc15a5526181b3e4459]

    disable_irq() might sleep, replace it with disable_irq_nosync(). For
    synchronisation 'irq_poll_scheduled' is sufficient

    Fixes: 320e77acb3 scsi: mpt3sas: Irq poll to avoid CPU hard lockups
    Link: https://lore.kernel.org/r/20200901145026.12174-1-thenzl@redhat.com
    Signed-off-by: Tomas Henzl <thenzl@redhat.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 40158298158da2e26c7b12d2d6ca01d3798ad32f
Author: Tomas Henzl <thenzl@redhat.com>
Date: Thu Aug 27 18:53:32 2020 +0200

    scsi: megaraid_sas: Don't call disable_irq from process IRQ poll

    [Upstream commit d2af39141eea34ef651961e885f49d96781a1016]

    disable_irq() might sleep. Replace it with disable_irq_nosync() which is
    sufficient as irq_poll_scheduled protects against concurrently running
    complete_cmd_fusion() from megasas_irqpoll() and megasas_isr_fusion().

    Link: https://lore.kernel.org/r/20200827165332.8432-1-thenzl@redhat.com
    Fixes: a6ffd5bf681 scsi: megaraid_sas: Call disable_irq from process IRQ poll
    Signed-off-by: Tomas Henzl <thenzl@redhat.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>

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

commit 1398cf27e62ff0bd9abdb8d694c132bad6339c1f
Author: Kamal Heib <kamalheib1@gmail.com>
Date:   Wed Sep 2 15:43:04 2020 +0300

    RDMA/core: Fix reported speed and width

    [ Upstream commit 28b0865714b315e318ac45c4fc9156f3d4649646 ]

    When the returned speed from __ethtool_get_link_ksettings() is
    SPEED_UNKNOWN this will lead to reporting a wrong speed and width for
    providers that uses ib_get_eth_speed(), fix that by defaulting the
    netdev_speed to SPEED_1000 in case the returned value from
    __ethtool_get_link_ksettings() is SPEED_UNKNOWN.

    Fixes: d41861942fc5 ("IB/core: Add generic function to extract IB speed from netdev")
    Link: https://lore.kernel.org/r/20200902124304.170912-1-kamalheib1@gmail.com
    Signed-off-by: Kamal Heib <kamalheib1@gmail.com>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c5452d1e9143d2d70dde5887c19d32c7e58e33a3
Author: Xi Wang <wangxi11@huawei.com>
Date:   Tue Sep 1 20:38:55 2020 +0800

    RDMA/core: Fix unsafe linked list traversal after failing to allocate CQ

    [ Upstream commit 8aa64be019567c4f90d45c5082a4b6f22e182d00 ]

    It's not safe to access the next CQ in list_for_each_entry() after
    invoking ib_free_cq(), because the CQ has already been freed in current
    iteration.  It should be replaced by list_for_each_entry_safe().

    Fixes: c7ff819aefea ("RDMA/core: Introduce shared CQ pool API")
    Link: https://lore.kernel.org/r/1598963935-32335-1-git-send-email-liweihang@huawei.com
    Signed-off-by: Xi Wang <wangxi11@huawei.com>
    Signed-off-by: Weihang Li <liweihang@huawei.com>
    Reviewed-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f420254168c08191110fe86135aaecbfd8ef8b91
Author: Gerd Hoffmann <kraxel@redhat.com>
Date:   Tue Aug 18 09:25:10 2020 +0200

    drm/virtio: fix unblank

    [ Upstream commit c6016c6e39c3ee8fd671532520be3cc13e439db2 ]

    When going through a disable/enable cycle without changing the
    framebuffer the optimization added by commit 3954ff10e06e ("drm/virtio:
    skip set_scanout if framebuffer didn't change") causes the screen stay
    blank.  Add a bool to force an update to fix that.

    v2: use drm_atomic_crtc_needs_modeset() (Daniel).

    Cc: 1882851@bugs.launchpad.net
    Fixes: 3954ff10e06e ("drm/virtio: skip set_scanout if framebuffer didn't change")
    Signed-off-by: Gerd Hoffmann <kraxel@redhat.com>
    Tested-by: Jiri Slaby <jirislaby@kernel.org>
    Tested-by: Diego Viola <diego.viola@gmail.com>
    Acked-by: Daniel Vetter <daniel.vetter@ffwll.ch>
    Link: http://patchwork.freedesktop.org/patch/msgid/20200818072511.6745-2-
kraxel@redhat.com
```

```
    (cherry picked from commit 1bc371cd0ec907bab870cacb6e898105f9c41dc8)
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8817b6fe0f221d29d18ff9d8a316c81ccdc8ab69
Author: Luo Jiaxing <luojiaxing@huawei.com>
Date:   Wed Aug 26 15:24:26 2020 +0800

    scsi: libsas: Set data_dir as DMA_NONE if libata marks qc as NODATA

    [ Upstream commit 53de092f47ff40e8d4d78d590d95819d391bf2e0 ]

    It was discovered that sdparm will fail when attempting to disable write
    cache on a SATA disk connected via libsas.

    In the ATA command set the write cache state is controlled through the SET
    FEATURES operation. This is roughly corresponds to MODE SELECT in SCSI and
    the latter command is what is used in the SCSI-ATA translation layer. A
    subtle difference is that a MODE SELECT carries data whereas SET FEATURES
    is defined as a non-data command in ATA.

    Set the DMA data direction to DMA_NONE if the requested ATA command is
    identified as non-data.

    [mkp: commit desc]

    Fixes: fa1c1e8f1ece ("[SCSI] Add SATA support to libsas")
    Link: https://lore.kernel.org/r/1598426666-54544-1-git-send-email-luojiaxing@huawei.com
    Reviewed-by: John Garry <john.garry@huawei.com>
    Reviewed-by: Jason Yan <yanaijie@huawei.com>
    Signed-off-by: Luo Jiaxing <luojiaxing@huawei.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 34a5bbcd814c1918bb46276c5a43fa7e0314dcbd
Author: René Rebe <rene@exactcode.com>
Date:   Thu Aug 27 22:27:29 2020 +0200

    scsi: qla2xxx: Fix regression on sparc64

    [ Upstream commit 2a87d485c4cb4d1b34be6c278a1c6ce3e15c8b8a ]

    Commit 98aee70d19a7 ("qla2xxx: Add endianizer to max_payload_size
    modifier.") in 2014 broke qla2xxx on sparc64, e.g. as in the Sun Blade 1000
    / 2000. Unbreak by partial revert to fix endianness in nvram firmware
    default initialization. Also mark the second frame_payload_size in nvram_t
    __le16 to avoid new sparse warnings.

    Link: https://lore.kernel.org/r/20200827.222729.1875148247374704975.rene@exactcode.com
    Fixes: 98aee70d19a7 ("qla2xxx: Add endianizer to max_payload_size modifier.")
    Reviewed-by: Himanshu Madhani <himanshu.madhani@oracle.com>
    Reviewed-by: Bart Van Assche <bvanassche@acm.org>
    Acked-by: Arun Easi <aeasi@marvell.com>
    Signed-off-by: René Rebe <rene@exactcode.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit cc8785cc8fd457b416aa5354d8262a14bc8c69e5
Author: Ondrej Jirman <megous@megous.com>
Date:   Fri Aug 28 14:50:32 2020 +0200

    drm/sun4i: Fix dsi dcs long write function

    [ Upstream commit fd90e3808fd2c207560270c39b86b71af2231aa1 ]

    It's writing too much data. regmap_bulk_write expects number of
```

```
    register sized chunks to write, not a byte sized length of the
    bounce buffer. Bounce buffer needs to be padded too, so that
    regmap_bulk_write will not read past the end of the buffer.

    Fixes: 133add5b5ad4 ("drm/sun4i: Add Allwinner A31 MIPI-DSI controller support")
    Signed-off-by: Ondrej Jirman <megous@megous.com>
    Signed-off-by: Maxime Ripard <maxime@cerno.tech>
    Reviewed-by: Jernej Skrabec <jernej.skrabec@siol.net>
    Link: https://patchwork.freedesktop.org/patch/msgid/20200828125032.937148-1-
megous@megous.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit a2e7a8bff0ff530a10fe715bd08e09d214e7b4bd
Author: Krzysztof Kozlowski <krzk@kernel.org>
Date:   Sat Aug 29 13:12:48 2020 +0200

    arm64: dts: imx8mq: Fix TMU interrupt property

    [ Upstream commit 1f2f98f2703e8134678fe20982886085631eda23 ]

    "interrupt" is not a valid property.  Using proper name fixes dtbs_check
    warning:

      arch/arm64/boot/dts/freescale/imx8mq-zii-ultra-zest.dt.yaml: tmu@30260000:
'interrupts' is a required property

    Fixes: e464fd2ba4d4 ("arm64: dts: imx8mq: enable the multi sensor TMU")
    Signed-off-by: Krzysztof Kozlowski <krzk@kernel.org>
    Signed-off-by: Shawn Guo <shawnguo@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9abd480dbf3bdac071b944ecceb6f6932e1f8495
Author: Yu Kuai <yukuai3@huawei.com>
Date:   Wed Aug 26 09:08:26 2020 +0800

    drm/sun4i: add missing put_device() call in sun8i_r40_tcon_tv_set_mux()

    [ Upstream commit 07b5b12d97dc9f47ff3dff46c4f944a15bd762e5 ]

    If sun8i_r40_tcon_tv_set_mux() succeed, sun8i_r40_tcon_tv_set_mux()
    doesn't have a corresponding put_device(). Thus add put_device()
    to fix the exception handling for this function implementation.

    Fixes: 0305189afb32 ("drm/sun4i: tcon: Add support for R40 TCON")
    Signed-off-by: Yu Kuai <yukuai3@huawei.com>
    Signed-off-by: Maxime Ripard <maxime@cerno.tech>
    Link: https://patchwork.freedesktop.org/patch/msgid/20200826010826.1785487-1-
yukuai3@huawei.com
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6716ed0d4e1139bb30e473395fc712af821fcac0
Author: Selvin Xavier <selvin.xavier@broadcom.com>
Date:   Mon Aug 24 11:14:31 2020 -0700

    RDMA/bnxt_re: Remove the qp from list only if the qp destroy succeeds

    [ Upstream commit 097a9d23b7250355b182c5fd47dd4c55b22b1c33 ]

    Driver crashes when destroy_qp is re-tried because of an error
    returned. This is because the qp entry was removed from the qp list during
    the first call.

    Remove qp from the list only if destroy_qp returns success.

    The driver will still trigger a WARN_ON due to the memory leaking, but at
```

```
    least it isn't corrupting memory too.

    Fixes: 8dae419f9ec7 ("RDMA/bnxt_re: Refactor queue pair creation code")
    Link: https://lore.kernel.org/r/1598292876-26529-2-git-send-email-
selvin.xavier@broadcom.com
    Signed-off-by: Selvin Xavier <selvin.xavier@broadcom.com>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit f51174da6e5137736e25e36946671ab885b39450
Author: Naresh Kumar PBS <nareshkumar.pbs@broadcom.com>
Date:   Mon Aug 24 11:14:36 2020 -0700

    RDMA/bnxt_re: Fix driver crash on unaligned PSN entry address

    [ Upstream commit 934d0ac9a64d21523e3ad03ea4098da7826bc788 ]

    When computing the first psn entry, driver checks for page alignment. If
    this address is not page aligned,it attempts to compute the offset in that
    page for later use by using ALIGN macro. ALIGN macro does not return
    offset bytes but the requested aligned address and hence cannot be used
    directly to store as offset.  Since driver was using the address itself
    instead of offset, it resulted in invalid address when filling the psn
    buffer.

    Fixed driver to use PAGE_MASK macro to calculate the offset.

    Fixes: fddcbbb02af4 ("RDMA/bnxt_re: Simplify obtaining queue entry from hw ring")
    Link: https://lore.kernel.org/r/1598292876-26529-7-git-send-email-
selvin.xavier@broadcom.com
    Signed-off-by: Naresh Kumar PBS <nareshkumar.pbs@broadcom.com>
    Signed-off-by: Selvin Xavier <selvin.xavier@broadcom.com>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit d71b3a81c0acf91e2932bd2b2eaaf3dd770f2cb7
Author: Naresh Kumar PBS <nareshkumar.pbs@broadcom.com>
Date:   Mon Aug 24 11:14:34 2020 -0700

    RDMA/bnxt_re: Static NQ depth allocation

    [ Upstream commit f86b31c6a28f06eed3f6d9dc958079853b0792f1 ]

    At first, driver allocates memory for NQ based on qplib_ctx->cq_count and
    qplib_ctx->srqc_count.  Later when creating ring, it uses a static value
    of 128K -1.

    Fixing this with a static value for now.

    Fixes: b08fe048a69d ("RDMA/bnxt_re: Refactor net ring allocation function")
    Link: https://lore.kernel.org/r/1598292876-26529-5-git-send-email-
selvin.xavier@broadcom.com
    Signed-off-by: Naresh Kumar PBS <nareshkumar.pbs@broadcom.com>
    Signed-off-by: Selvin Xavier <selvin.xavier@broadcom.com>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 1e608f9308aa30b2e42ddd06b5e9d8e13b97758d
Author: Selvin Xavier <selvin.xavier@broadcom.com>
Date:   Mon Aug 24 11:14:33 2020 -0700

    RDMA/bnxt_re: Fix the qp table indexing

    [ Upstream commit 84cf229f4001c1216afc3e4c7f05e1620a0dd4bc ]
```

```
    qp->id can be a value outside the max number of qp. Indexing the qp table
    with the id can cause out of bounds crash. So changing the qp table
    indexing by (qp->id % max_qp -1).

    Allocating one extra entry for QP1. Some adapters create one more than the
    max_qp requested to accommodate QP1.  If the qp->id is 1, store the
    inforamtion in the last entry of the qp table.

    Fixes: f218d67ef004 ("RDMA/bnxt_re: Allow posting when QPs are in error")
    Link: https://lore.kernel.org/r/1598292876-26529-4-git-send-email-
selvin.xavier@broadcom.com
    Signed-off-by: Selvin Xavier <selvin.xavier@broadcom.com>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 6bcab9b51bab1a36cdf7bb3e5c814cabe8325d8e
Author: Selvin Xavier <selvin.xavier@broadcom.com>
Date:   Mon Aug 24 11:14:32 2020 -0700

    RDMA/bnxt_re: Do not report transparent vlan from QP1

    [ Upstream commit 2d0e60ee322d512fa6bc62d23a6760b39a380847 ]

    QP1 Rx CQE reports transparent VLAN ID in the completion and this is used
    while reporting the completion for received MAD packet. Check if the vlan
    id is configured before reporting it in the work completion.

    Fixes: 84511455ac5b ("RDMA/bnxt_re: report vlan_id and sl in qp1 recv completion")
    Link: https://lore.kernel.org/r/1598292876-26529-3-git-send-email-
selvin.xavier@broadcom.com
    Signed-off-by: Selvin Xavier <selvin.xavier@broadcom.com>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 9a65a5142af4cbaad644146662299c448b988bc1
Author: Kamal Heib <kamalheib1@gmail.com>
Date:   Tue Aug 25 18:17:25 2020 +0300

    RDMA/rxe: Fix panic when calling kmem_cache_create()

    [ Upstream commit d862060a4b43479887ae8e2c0b74a58c4e27e5f3 ]

    To avoid the following kernel panic when calling kmem_cache_create() with
    a NULL pointer from pool_cache(), Block the rxe_param_set_add() from
    running if the rdma_rxe module is not initialized.

     BUG: unable to handle kernel NULL pointer dereference at 000000000000000b
     PGD 0 P4D 0
     Oops: 0000 [#1] SMP NOPTI
     CPU: 4 PID: 8512 Comm: modprobe Kdump: loaded Not tainted 4.18.0-231.el8.x86_64 #1
     Hardware name: HPE ProLiant DL385 Gen10/ProLiant DL385 Gen10, BIOS A40 10/02/2018
     RIP: 0010:kmem_cache_alloc+0xd1/0x1b0
     Code: 8b 57 18 45 8b 77 1c 48 8b 5c 24 30 0f 1f 44 00 00 5b 48 89 e8 5d 41 5c 41 5d 41
5e 41 5f c3 81 e3 00 00 10 00 75 0e 4d 89 fe <41> f6 47 0b 04 0f 84 6c ff ff ff 4c 89 ff e8
cc da 01 00 49 89 c6
     RSP: 0018:ffffa2b8c773f9d0 EFLAGS: 00010246
     RAX: 0000000000000000 RBX: 0000000000000000 RCX: 0000000000000005
     RDX: 0000000000000004 RSI: 00000000006080c0 RDI: 0000000000000000
     RBP: ffff8ea0a8634fd0 R08: ffffa2b8c773f988 R09: 00000000006000c0
     R10: 0000000000000000 R11: 0000000000000230 R12: 00000000006080c0
     R13: ffffffffc0a97fc8 R14: 0000000000000000 R15: 0000000000000000
     FS:  00007f9138ed9740(0000) GS:ffff8ea4ae800000(0000) knlGS:0000000000000000
     CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
     CR2: 000000000000000b CR3: 000000046d59a000 CR4: 00000000003406e0
     Call Trace:
```

```
       rxe_alloc+0xc8/0x160 [rdma_rxe]
       rxe_get_dma_mr+0x25/0xb0 [rdma_rxe]
       __ib_alloc_pd+0xcb/0x160 [ib_core]
       ib_mad_init_device+0x296/0x8b0 [ib_core]
       add_client_context+0x11a/0x160 [ib_core]
       enable_device_and_get+0xdc/0x1d0 [ib_core]
       ib_register_device+0x572/0x6b0 [ib_core]
       ? crypto_create_tfm+0x32/0xe0
       ? crypto_create_tfm+0x7a/0xe0
       ? crypto_alloc_tfm+0x58/0xf0
       rxe_register_device+0x19d/0x1c0 [rdma_rxe]
       rxe_net_add+0x3d/0x70 [rdma_rxe]
       ? dev_get_by_name_rcu+0x73/0x90
       rxe_param_set_add+0xaf/0xc0 [rdma_rxe]
       parse_args+0x179/0x370
       ? ref_module+0x1b0/0x1b0
       load_module+0x135e/0x17e0
       ? ref_module+0x1b0/0x1b0
       ? __do_sys_init_module+0x13b/0x180
       __do_sys_init_module+0x13b/0x180
       do_syscall_64+0x5b/0x1a0
       entry_SYSCALL_64_after_hwframe+0x65/0xca
     RIP: 0033:0x7f9137ed296e
```

    This can be triggered if a user tries to use the 'module option' which is
    not actually a real module option but some idiotic (and thankfully no
    obsolete) sysfs interface.

    Fixes: 8700e3e7c485 ("Soft RoCE driver")
    Link: https://lore.kernel.org/r/20200825151725.254046-1-kamalheib1@gmail.com
    Signed-off-by: Kamal Heib <kamalheib1@gmail.com>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2b993e2073a755a20caf2d6d8b1b426f343d4a82
Author: Kamal Heib <kamalheib1@gmail.com>
Date: Sun Jul 5 13:43:10 2020 +0300

    RDMA/rxe: Drop pointless checks in rxe_init_ports

    [Upstream commit 6112ef62826e91afbae5446d5d47b38e25f47e3f]

    Both pkey_tbl_len and gid_tbl_len are set in rxe_init_port_param() - so no
    need to check if they aren't set.

    Fixes: 8700e3e7c485 ("Soft RoCE driver")
    Link: https://lore.kernel.org/r/20200705104313.283034-2-kamalheib1@gmail.com
    Signed-off-by: Kamal Heib <kamalheib1@gmail.com>
    Reviewed-by: Leon Romanovsky <leonro@mellanox.com>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit bc7a51ca9cef842487878d932b5011a30d4419ad
Author: Dinghao Liu <dinghao.liu@zju.edu.cn>
Date: Wed Aug 19 15:56:32 2020 +0800

    RDMA/rxe: Fix memleak in rxe_mem_init_user

    [Upstream commit e3ddd6067ee62f6e76ebcf61ff08b2c729ae412b]

    When page_address() fails, umem should be freed just like when
    rxe_mem_alloc() fails.

    Fixes: 8700e3e7c485 ("Soft RoCE driver")
    Link: https://lore.kernel.org/r/20200819075632.22285-1-dinghao.liu@zju.edu.cn

```
    Signed-off-by: Dinghao Liu <dinghao.liu@zju.edu.cn>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 81aaf0645c8135b354c4df91bfcb96f5e26f7d01
Author: Md Haris Iqbal <haris.iqbal@cloud.ionos.com>
Date:   Tue Aug 11 14:57:22 2020 +0530

    RDMA/rtrs-srv: Replace device_register with device_initialize and device_add

    [ Upstream commit baa5b28b7a474f66a511ebf71a2ade510652a2f6 ]

    There are error cases when we will call free_srv before device kobject is
    initialized; in such cases calling put_device generates the following
    warning:

     kobject: '(null)' (000000009f5445ed): is not initialized, yet
     kobject_put() is being called.

    So call device_initialize() only once when the server is allocated. If we
    end up calling put_srv() and subsequently free_srv(), our call to
    put_device() would result in deletion of the obj. Call device_add() later
    when we actually have a connection. Correspondingly, call device_del()
    instead of device_unregister() when srv->dev_ref falls to 0.

    Fixes: 9cb837480424 ("RDMA/rtrs: server: main functionality")
    Link: https://lore.kernel.org/r/20200811092722.2450-1-haris.iqbal@cloud.ionos.com
    Suggested-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Md Haris Iqbal <haris.iqbal@cloud.ionos.com>
    Reviewed-by: Jack Wang <jinpu.wang@cloud.ionos.com>
    Signed-off-by: Jason Gunthorpe <jgg@nvidia.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8edf8c890ee27b0a4973afdebf4a89f6791573cf
Author: Chris Healy <cphealy@gmail.com>
Date:   Sat Aug 22 19:25:05 2020 -0700

    ARM: dts: imx7d-zii-rmu2: fix rgmii phy-mode for ksz9031 phy

    [ Upstream commit 5cbb80d5236b47b149da292b86d5fc99a680894b ]

    Since commit bcf3440c6dd7 ("net: phy: micrel: add phy-mode support for the
    KSZ9031 PHY") the networking is broken on the imx7d-zii-rmu2 board.

    The end result is that network receive behaviour is marginal with lots of
    RX CRC errors experienced and NFS frequently failing.

    Quoting the explanation from Andrew Lunn in commit 0672d22a19244
    ("ARM: dts: imx: Fix the AR803X phy-mode"):

    "The problem here is, all the DTs were broken since day 0. However,
    because the PHY driver was also broken, nobody noticed and it
    worked. Now that the PHY driver has been fixed, all the bugs in the
    DTs now become an issue"

    Fix it by switching to phy-mode = "rgmii-id".

    Fixes: bcf3440c6dd7 ("net: phy: micrel: add phy-mode support for the KSZ9031 PHY")
    Signed-off-by: Chris Healy <cphealy@gmail.com>
    Reviewed-by: Fabio Estevam <festevam@gmail.com>
    Signed-off-by: Shawn Guo <shawnguo@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 86047782e0f97e49b8051024a01274d8d39f16a5
Author: Rob Herring <robh@kernel.org>
```

```
Date:   Tue Aug 18 14:25:31 2020 -0600

    arm64: dts: imx: Add missing imx8mm-beacon-kit.dtb to build

    [ Upstream commit 56e79dfd036b538940227fb31371c1cd67b2467f ]

    The imx8mm-beacon-kit.dtb was never added to dtbs-y and wasn't getting
    built. Fix it.

    Fixes: 593816fa2f35 ("arm64: dts: imx: Add Beacon i.MX8m-Mini development kit")
    Cc: Shawn Guo <shawnguo@kernel.org>
    Cc: Sascha Hauer <s.hauer@pengutronix.de>
    Cc: Pengutronix Kernel Team <kernel@pengutronix.de>
    Cc: Fabio Estevam <festevam@gmail.com>
    Cc: NXP Linux Team <linux-imx@nxp.com>
    Signed-off-by: Rob Herring <robh@kernel.org>
    Reviewed-by: Fabio Estevam <festevam@gmail.com>
    Signed-off-by: Shawn Guo <shawnguo@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c29e3108c40cfdcd460f041881a7df034f9e33ed
Author: Anson Huang <Anson.Huang@nxp.com>
Date:   Mon Aug 3 16:01:24 2020 +0800

    ARM: dts: imx7ulp: Correct gpio ranges

    [ Upstream commit deb6323b739c54e1a1e83cd3a2bae4901e3eebf6 ]

    Correct gpio ranges according to i.MX7ULP pinctrl driver:

    gpio_ptc: ONLY pin 0~19 are available;
    gpio_ptd: ONLY pin 0~11 are available;
    gpio_pte: ONLY pin 0~15 are available;
    gpio_ptf: ONLY pin 0~19 are available;

    Fixes: 20434dc92c05 ("ARM: dts: imx: add common imx7ulp dtsi support")
    Signed-off-by: Anson Huang <Anson.Huang@nxp.com>
    Signed-off-by: Shawn Guo <shawnguo@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 23057a8452f65dc983b47c94c422d8cebbf63aae
Author: Matthias Schiffer <matthias.schiffer@ew.tq-group.com>
Date:   Tue Jul 28 12:50:06 2020 +0200

    ARM: dts: ls1021a: fix QuadSPI-memory reg range

    [ Upstream commit 81dbbb417da4d1ac407dca5b434d39d5b6b91ef3 ]

    According to the Reference Manual, the correct size is 512 MiB.

    Without this fix, probing the QSPI fails:

        fsl-quadspi 1550000.spi: ioremap failed for resource
            [mem 0x40000000-0x7fffffff]
        fsl-quadspi 1550000.spi: Freescale QuadSPI probe failed
        fsl-quadspi: probe of 1550000.spi failed with error -12

    Fixes: 85f8ee78ab72 ("ARM: dts: ls1021a: Add support for QSPI with ls1021a SoC")
    Signed-off-by: Matthias Schiffer <matthias.schiffer@ew.tq-group.com>
    Signed-off-by: Shawn Guo <shawnguo@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 82649acf840dede59d0eeef82dfb90507bcd7e57
Author: Po-Hsu Lin <po-hsu.lin@canonical.com>
Date:   Wed Mar 18 10:42:15 2020 +0800
```

```
    selftests/timers: Turn off timeout setting

    [ Upstream commit 5c1e4f7e9e49b6925b1fb5c507d2c614f3edb292 ]

    The following 4 tests in timers can take longer than the default 45
    seconds that added in commit 852c8cbf34d3 ("selftests/kselftest/runner.sh:
    Add 45 second timeout per test") to run:
      * nsleep-lat - 2m7.350s
      * set-timer-lat - 2m0.66s
      * inconsistency-check - 1m45.074s
      * raw_skew - 2m0.013s

    Thus they will be marked as failed with the current 45s setting:
      not ok 3 selftests: timers: nsleep-lat # TIMEOUT
      not ok 4 selftests: timers: set-timer-lat # TIMEOUT
      not ok 6 selftests: timers: inconsistency-check # TIMEOUT
      not ok 7 selftests: timers: raw_skew # TIMEOUT

    Disable the timeout setting for timers can make these tests finish
    properly:
      ok 3 selftests: timers: nsleep-lat
      ok 4 selftests: timers: set-timer-lat
      ok 6 selftests: timers: inconsistency-check
      ok 7 selftests: timers: raw_skew

    https://bugs.launchpad.net/bugs/1864626
    Fixes: 852c8cbf34d3 ("selftests/kselftest/runner.sh: Add 45 second timeout per test")
    Signed-off-by: Po-Hsu Lin <po-hsu.lin@canonical.com>
    Acked-by: John Stultz <john.stultz@linaro.org>
    Signed-off-by: Shuah Khan <skhan@linuxfoundation.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 2e173929f1d4c978c6d0d32c69daae2245db10e3
Author: David Shah <dave@ds0.me>
Date:   Tue Aug 18 10:51:00 2020 +0100

    ARM: dts: omap5: Fix DSI base address and clocks

    [ Upstream commit 6542e2b613c2b1952e83973dc434831332ce8e27 ]

    DSI was not probing due to base address off by 0x1000, and sys_clk
    missing.

    With this patch, the Pyra display works if HDMI is disabled in the
    device tree.

    Fixes: 5a507162f096 ("ARM: dts: Configure interconnect target module for omap5 dsi1")
    Signed-off-by: David Shah <dave@ds0.me>
    Tested-by: H. Nikolaus Schaller <hns@goldelico.com>
    [tony@atomide.com: standardized subject line, added fixes tag]
    Signed-off-by: Tony Lindgren <tony@atomide.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 91dff93e5584278c23b1e787435210e94369de75
Author: Dinh Nguyen <dinguyen@kernel.org>
Date:   Fri Jul 31 10:26:40 2020 -0500

    ARM: dts: socfpga: fix register entry for timer3 on Arria10

    [ Upstream commit 0ff5a4812be4ebd4782bbb555d369636eea164f7 ]

    Fixes the register address for the timer3 entry on Arria10.

    Fixes: 475dc86d08de4 ("arm: dts: socfpga: Add a base DTSI for Altera's Arria10 SOC")
```

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

commit 1dc746dfc129ac62df219ca95ebc5b0296d68da4
Author: Michał Mirosław <mirq-linux@rere.qmqm.pl>
Date:   Wed Aug 12 03:31:38 2020 +0200

    regulator: remove superfluous lock in regulator_resolve_coupling()

    [ Upstream commit a577f3456c0a2fac3dee037c483753e6e68f3e49 ]

    The code modifies rdev, but locks c_rdev instead. Remove the lock
    as this is held together by regulator_list_mutex taken in the caller.

    Fixes: f9503385b187 ("regulator: core: Mutually resolve regulators coupling")
    Signed-off-by: Michał Mirosław <mirq-linux@rere.qmqm.pl>
    Reviewed-by: Dmitry Osipenko <digetx@gmail.com>
    Link:
https://lore.kernel.org/r/25eb81cefb37a646f3e44eaaf1d8ae8881cfde52.1597195321.git.mirq-
linux@rere.qmqm.pl
    Signed-off-by: Mark Brown <broonie@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 8dc8bed9be37ca33556a06066959563986214 09d
Author: Michał Mirosław <mirq-linux@rere.qmqm.pl>
Date:   Wed Aug 12 03:31:34 2020 +0200

    regulator: push allocation in regulator_ena_gpio_request() out of lock

    [ Upstream commit 467bf30142c64f2eb64e2ac67fa4595126230efd ]

    Move another allocation out of regulator_list_mutex-protected region, as
    reclaim might want to take the same lock.

    WARNING: possible circular locking dependency detected
    5.7.13+ #534 Not tainted
    ------------------------------------------------------
    kswapd0/383 is trying to acquire lock:
    c0e5d920 (regulator_list_mutex){+.+.}-{3:3}, at: regulator_lock_dependent+0x54/0x2c0

    but task is already holding lock:
    c0e38518 (fs_reclaim){+.+.}-{0:0}, at: __fs_reclaim_acquire+0x0/0x50

    which lock already depends on the new lock.

    the existing dependency chain (in reverse order) is:

    -> #1 (fs_reclaim){+.+.}-{0:0}:
           fs_reclaim_acquire.part.11+0x40/0x50
           fs_reclaim_acquire+0x24/0x28
           kmem_cache_alloc_trace+0x40/0x1e8
           regulator_register+0x384/0x1630
           devm_regulator_register+0x50/0x84
           reg_fixed_voltage_probe+0x248/0x35c
    [...]
    other info that might help us debug this:

     Possible unsafe locking scenario:

           CPU0                    CPU1
           ----                    ----
      lock(fs_reclaim);
                                   lock(regulator_list_mutex);
                                   lock(fs_reclaim);
      lock(regulator_list_mutex);
```

```
      *** DEADLOCK ***
     [...]
     2 locks held by kswapd0/383:
      #0: c0e38518 (fs_reclaim){+.+.}-{0:0}, at: __fs_reclaim_acquire+0x0/0x50
      #1: cb70e5e0 (hctx->srcu){....}-{0:0}, at: hctx_lock+0x60/0xb8
     [...]

     Fixes: 541d052d7215 ("regulator: core: Only support passing enable GPIO descriptors")
     [this commit only changes context]
     Fixes: f8702f9e4aa7 ("regulator: core: Use ww_mutex for regulators locking")
     [this is when the regulator_list_mutex was introduced in reclaim locking path]

     Signed-off-by: Michał Mirosław <mirq-linux@rere.qmqm.pl>
     Link:
 https://lore.kernel.org/r/41fe6a9670335721b48e8f5195038c3d67a3bf92.1597195321.git.mirq-
 linux@rere.qmqm.pl
     Signed-off-by: Mark Brown <broonie@kernel.org>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit 421d06d4643726a6fd099c825932f13b34049cac
 Author: Adam Ford <aford173@gmail.com>
 Date:   Fri Aug 14 07:24:41 2020 -0500

     ARM: dts: logicpd-som-lv-baseboard: Fix missing video

     [ Upstream commit d1db7b80a6c8c5f81db0e80664d29b374750e2c6 ]

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

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

     Signed-off-by: Adam Ford <aford173@gmail.com>
     Signed-off-by: Tony Lindgren <tony@atomide.com>
     Signed-off-by: Sasha Levin <sashal@kernel.org>

 commit 2a0b7124e5a123b87b6bff2e5093e69c7742149d
 Author: Adam Ford <aford173@gmail.com>
 Date:   Fri Aug 14 07:53:38 2020 -0500

     ARM: dts: logicpd-som-lv-baseboard: Fix broken audio

     [ Upstream commit 4d26e9a028e3d88223e06fa133c3d55af7ddbceb ]

     Older versions of U-Boot would pinmux the whole board, but as
     the bootloader got updated, it started to only pinmux the pins
     it needed, and expected Linux to configure what it needed.

     Unfortunately this caused an issue with the audio, because the
     mcbsp2 pins were configured in the device tree but never
     referenced by the driver. When U-Boot stopped muxing the audio
     pins, the audio died.

     This patch adds the references to the associate the pin controller
     with the mcbsp2 driver which makes audio operate again.

     Fixes: 5cb8b0fa55a9 ("ARM: dts: Move most of logicpd-som-lv-37xx-devkit.dts to logicpd-
 som-lv-baseboard.dtsi")

     Signed-off-by: Adam Ford <aford173@gmail.com>
```

```
    Signed-off-by: Tony Lindgren <tony@atomide.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit c486743abc1bdb10cbe80ccc33d2a151e9af4803
Author: Adam Ford <aford173@gmail.com>
Date:   Sat Aug 8 21:56:10 2020 -0500

    ARM: dts: logicpd-torpedo-baseboard: Fix broken audio

    [ Upstream commit d7dfee67688ac7f2dfd4c3bc70c053ee990c40b5 ]

    Older versions of U-Boot would pinmux the whole board, but as
    the bootloader got updated, it started to only pinmux the pins
    it needed, and expected Linux to configure what it needed.

    Unfortunately this caused an issue with the audio, because the
    mcbsp2 pins were configured in the device tree, they were never
    referenced by the driver. When U-Boot stopped muxing the audio
    pins, the audio died.

    This patch adds the references to the associate the pin controller
    with the mcbsp2 driver which makes audio operate again.

    Fixes: 739f85bba5ab ("ARM: dts: Move most of logicpd-torpedo-37xx-devkit to logicpd-
torpedo-baseboard")

    Signed-off-by: Adam Ford <aford173@gmail.com>
    Signed-off-by: Tony Lindgren <tony@atomide.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit aab25ff4643ebdd2a4462f49005042a48866d554
Author: Jing Xiangfeng <jingxiangfeng@huawei.com>
Date:   Fri Jul 24 11:54:30 2020 +0800

    ARM: OMAP2+: Fix an IS_ERR() vs NULL check in _get_pwrdm()

    [ Upstream commit a58cfdba2039ff2d5758840e97a23a2dedecf1e8 ]

    The of_clk_get() function returns error pointers, it never returns NULL.

    Fixes: 4ea3711aece4 ("ARM: OMAP2+: omap-iommu.c conversion to ti-sysc")
    Signed-off-by: Jing Xiangfeng <jingxiangfeng@huawei.com>
    Signed-off-by: Tony Lindgren <tony@atomide.com>
    Signed-off-by: Sasha Levin <sashal@kernel.org>
```