```
commit 937cc0cc22a27757746f64769d3b1ba3e424e9e4
Author: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
Date:   Sun Jun 9 09:16:18 2019 +0200

    Linux 5.1.8

commit 94395ad60acac7e0fcf2e88cfe5ed6ee3d606139
Author: Nadav Amit <namit@vmware.com>
Date:   Thu Apr 25 17:11:30 2019 -0700

    x86/kprobes: Set instruction page as executable

    [ Upstream commit 7298e24f904224fa79eb8fd7e0fbd78950ccf2db ]

    Set the page as executable after allocation.  This patch is a
    preparatory patch for a following patch that makes module allocated
    pages non-executable.

    While at it, do some small cleanup of what appears to be unnecessary
    masking.

    Signed-off-by: Nadav Amit <namit@vmware.com>
    Signed-off-by: Rick Edgecombe <rick.p.edgecombe@intel.com>
    Signed-off-by: Peter Zijlstra (Intel) <peterz@infradead.org>
    Cc: <akpm@linux-foundation.org>
    Cc: <ard.biesheuvel@linaro.org>
    Cc: <deneen.t.dock@intel.com>
    Cc: <kernel-hardening@lists.openwall.com>
    Cc: <kristen@linux.intel.com>
    Cc: <linux_dti@icloud.com>
    Cc: <will.deacon@arm.com>
    Cc: Andy Lutomirski <luto@kernel.org>
    Cc: Borislav Petkov <bp@alien8.de>
    Cc: Dave Hansen <dave.hansen@linux.intel.com>
    Cc: H. Peter Anvin <hpa@zytor.com>
    Cc: Linus Torvalds <torvalds@linux-foundation.org>
    Cc: Rik van Riel <riel@surriel.com>
    Cc: Thomas Gleixner <tglx@linutronix.de>
    Link: https://lkml.kernel.org/r/20190426001143.4983-11-namit@vmware.com
    Signed-off-by: Ingo Molnar <mingo@kernel.org>
    Signed-off-by: Sasha Levin <sashal@kernel.org>

commit 474ec2dcfbcf268a4124145b5e08847595f67a4c
Author: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
Date:   Wed Jun 5 20:40:30 2019 +0200

    Revert "x86/build: Move _etext to actual end of .text"

    This reverts commit 392bef709659abea614abfe53cf228e7a59876a4.

    It seems to cause lots of problems when using the gold linker, and no
    one really needs this at the moment, so just revert it from the stable
    trees.

    Cc: Sami Tolvanen <samitolvanen@google.com>
    Reported-by: Kees Cook <keescook@chromium.org>
    Cc: Borislav Petkov <bp@suse.de>
    Cc: Linus Torvalds <torvalds@linux-foundation.org>
    Cc: Peter Zijlstra <peterz@infradead.org>
    Cc: Thomas Gleixner <tglx@linutronix.de>
    Reported-by: Alec Ari <neotheuser@gmail.com>
    Cc: Ingo Molnar <mingo@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit 198c6c24db7c754dbb4e75a72c8ed9cb7aabfb69
Author: Noralf Trønnes <noralf@tronnes.org>
Date:   Fri Apr 26 14:47:53 2019 +0200

    drm/cma-helper: Fix drm_gem_cma_free_object()

    commit 23e35c0eb81a23d40c079a7eb187fc321fa1deb7 upstream.

    The logic for freeing an imported buffer with a virtual address is
    broken. It will free the buffer instead of unmapping the dma buf.
    Fix by reversing the if ladder and first check if the buffer is imported.

    Fixes: b9068cde51ee ("drm/cma-helper: Add DRM_GEM_CMA_VMAP_DRIVER_OPS")
    Cc: stable@vger.kernel.org
    Reported-by: "Li, Tingqian" <tingqian.li@intel.com>
    Signed-off-by: Noralf Trønnes <noralf@tronnes.org>
    Reviewed-by: Oleksandr Andrushchenko <oleksandr_andrushchenko@epam.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20190426124753.53722-1-
noralf@tronnes.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 89fd996087277e43cf3aa4c413a74c8bf42e0918
Author: Daniel Vetter <daniel.vetter@ffwll.ch>
Date:   Thu Feb 28 15:49:08 2019 +0100

    drm/lease: Make sure implicit planes are leased

    commit 204f640da6914844b3270b41b29c84f6e3b74083 upstream.

    If userspace doesn't enable universal planes, then we automatically
    add the primary and cursor planes. But for universal userspace there's
    no such check (and maybe we only want to give the lessee one plane,
    maybe not even the primary one), hence we need to check for the
    implied plane.

    v2: don't forget setcrtc ioctl.

    v3: Still allow disabling of the crtc in SETCRTC.

    Cc: stable@vger.kernel.org
    Cc: Keith Packard <keithp@keithp.com>
    Reviewed-by: Boris Brezillon <boris.brezillon@collabora.com>
    Signed-off-by: Daniel Vetter <daniel.vetter@intel.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20190228144910.26488-6-
daniel.vetter@ffwll.ch
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 02158552424608a99bd7724f0bfb1168eab639a2
Author: Vicente Bergas <vicencb@gmail.com>
Date:   Tue Apr 2 13:37:53 2019 +0200

    drm/rockchip: shutdown drm subsystem on shutdown

    commit b8f9d7f37b6af829c34c49d1a4f73ce6ed58e403 upstream.

    As explained by Robin Murphy:
    > the IOMMU shutdown disables paging, so if the VOP is still
    > scanning out then that will result in whatever IOVAs it was using now going
    > straight out onto the bus as physical addresses.

    We had a more radical approach before in commit
    7f3ef5dedb14 ("drm/rockchip: Allow driver to be shutdown on reboot/kexec")
    but that resulted in new warnings and oopses on shutdown on rk3399
    chromeos devices.
```

```
    So second try is resurrecting Vicentes shutdown change which should
    achieve the same result but in a less drastic way.

    Fixes: 63238173b2fa ("Revert "drm/rockchip: Allow driver to be shutdown on
reboot/kexec"")
    Cc: Jeffy Chen <jeffy.chen@rock-chips.com>
    Cc: Robin Murphy <robin.murphy@arm.com>
    Cc: Marc Zyngier <marc.zyngier@arm.com>
    Cc: Brian Norris <briannorris@chromium.org>
    Cc: Doug Anderson <dianders@chromium.org>
    Cc: stable@vger.kernel.org
    Suggested-by: JeffyChen <jeffy.chen@rock-chips.com>
    Suggested-by: Robin Murphy <robin.murphy@arm.com>
    Signed-off-by: Vicente Bergas <vicencb@gmail.com>
    [adapted commit message to explain the history]
    Signed-off-by: Heiko Stuebner <heiko@sntech.de>
    Tested-by: Brian Norris <briannorris@chromium.org>
    Tested-by: Douglas Anderson <dianders@chromium.org>
    Acked-by: Marc Zyngier <marc.zyngier@arm.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20190402113753.10118-1-
heiko@sntech.de
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a7ca9dd7ae521ac6e7047ac7a63fc6690f18a356
Author: Deepak Rawat <drawat@vmware.com>
Date:   Mon Apr 15 10:28:05 2019 -0700

    drm: Expose "FB_DAMAGE_CLIPS" property to atomic aware user-space only

    commit c8f005684c98f4d9942baec13ad98054dbf312a0 upstream.

    Plane property "FB_DAMAGE_CLIPS" can only be used by atomic aware
    user-space, so no point exposing it otherwise.

    Cc: <stable@vger.kernel.org>
    Signed-off-by: Deepak Rawat <drawat@vmware.com>
    Reviewed-by: Daniel Vetter <daniel.vetter@ffwll.ch>
    Fixes: d3b21767821e ("drm: Add a new plane property to send damage during plane update")
    Link: https://patchwork.freedesktop.org/patch/msgid/20190415172814.9840-1-
drawat@vmware.com
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 425a8b82eb57e7b0f93634deabc882dd7fb0e271
Author: Daniel Vetter <daniel.vetter@ffwll.ch>
Date:   Thu Feb 28 15:49:09 2019 +0100

    drm/atomic: Wire file_priv through for property changes

    commit 36e4523aaf4a35de963e190064b53839fa131653 upstream.

    We need this to make sure lessees can only connect their
    plane/connectors to crtc objects they own. And note that this is
    irrespective of whether the lessor is atomic or not, lessor cannot
    prevent lessees from enabling atomic.

    Cc: stable@vger.kernel.org
    Cc: Keith Packard <keithp@keithp.com>
    Reviewed-by: Dave Airlie <airlied@redhat.com>
    Signed-off-by: Daniel Vetter <daniel.vetter@intel.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20190228144910.26488-7-
daniel.vetter@ffwll.ch
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 86d6482085970ccafa74a1c9f5e727f7de657137
Author: Noralf Trønnes <noralf@tronnes.org>
```

```
Date:   Mon Apr 1 16:13:58 2019 +0200

    drm/fb-helper: generic: Call drm_client_add() after setup is done

    commit 6e3f17ee73f7e3c2ef0e2c8fd8624b2ece8ef2c9 upstream.

    Hotplug can happen while drm_fbdev_generic_setup() is running so move
    drm_client_add() call after setup is done to avoid
    drm_fbdev_client_hotplug() running in two threads at the same time.

    Fixes: 9060d7f49376 ("drm/fb-helper: Finish the generic fbdev emulation")
    Cc: stable@vger.kernel.org
    Reported-by: Daniel Vetter <daniel.vetter@ffwll.ch>
    Signed-off-by: Noralf Trønnes <noralf@tronnes.org>
    Reviewed-by: Daniel Vetter <daniel.vetter@ffwll.ch>
    Link: https://patchwork.freedesktop.org/patch/msgid/20190401141358.25309-1-
noralf@tronnes.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 71b59e7aecf6091c1efe422c54a5046954fc2b6e
Author: Philipp Zabel <p.zabel@pengutronix.de>
Date:   Thu May 9 11:01:39 2019 +0200

    drm/imx: ipuv3-plane: fix atomic update status query for non-plus i.MX6Q

    commit 137caa702f2308f7ef03876e164b0d0f3300712a upstream.

    The current buffer check halves the frame rate on non-plus i.MX6Q,
    as the IDMAC current buffer pointer is not yet updated when
    ipu_plane_atomic_update_pending is called from the EOF irq handler.

    Fixes: 70e8a0c71e9 ("drm/imx: ipuv3-plane: add function to query atomic update status")
    Tested-by: Marco Felsch <m.felsch@pengutronix.de>
    Signed-off-by: Philipp Zabel <p.zabel@pengutronix.de>
    Cc: stable@vger.kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3a5c0d36964f6fe1a14da864f115c5444729b388
Author: Jernej Skrabec <jernej.skrabec@siol.net>
Date:   Tue May 14 22:43:37 2019 +0200

    drm/sun4i: Fix sun8i HDMI PHY configuration for > 148.5 MHz

    commit 831adffb3b7b8df4c8e20b7b00843129fb87a166 upstream.

    Vendor provided documentation says that EMP bits should be set to 3 for
    pixel clocks greater than 148.5 MHz.

    Fix that.

    Cc: stable@vger.kernel.org # 4.17+
    Fixes: 4f86e81748fe ("drm/sun4i: Add support for H3 HDMI PHY variant")
    Signed-off-by: Jernej Skrabec <jernej.skrabec@siol.net>
    Signed-off-by: Maxime Ripard <maxime.ripard@bootlin.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20190514204337.11068-3-
jernej.skrabec@siol.net
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 9894c3a4bd6ee2f75d41d063aff154e9e1e76f67
Author: Jernej Skrabec <jernej.skrabec@siol.net>
Date:   Tue May 14 22:43:36 2019 +0200

    drm/sun4i: Fix sun8i HDMI PHY clock initialization

    commit 8a943c6021ba8b95a36c842327e468df1fddd4a7 upstream.
```

```
    Current code initializes HDMI PHY clock driver before reset line is
    deasserted and clocks enabled. Because of that, initial readout of
    clock divider is incorrect (0 instead of 2). This causes any clock
    rate with divider 1 (register value 0) to be set incorrectly.

    Fix this by moving initialization of HDMI PHY clock driver after reset
    line is deasserted and clocks enabled.

    Cc: stable@vger.kernel.org # 4.17+
    Fixes: 4f86e81748fe ("drm/sun4i: Add support for H3 HDMI PHY variant")
    Signed-off-by: Jernej Skrabec <jernej.skrabec@siol.net>
    Signed-off-by: Maxime Ripard <maxime.ripard@bootlin.com>
    Link: https://patchwork.freedesktop.org/patch/msgid/20190514204337.11068-2-
jernej.skrabec@siol.net
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 537ac1cd44e70bf132f6c9bd0a58373d206248c3
Author: Thomas Hellstrom <thellstrom@vmware.com>
Date:   Tue May 7 11:07:53 2019 +0200

    drm/vmwgfx: Don't send drm sysfs hotplug events on initial master set

    commit 63cb44441826e842b7285575b96db631cc9f2505 upstream.

    This may confuse user-space clients like plymouth that opens a drm
    file descriptor as a result of a hotplug event and then generates a
    new event...

    Cc: <stable@vger.kernel.org>
    Fixes: 5ea1734827bb ("drm/vmwgfx: Send a hotplug event at master_set")
    Signed-off-by: Thomas Hellstrom <thellstrom@vmware.com>
    Reviewed-by: Deepak Rawat <drawat@vmware.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c1d4043b0f985c14be12a62d7fd44ca472dd4aa1
Author: Thomas Hellstrom <thellstrom@vmware.com>
Date:   Thu Apr 4 13:25:43 2019 +0000

    drm/vmwgfx: Fix compat mode shader operation

    commit e41c20cf50a8a7d0dfa337a7530590aacef4193b upstream.

    In compat mode, we allowed host-backed user-space with guest-backed
    kernel / device. In this mode, set shader commands was broken since
    no relocations were emitted. Fix this.

    Cc: <stable@vger.kernel.org>
    Fixes: e8c66efbfe3a ("drm/vmwgfx: Make user resource lookups reference-free during
validation")
    Signed-off-by: Thomas Hellstrom <thellstrom@vmware.com>
    Reviewed-by: Brian Paul <brianp@vmware.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 7cb13911b67ad382bc57002ddc384d58022d092d
Author: Thomas Hellstrom <thellstrom@vmware.com>
Date:   Tue May 7 11:10:10 2019 +0200

    drm/vmwgfx: Fix user space handle equal to zero

    commit 8407f8a1d940fe28c4243ad4f0cb6a44dcee88f6 upstream.

    User-space handles equal to zero are interpreted as uninitialized or
    illegal by some drm systems (most notably kms). This means that a
    dumb buffer or surface with a zero user-space handle can never be
```

```
    used as a kms frame-buffer.

    Cc: <stable@vger.kernel.org>
    Fixes: c7eae62666ad ("drm/vmwgfx: Make the object handles idr-generated")
    Signed-off-by: Thomas Hellstrom <thellstrom@vmware.com>
    Reviewed-by: Deepak Rawat <drawat@vmware.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 37dfbf0ad37b897f15d90f8cecc5e0dc5c5bb0b2
Author: Dmitry Osipenko <digetx@gmail.com>
Date:   Thu Mar 7 01:55:19 2019 +0300

    drm/tegra: gem: Fix CPU-cache maintenance for BO's allocated using get_pages()

    commit 61b51fb51c01a519a249d28ec55c6513a13be5a3 upstream.

    The allocated pages need to be invalidated in CPU caches. On ARM32 the
    DMA_BIDIRECTIONAL flag only ensures that data is written-back to DRAM and
    the data stays in CPU cache lines. While the DMA_FROM_DEVICE flag ensures
    that the corresponding CPU cache lines are getting invalidated and nothing
    more, that's exactly what is needed for a newly allocated pages.

    This fixes randomly failing rendercheck tests on Tegra30 using the
    Opentegra driver for tests that use small-sized pixmaps (10x10 and less,
    i.e. 1-2 memory pages) because apparently CPU reads out stale data from
    caches and/or that data is getting evicted to DRAM at the time of HW job
    execution.

    Fixes: bd43c9f0fa1f ("drm/tegra: gem: Map pages via the DMA API")
    Cc: stable <stable@vger.kernel.org>
    Signed-off-by: Dmitry Osipenko <digetx@gmail.com>
    Signed-off-by: Thierry Reding <treding@nvidia.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c011c3e132df518164b9f39e96441daf928c9335
Author: Kees Cook <keescook@chromium.org>
Date:   Mon May 20 11:50:42 2019 -0700

    gcc-plugins: Fix build failures under Darwin host

    commit 7210e060155b9cf557fb13128353c3e494fa5ed3 upstream.

    The gcc-common.h file did not take into account certain macros that
    might have already been defined in the build environment. This updates
    the header to avoid redefining the macros, as seen on a Darwin host
    using gcc 4.9.2:

     HOSTCXX -fPIC scripts/gcc-plugins/arm_ssp_per_task_plugin.o - due to: scripts/gcc-
plugins/gcc-common.h
    In file included from scripts/gcc-plugins/arm_ssp_per_task_plugin.c:3:0:
    scripts/gcc-plugins/gcc-common.h:153:0: warning: "__unused" redefined
    ^
    In file included from /usr/include/stdio.h:64:0,
                     from
/Users/hns/Documents/Projects/QuantumSTEP/System/Library/Frameworks/System.framework/Version
s-jessie/x86_64-apple-darwin15.0.0/gcc/arm-linux-gnueabi/bin/../lib/gcc/arm-linux-
gnueabi/4.9.2/plugin/include/system.h:40,
                     from
/Users/hns/Documents/Projects/QuantumSTEP/System/Library/Frameworks/System.framework/Version
s-jessie/x86_64-apple-darwin15.0.0/gcc/arm-linux-gnueabi/bin/../lib/gcc/arm-linux-
gnueabi/4.9.2/plugin/include/gcc-plugin.h:28,
                     from
/Users/hns/Documents/Projects/QuantumSTEP/System/Library/Frameworks/System.framework/Version
s-jessie/x86_64-apple-darwin15.0.0/gcc/arm-linux-gnueabi/bin/../lib/gcc/arm-linux-
gnueabi/4.9.2/plugin/include/plugin.h:23,
```

```
                    from scripts/gcc-plugins/gcc-common.h:9,
                    from scripts/gcc-plugins/arm_ssp_per_task_plugin.c:3:
    /usr/include/sys/cdefs.h:161:0: note: this is the location of the previous definition
    ^

    Reported-and-tested-by: "H. Nikolaus Schaller" <hns@goldelico.com>
    Fixes: 189af4657186 ("ARM: smp: add support for per-task stack canaries")
    Cc: stable@vger.kernel.org
    Signed-off-by: Kees Cook <keescook@chromium.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 66c076d6fb49dccecebc60ba72e6c2c632fe4a6b
Author: Benjamin Coddington <bcodding@redhat.com>
Date:   Mon May 20 10:33:07 2019 -0400

    Revert "lockd: Show pid of lockd for remote locks"

    commit 141731d15d6eb2fd9aaefbf9b935ce86ae243074 upstream.

    This reverts most of commit b8eee0e90f97 ("lockd: Show pid of lockd for
    remote locks"), which caused remote locks to not be differentiated between
    remote processes for NLM.

    We retain the fixup for setting the client's fl_pid to a negative value.

    Fixes: b8eee0e90f97 ("lockd: Show pid of lockd for remote locks")
    Cc: stable@vger.kernel.org

    Signed-off-by: Benjamin Coddington <bcodding@redhat.com>
    Reviewed-by: XueWei Zhang <xueweiz@google.com>
    Signed-off-by: J. Bruce Fields <bfields@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b3f15a84f20672155dd9b8ce6e958868dbc1adc2
Author: Roberto Bergantinos Corpas <rbergant@redhat.com>
Date:   Tue May 28 09:38:14 2019 +0200

    CIFS: cifs_read_allocate_pages: don't iterate through whole page array on ENOMEM

    commit 31fad7d41e73731f05b8053d17078638cf850fa6 upstream.

     In cifs_read_allocate_pages, in case of ENOMEM, we go through
    whole rdata->pages array but we have failed the allocation before
    nr_pages, therefore we may end up calling put_page with NULL
    pointer, causing oops

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

commit 28341796e0228f8166f8b2e13c3c15952d63a815
Author: Colin Ian King <colin.king@canonical.com>
Date:   Fri May 17 09:12:33 2019 +0100

    cifs: fix memory leak of pneg_inbuf on -EOPNOTSUPP ioctl case

    commit 210782038b54ec8e9059a3c12d6f6ae173efa3a9 upstream.

    Currently in the case where SMB2_ioctl returns the -EOPNOTSUPP error
    there is a memory leak of pneg_inbuf. Fix this by returning via
    the out_free_inbuf exit path that will perform the relevant kfree.

    Addresses-Coverity: ("Resource leak")
```

```
    Fixes: 969ae8e8d4ee ("cifs: Accept validate negotiate if server return
NT_STATUS_NOT_SUPPORTED")
    CC: Stable <stable@vger.kernel.org> # v5.1+
    Signed-off-by: Colin Ian King <colin.king@canonical.com>
    Signed-off-by: Steve French <stfrench@microsoft.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 31674975bfb84003c880b9ab291448c012a108db
Author: Tim Collier <osdevtc@gmail.com>
Date:   Sat May 11 18:40:46 2019 +0100

    staging: wlan-ng: fix adapter initialization failure

    commit a67fedd788182764dc8ed59037c604b7e60349f1 upstream.

    Commit e895f00a8496 ("Staging: wlan-ng: hfa384x_usb.c Fixed too long
    code line warnings.") moved the retrieval of the transfer buffer from
    the URB from the top of function hfa384x_usbin_callback to a point
    after reposting of the URB via a call to submit_rx_urb. The reposting
    of the URB allocates a new transfer buffer so the new buffer is
    retrieved instead of the buffer containing the response passed into
    the callback. This results in failure to initialize the adapter with
    an error reported in the system log (something like "CTLX[1] error:
    state(Request failed)").

    This change moves the retrieval to just before the point where the URB
    is reposted so that the correct transfer buffer is retrieved and
    initialization of the device succeeds.

    Signed-off-by: Tim Collier <osdevtc@gmail.com>
    Fixes: e895f00a8496 ("Staging: wlan-ng: hfa384x_usb.c Fixed too long code line
warnings.")
    Cc: stable <stable@vger.kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 886c20bc7eb98cbf13231709967780e31508bfb2
Author: Dan Carpenter <dan.carpenter@oracle.com>
Date:   Wed May 15 12:38:33 2019 +0300

    staging: vc04_services: prevent integer overflow in create_pagelist()

    commit ca641bae6da977d638458e78cd1487b6160a2718 upstream.

    The create_pagelist() "count" parameter comes from the user in
    vchiq_ioctl() and it could overflow.  If you look at how create_page()
    is called in vchiq_prepare_bulk_data(), then the "size" variable is an
    int so it doesn't make sense to allow negatives or larger than INT_MAX.

    I don't know this code terribly well, but I believe that typical values
    of "count" are typically quite low and I don't think this check will
    affect normal valid uses at all.

    The "pagelist_size" calculation can also overflow on 32 bit systems, but
    not on 64 bit systems.  I have added an integer overflow check for that
    as well.

    The Raspberry PI doesn't offer the same level of memory protection that
    x86 does so these sorts of bugs are probably not super critical to fix.

    Fixes: 71bad7f08641 ("staging: add bcm2708 vchiq driver")
    Signed-off-by: Dan Carpenter <dan.carpenter@oracle.com>
    Cc: stable <stable@vger.kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 26b29a6cbe875f1c71c77d2ed9dffd2b8dee8890
```

```
Author: George G. Davis <george_davis@mentor.com>
Date:   Tue May 14 23:29:34 2019 -0400

    serial: sh-sci: disable DMA for uart_console

    commit 099506cbbc79c0bd52b19cb6b930f256dabc3950 upstream.

    As noted in commit 84b40e3b57ee ("serial: 8250: omap: Disable DMA for
    console UART"), UART console lines use low-level PIO only access functions
    which will conflict with use of the line when DMA is enabled, e.g. when
    the console line is also used for systemd messages. So disable DMA
    support for UART console lines.

    Reported-by: Michael Rodin <mrodin@de.adit-jv.com>
    Link: https://patchwork.kernel.org/patch/10929511/
    Tested-by: Eugeniu Rosca <erosca@de.adit-jv.com>
    Reviewed-by: Simon Horman <horms+renesas@verge.net.au>
    Reviewed-by: Wolfram Sang <wsa+renesas@sang-engineering.com>
    Reviewed-by: Geert Uytterhoeven <geert+renesas@glider.be>
    Cc: stable@vger.kernel.org
    Signed-off-by: George G. Davis <george_davis@mentor.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 863f40c7a7036766c145f09dba6e1545c5eccdb0
Author: Grzegorz Halat <ghalat@redhat.com>
Date:   Fri Apr 26 16:59:46 2019 +0200

    vt/fbcon: deinitialize resources in visual_init() after failed memory allocation

    commit a1ad1cc9704f64c169261a76e1aee1cf1ae51832 upstream.

    After memory allocation failure vc_allocate() doesn't clean up data
    which has been initialized in visual_init(). In case of fbcon this
    leads to divide-by-0 in fbcon_init() on next open of the same tty.

    memory allocation in vc_allocate() may fail here:
    1097:     vc->vc_screenbuf = kzalloc(vc->vc_screenbuf_size, GFP_KERNEL);

    on next open() fbcon_init() skips vc_font.data initialization:
    1088:     if (!p->fontdata) {

    division by zero in fbcon_init() happens here:
    1149:     new_cols /= vc->vc_font.width;

    Additional check is needed in fbcon_deinit() to prevent
    usage of uninitialized vc_screenbuf:

    1251:        if (vc->vc_hi_font_mask && vc->vc_screenbuf)
    1252:                set_vc_hi_font(vc, false);

    Crash:

     #6 [ffffc90001eafa60] divide_error at ffffffff81a00be4
        [exception RIP: fbcon_init+463]
        RIP: ffffffff814b860f  RSP: ffffc90001eafb18  RFLAGS: 00010246
    ...
     #7 [ffffc90001eafb60] visual_init at ffffffff8154c36e
     #8 [ffffc90001eafb80] vc_allocate at ffffffff8154f53c
     #9 [ffffc90001eafbc8] con_install at ffffffff8154f624
    ...

    Signed-off-by: Grzegorz Halat <ghalat@redhat.com>
    Reviewed-by: Oleksandr Natalenko <oleksandr@redhat.com>
    Acked-by: Bartlomiej Zolnierkiewicz <b.zolnierkie@samsung.com>
```

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

commit 62f3e22c77047fc29ca798c22fe85922793d2586
Author: Peng Fan <peng.fan@nxp.com>
Date:   Mon May 20 02:03:19 2019 +0000

    clk: imx: imx8mm: fix int pll clk gate

    commit b3fddd5b100e4aee4f7ec58360435024971dea47 upstream.

    To Frac pll, the gate shift is 13, however to Int PLL the gate shift
    is 11.

    Cc: <stable@vger.kernel.org>
    Fixes: ba5625c3e27 ("clk: imx: Add clock driver support for imx8mm")
    Signed-off-by: Peng Fan <peng.fan@nxp.com>
    Reviewed-by: Fabio Estevam <festevam@gmail.com>
    Reviewed-by: Jacky Bai <ping.bai@nxp.com>
    Signed-off-by: Stephen Boyd <sboyd@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 2f20066d32a90b034ccfbc0b46be21a77e59bdb0
Author: Roberto Sassu <roberto.sassu@huawei.com>
Date:   Wed May 29 15:30:33 2019 +0200

    evm: check hash algorithm passed to init_desc()

    commit 221be106d75c1b511973301542f47d6000d0b63e upstream.

    This patch prevents memory access beyond the evm_tfm array by checking the
    validity of the index (hash algorithm) passed to init_desc(). The hash
    algorithm can be arbitrarily set if the security.ima xattr type is not
    EVM_XATTR_HMAC.

    Fixes: 5feeb61183dde ("evm: Allow non-SHA1 digital signatures")
    Signed-off-by: Roberto Sassu <roberto.sassu@huawei.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Mimi Zohar <zohar@linux.ibm.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b2164f6c3bc97917e51d59fc42ac0d67e400d483
Author: Roberto Sassu <roberto.sassu@huawei.com>
Date:   Wed May 29 15:30:35 2019 +0200

    ima: show rules with IMA_INMASK correctly

    commit 8cdc23a3d9ec0944000ad43bad588e36afdc38cd upstream.

    Show the '^' character when a policy rule has flag IMA_INMASK.

    Fixes: 80eae209d63ac ("IMA: allow reading back the current IMA policy")
    Signed-off-by: Roberto Sassu <roberto.sassu@huawei.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Mimi Zohar <zohar@linux.ibm.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 407eb63f2aa4f1f9b0962df530f3aeec401d6e3e
Author: Petr Vorel <pvorel@suse.cz>
Date:   Wed May 15 08:18:07 2019 +0200

    ima: fix wrong signed policy requirement when not appraising

    commit f40019475bbbe9b455e7fd4385fcf13896c492ca upstream.
```

```
    Kernel booted just with ima_policy=tcb (not with
    ima_policy=appraise_tcb) shouldn't require signed policy.

    Regression found with LTP test ima_policy.sh.

    Fixes: c52657d93b05 ("ima: refactor ima_init_policy()")
    Cc: stable@vger.kernel.org  (linux-5.0)
    Signed-off-by: Petr Vorel <pvorel@suse.cz>
    Signed-off-by: Mimi Zohar <zohar@linux.ibm.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c59e2a64128e30a16394b2fa09b7470c2e2f6f68
Author: Scott Wood <swood@redhat.com>
Date:   Tue Apr 23 17:48:07 2019 -0500

    x86/ima: Check EFI_RUNTIME_SERVICES before using

    commit 558b523d46289f111d53d7c42211069063be5985 upstream.

    Checking efi_enabled(EFI_BOOT) is not sufficient to ensure that
    EFI runtime services are available, e.g. if efi=noruntime is used.

    Without this, I get an oops on a PREEMPT_RT kernel where efi=noruntime is
    the default.

    Fixes: 399574c64eaf94e8 ("x86/ima: retry detecting secure boot mode")
    Cc: stable@vger.kernel.org  (linux-5.0)
    Signed-off-by: Scott Wood <swood@redhat.com>
    Signed-off-by: Mimi Zohar <zohar@linux.ibm.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit bd411a6249654e42ff5d5372e84c2db4adeebddb
Author: Jonathan Corbet <corbet@lwn.net>
Date:   Tue May 21 14:23:43 2019 -0600

    doc: Cope with Sphinx logging deprecations

    commit 096ea522e84ea68f8e6c41e5e7294731a81e29bc upstream.

    Recent versions of sphinx will emit messages like:

      Documentation/sphinx/kerneldoc.py:103:
         RemovedInSphinx20Warning: app.warning() is now deprecated.
         Use sphinx.util.logging instead.

    Switch to sphinx.util.logging to make this unsightly message go away.
    Alas, that interface was only added in version 1.6, so we have to add a
    version check to keep things working with older sphinxes.

    Cc: stable@vger.kernel.org
    Signed-off-by: Jonathan Corbet <corbet@lwn.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 32a4c9e04c09af21d423ab57a5819b854fa5f3e2
Author: Jonathan Corbet <corbet@lwn.net>
Date:   Tue May 21 14:42:34 2019 -0600

    doc: Cope with the deprecation of AutoReporter

    commit 2404dad1f67f8917e30fc22a85e0dbcc85b99955 upstream.

    AutoReporter is going away; recent versions of sphinx emit a warning like:

      Documentation/sphinx/kerneldoc.py:125:
          RemovedInSphinx20Warning: AutodocReporter is now deprecated.
```

```
          Use sphinx.util.docutils.switch_source_input() instead.

    Make the switch.  But switch_source_input() only showed up in 1.7, so we
    have to do ugly version checks to keep things working in older versions.

    Cc: stable@vger.kernel.org
    Signed-off-by: Jonathan Corbet <corbet@lwn.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit d6fe20f24d9e7c9e7781d099c9224d3c26fca64e
Author: Jonathan Corbet <corbet@lwn.net>
Date:   Wed May 22 14:30:45 2019 -0600

    docs: Fix conf.py for Sphinx 2.0

    commit 3bc8088464712fdcb078eefb68837ccfcc413c88 upstream.

    Our version check in Documentation/conf.py never envisioned a world where
    Sphinx moved beyond 1.x.  Now that the unthinkable has happened, fix our
    version check to handle higher version numbers correctly.

    Cc: stable@vger.kernel.org
    Signed-off-by: Jonathan Corbet <corbet@lwn.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit ff6503c80a92976af811dd88760543b868c7ac7b
Author: Catalin Marinas <catalin.marinas@arm.com>
Date:   Tue May 28 12:39:34 2019 +0100

    arm64: Fix the arm64_personality() syscall wrapper redirection

    commit 00377277166bac6939d8f72b429301369acaf2d8 upstream.

    Following commit 4378a7d4be30 ("arm64: implement syscall wrappers"), the
    syscall function names gained the '__arm64_' prefix. Ensure that we
    have the correct #define for redirecting a default syscall through a
    wrapper.

    Fixes: 4378a7d4be30 ("arm64: implement syscall wrappers")
    Cc: <stable@vger.kernel.org> # 4.19.x-
    Acked-by: Mark Rutland <mark.rutland@arm.com>
    Signed-off-by: Catalin Marinas <catalin.marinas@arm.com>
    Signed-off-by: Will Deacon <will.deacon@arm.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 6b036c707d5b4476b6be9213fc5253c28324589d
Author: Suzuki K Poulose <suzuki.poulose@arm.com>
Date:   Fri May 31 22:30:59 2019 -0700

    mm, compaction: make sure we isolate a valid PFN

    commit e577c8b64d58fe307ea4d5149d31615df2d90861 upstream.

    When we have holes in a normal memory zone, we could endup having
    cached_migrate_pfns which may not necessarily be valid, under heavy memory
    pressure with swapping enabled ( via __reset_isolation_suitable(),
    triggered by kswapd).

    Later if we fail to find a page via fast_isolate_freepages(), we may end
    up using the migrate_pfn we started the search with, as valid page.  This
    could lead to accessing NULL pointer derefernces like below, due to an
    invalid mem_section pointer.

    Unable to handle kernel NULL pointer dereference at virtual address 0000000000000008
[47/1825]
```

```
      Mem abort info:
        ESR = 0x96000004
        Exception class = DABT (current EL), IL = 32 bits
        SET = 0, FnV = 0
        EA = 0, S1PTW = 0
      Data abort info:
        ISV = 0, ISS = 0x00000004
        CM = 0, WnR = 0
      user pgtable: 4k pages, 48-bit VAs, pgdp = 0000000082f94ae9
      [0000000000000008] pgd=0000000000000000
      Internal error: Oops: 96000004 [#1] SMP
      ...
      CPU: 10 PID: 6080 Comm: qemu-system-aar Not tainted 510-rc1+ #6
      Hardware name: AmpereComputing(R) OSPREY EV-883832-X3-0001/OSPREY, BIOS 4819 09/25/2018
      pstate: 60000005 (nZCv daif -PAN -UAO)
      pc : set_pfnblock_flags_mask+0x58/0xe8
      lr : compaction_alloc+0x300/0x950
      [...]
      Process qemu-system-aar (pid: 6080, stack limit = 0x0000000095070da5)
      Call trace:
       set_pfnblock_flags_mask+0x58/0xe8
       compaction_alloc+0x300/0x950
       migrate_pages+0x1a4/0xbb0
       compact_zone+0x750/0xde8
       compact_zone_order+0xd8/0x118
       try_to_compact_pages+0xb4/0x290
       __alloc_pages_direct_compact+0x84/0x1e0
       __alloc_pages_nodemask+0x5e0/0xe18
       alloc_pages_vma+0x1cc/0x210
       do_huge_pmd_anonymous_page+0x108/0x7c8
       __handle_mm_fault+0xdd4/0x1190
       handle_mm_fault+0x114/0x1c0
       __get_user_pages+0x198/0x3c0
       get_user_pages_unlocked+0xb4/0x1d8
       __gfn_to_pfn_memslot+0x12c/0x3b8
       gfn_to_pfn_prot+0x4c/0x60
       kvm_handle_guest_abort+0x4b0/0xcd8
       handle_exit+0x140/0x1b8
       kvm_arch_vcpu_ioctl_run+0x260/0x768
       kvm_vcpu_ioctl+0x490/0x898
       do_vfs_ioctl+0xc4/0x898
       ksys_ioctl+0x8c/0xa0
       __arm64_sys_ioctl+0x28/0x38
       el0_svc_common+0x74/0x118
       el0_svc_handler+0x38/0x78
       el0_svc+0x8/0xc
      Code: f8607840 f100001f 8b011401 9a801020 (f9400400)
      ---[ end trace af6a35219325a9b6 ]---

    The issue was reported on an arm64 server with 128GB with holes in the
    zone (e.g, [32GB@4GB, 96GB@544GB]), with a swap device enabled, while
    running 100 KVM guest instances.

    This patch fixes the issue by ensuring that the page belongs to a valid
    PFN when we fallback to using the lower limit of the scan range upon
    failure in fast_isolate_freepages().

    Link: http://lkml.kernel.org/r/1558711908-15688-1-git-send-email-suzuki.poulose@arm.com
    Fixes: 5a811889de10f1eb ("mm, compaction: use free lists to quickly locate a migration
 target")
    Signed-off-by: Suzuki K Poulose <suzuki.poulose@arm.com>
    Reported-by: Marc Zyngier <marc.zyngier@arm.com>
    Reviewed-by: Mel Gorman <mgorman@techsingularity.net>
    Reviewed-by: Anshuman Khandual <anshuman.khandual@arm.com>
    Cc: Michal Hocko <mhocko@suse.com>
```

```
    Cc: Qian Cai <cai@lca.pw>
    Cc: Marc Zyngier <marc.zyngier@arm.com>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 013d8fcb191e7af39dc7d1062a46670c4f7dc461
Author: Eric W. Biederman <ebiederm@xmission.com>
Date:   Thu May 23 11:11:19 2019 -0500

    signal/arm64: Use force_sig not force_sig_fault for SIGKILL

    commit d76cac67db40c172791ce07948367b96a758e45b upstream.

    I don't think this is userspace visible but SIGKILL does not have
    any si_codes that use the fault member of the siginfo union.  Correct
    this the simple way and call force_sig instead of force_sig_fault when
    the signal is SIGKILL.

    The two know places where synchronous SIGKILL are generated are
    do_bad_area and fpsimd_save.  The call paths to force_sig_fault are:
    do_bad_area
      arm64_force_sig_fault
        force_sig_fault
    force_signal_inject
      arm64_notify_die
        arm64_force_sig_fault
           force_sig_fault

    Which means correcting this in arm64_force_sig_fault is enough
    to ensure the arm64 code is not misusing the generic code, which
    could lead to maintenance problems later.

    Cc: stable@vger.kernel.org
    Cc: Dave Martin <Dave.Martin@arm.com>
    Cc: James Morse <james.morse@arm.com>
    Cc: Will Deacon <will.deacon@arm.com>
    Fixes: af40ff687bc9 ("arm64: signal: Ensure si_code is valid for all fault signals")
    Signed-off-by: "Eric W. Biederman" <ebiederm@xmission.com>
    Signed-off-by: Will Deacon <will.deacon@arm.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 02ea8867ae63664c2b9d6c802db21a77bd03fdb0
Author: Zhenliang Wei <weizhenliang@huawei.com>
Date:   Fri May 31 22:30:52 2019 -0700

    kernel/signal.c: trace_signal_deliver when signal_group_exit

    commit 98af37d624ed8c83f1953b1b6b2f6866011fc064 upstream.

    In the fixes commit, removing SIGKILL from each thread signal mask and
    executing "goto fatal" directly will skip the call to
    "trace_signal_deliver".  At this point, the delivery tracking of the
    SIGKILL signal will be inaccurate.

    Therefore, we need to add trace_signal_deliver before "goto fatal" after
    executing sigdelset.

    Note: SEND_SIG_NOINFO matches the fact that SIGKILL doesn't have any info.

    Link: http://lkml.kernel.org/r/20190425025812.91424-1-weizhenliang@huawei.com
    Fixes: cf43a757fd4944 ("signal: Restore the stop PTRACE_EVENT_EXIT")
    Signed-off-by: Zhenliang Wei <weizhenliang@huawei.com>
    Reviewed-by: Christian Brauner <christian@brauner.io>
```

```
    Reviewed-by: Oleg Nesterov <oleg@redhat.com>
    Cc: Eric W. Biederman <ebiederm@xmission.com>
    Cc: Ivan Delalande <colona@arista.com>
    Cc: Arnd Bergmann <arnd@arndb.de>
    Cc: Thomas Gleixner <tglx@linutronix.de>
    Cc: Deepa Dinamani <deepa.kernel@gmail.com>
    Cc: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 2257b0402edfc887a822ee6131b7ed4d324698c4
Author: Nathan Chancellor <natechancellor@gmail.com>
Date:   Fri May 31 22:30:42 2019 -0700

    kasan: initialize tag to 0xff in __kasan_kmalloc

    commit 0600597c854e53d2f9b7a6a718c1da2b8b4cb4db upstream.

    When building with -Wuninitialized and CONFIG_KASAN_SW_TAGS unset, Clang
    warns:

    mm/kasan/common.c:484:40: warning: variable 'tag' is uninitialized when
    used here [-Wuninitialized]
            kasan_unpoison_shadow(set_tag(object, tag), size);
                                          ^~~

    set_tag ignores tag in this configuration but clang doesn't realize it at
    this point in its pipeline, as it points to arch_kasan_set_tag as being
    the point where it is used, which will later be expanded to (void
    *)(object) without a use of tag.  Initialize tag to 0xff, as it removes
    this warning and doesn't change the meaning of the code.

    Link: https://github.com/ClangBuiltLinux/linux/issues/465
    Link: http://lkml.kernel.org/r/20190502163057.6603-1-natechancellor@gmail.com
    Fixes: 7f94ffbc4c6a ("kasan: add hooks implementation for tag-based mode")
    Signed-off-by: Nathan Chancellor <natechancellor@gmail.com>
    Reviewed-by: Andrey Konovalov <andreyknvl@google.com>
    Reviewed-by: Andrey Ryabinin <aryabinin@virtuozzo.com>
    Cc: Alexander Potapenko <glider@google.com>
    Cc: Dmitry Vyukov <dvyukov@google.com>
    Cc: Nick Desaulniers <ndesaulniers@google.com>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 770368348c84c613aef82d81d68208db733b54f0
Author: Jiri Slaby <jslaby@suse.cz>
Date:   Fri May 31 22:30:26 2019 -0700

    memcg: make it work on sparse non-0-node systems

    commit 3e8589963773a5c23e2f1fe4bcad0e9a90b7f471 upstream.

    We have a single node system with node 0 disabled:
      Scanning NUMA topology in Northbridge 24
      Number of physical nodes 2
      Skipping disabled node 0
      Node 1 MemBase 0000000000000000 Limit 00000000fbff0000
      NODE_DATA(1) allocated [mem 0xfbfda000-0xfbfeffff]

    This causes crashes in memcg when system boots:
      BUG: unable to handle kernel NULL pointer dereference at 0000000000000008
```

```
       #PF error: [normal kernel read fault]
     ...
       RIP: 0010:list_lru_add+0x94/0x170
     ...
       Call Trace:
        d_lru_add+0x44/0x50
        dput.part.34+0xfc/0x110
        __fput+0x108/0x230
        task_work_run+0x9f/0xc0
        exit_to_usermode_loop+0xf5/0x100

     It is reproducible as far as 4.12.  I did not try older kernels.  You have
     to have a new enough systemd, e.g.  241 (the reason is unknown -- was not
     investigated).  Cannot be reproduced with systemd 234.

     The system crashes because the size of lru array is never updated in
     memcg_update_all_list_lrus and the reads are past the zero-sized array,
     causing dereferences of random memory.

     The root cause are list_lru_memcg_aware checks in the list_lru code.  The
     test in list_lru_memcg_aware is broken: it assumes node 0 is always
     present, but it is not true on some systems as can be seen above.

     So fix this by avoiding checks on node 0.  Remember the memcg-awareness by
     a bool flag in struct list_lru.

     Link: http://lkml.kernel.org/r/20190522091940.3615-1-jslaby@suse.cz
     Fixes: 60d3fd32a7a9 ("list_lru: introduce per-memcg lists")
     Signed-off-by: Jiri Slaby <jslaby@suse.cz>
     Acked-by: Michal Hocko <mhocko@suse.com>
     Suggested-by: Vladimir Davydov <vdavydov.dev@gmail.com>
     Acked-by: Vladimir Davydov <vdavydov.dev@gmail.com>
     Reviewed-by: Shakeel Butt <shakeelb@google.com>
     Cc: Johannes Weiner <hannes@cmpxchg.org>
     Cc: Raghavendra K T <raghavendra.kt@linux.vnet.ibm.com>
     Cc: <stable@vger.kernel.org>
     Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
     Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
     Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

  commit f96109988a8edd15c668d521af49f51797e4621e
  Author: Chris Down <chris@chrisdown.name>
  Date:   Fri May 31 22:30:22 2019 -0700

     mm, memcg: consider subtrees in memory.events

     commit 9852ae3fe5293264f01c49f2571ef7688f7823ce upstream.

     memory.stat and other files already consider subtrees in their output, and
     we should too in order to not present an inconsistent interface.

     The current situation is fairly confusing, because people interacting with
     cgroups expect hierarchical behaviour in the vein of memory.stat,
     cgroup.events, and other files.  For example, this causes confusion when
     debugging reclaim events under low, as currently these always read "0" at
     non-leaf memcg nodes, which frequently causes people to misdiagnose breach
     behaviour.  The same confusion applies to other counters in this file when
     debugging issues.

     Aggregation is done at write time instead of at read-time since these
     counters aren't hot (unlike memory.stat which is per-page, so it does it
     at read time), and it makes sense to bundle this with the file
     notifications.

     After this patch, events are propagated up the hierarchy:
```

```
        [root@ktst ~]# cat /sys/fs/cgroup/system.slice/memory.events
        low 0
        high 0
        max 0
        oom 0
        oom_kill 0
        [root@ktst ~]# systemd-run -p MemoryMax=1 true
        Running as unit: run-r251162a189fb4562b9dabfdc9b0422f5.service
        [root@ktst ~]# cat /sys/fs/cgroup/system.slice/memory.events
        low 0
        high 0
        max 7
        oom 1
        oom_kill 1

    As this is a change in behaviour, this can be reverted to the old
    behaviour by mounting with the `memory_localevents' flag set.  However, we
    use the new behaviour by default as there's a lack of evidence that there
    are any current users of memory.events that would find this change
    undesirable.

    akpm: this is a behaviour change, so Cc:stable.  THis is so that
    forthcoming distros which use cgroup v2 are more likely to pick up the
    revised behaviour.

    Link: http://lkml.kernel.org/r/20190208224419.GA24772@chrisdown.name
    Signed-off-by: Chris Down <chris@chrisdown.name>
    Acked-by: Johannes Weiner <hannes@cmpxchg.org>
    Reviewed-by: Shakeel Butt <shakeelb@google.com>
    Cc: Michal Hocko <mhocko@kernel.org>
    Cc: Tejun Heo <tj@kernel.org>
    Cc: Roman Gushchin <guro@fb.com>
    Cc: Dennis Zhou <dennis@kernel.org>
    Cc: Suren Baghdasaryan <surenb@google.com>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 15ff2c6a8838bef403adc538f5a2984e848d1afd
Author: Joe Burmeister <joe.burmeister@devtank.co.uk>
Date:   Mon May 13 11:23:57 2019 +0100

    tty: max310x: Fix external crystal register setup

    commit 5d24f455c182d5116dd5db8e1dc501115ecc9c2c upstream.

    The datasheet states:

      Bit 4: ClockEnSet the ClockEn bit high to enable an external clocking
    (crystal or clock generator at XIN). Set the ClockEn bit to 0 to disable
    clocking
      Bit 1: CrystalEnSet the CrystalEn bit high to enable the crystal
    oscillator. When using an external clock source at XIN, CrystalEn must
    be set low.

    The bit 4, MAX310X_CLKSRC_EXTCLK_BIT, should be set and was not.

    This was required to make the MAX3107 with an external crystal on our
    board able to send or receive data.

    Signed-off-by: Joe Burmeister <joe.burmeister@devtank.co.uk>
    Cc: stable <stable@vger.kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit 5f436d8d774d87667a90c0611764867ba92d74aa
Author: Jorge Ramirez-Ortiz <jorge.ramirez-ortiz@linaro.org>
Date:   Mon May 20 20:38:48 2019 +0200

    tty: serial: msm_serial: Fix XON/XOFF

    commit 61c0e37950b88bad590056286c1d766b1f167f4e upstream.

    When the tty layer requests the uart to throttle, the current code
    executing in msm_serial will trigger "Bad mode in Error Handler" and
    generate an invalid stack frame in pstore before rebooting (that is if
    pstore is indeed configured: otherwise the user shall just notice a
    reboot with no further information dumped to the console).

    This patch replaces the PIO byte accessor with the word accessor
    already used in PIO mode.

    Fixes: 68252424a7c7 ("tty: serial: msm: Support big-endian CPUs")
    Cc: stable@vger.kernel.org
    Signed-off-by: Jorge Ramirez-Ortiz <jorge.ramirez-ortiz@linaro.org>
    Reviewed-by: Bjorn Andersson <bjorn.andersson@linaro.org>
    Reviewed-by: Stephen Boyd <swboyd@chromium.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 38338dcab7724a06a6a7a8b4215ce7c4954d4fbf
Author: Masahisa Kojima <masahisa.kojima@linaro.org>
Date:   Tue May 21 10:33:50 2019 +0900

    i2c: synquacer: fix synquacer_i2c_doxfer() return value

    commit ff9378904d9d7a3fcb8406604e089e535e357b1d upstream.

    master_xfer should return the number of messages successfully
    processed.

    Fixes: 0d676a6c4390 ("i2c: add support for Socionext SynQuacer I2C controller")
    Cc: <stable@vger.kernel.org> # v4.19+
    Signed-off-by: Okamoto Satoru <okamoto.satoru@socionext.com>
    Signed-off-by: Masahisa Kojima <masahisa.kojima@linaro.org>
    Acked-by: Ard Biesheuvel <ard.biesheuvel@linaro.org>
    Signed-off-by: Wolfram Sang <wsa@the-dreams.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1d7920d31b732f3a93f691768d9ca756f6404a7e
Author: Vadim Pasternak <vadimp@mellanox.com>
Date:   Thu May 16 17:15:41 2019 +0000

    i2c: mlxcpld: Fix wrong initialization order in probe

    commit 13067ef73f337336e3149f5bb9f3fd05fe7f87a0 upstream.

    Fix wrong order in probing routine initialization - field `base_addr'
    is used before it's initialized. Move assignment of 'priv->base_addr`
    to the beginning, prior the call to mlxcpld_i2c_read_comm().
    Wrong order caused the first read of capability register to be executed
    at wrong offset 0x0 instead of 0x2000. By chance it was a "good
    garbage" at 0x0 offset.

    Fixes: 313ce648b5a4 ("i2c: mlxcpld: Add support for extended transaction length for i2c-
mlxcpld")
    Signed-off-by: Vadim Pasternak <vadimp@mellanox.com>
    Signed-off-by: Wolfram Sang <wsa@the-dreams.de>
    Cc: stable@kernel.org
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit cd68344b283174a9b38e9488d5a929464e1f417c
Author: Lyude Paul <lyude@redhat.com>
Date:   Tue Apr 9 16:23:30 2019 -0400

    drm/nouveau/i2c: Disable i2c bus access after ->fini()

    commit 342406e4fbba9a174125fbfe6aeac3d64ef90f76 upstream.

    For a while, we've had the problem of i2c bus access not grabbing
    a runtime PM ref when it's being used in userspace by i2c-dev, resulting
    in nouveau spamming the kernel log with errors if anything attempts to
    access the i2c bus while the GPU is in runtime suspend. An example:

    [  130.078386] nouveau 0000:01:00.0: i2c: aux 000d: begin idle timeout ffffffff

    Since the GPU is in runtime suspend, the MMIO region that the i2c bus is
    on isn't accessible. On x86, the standard behavior for accessing an
    unavailable MMIO region is to just return ~0.

    Except, that turned out to be a lie. While computers with a clean
    concious will return ~0 in this scenario, some machines will actually
    completely hang a CPU on certian bad MMIO accesses. This was witnessed
    with someone's Lenovo ThinkPad P50, where sensors-detect attempting to
    access the i2c bus while the GPU was suspended would result in a CPU
    hang:

      CPU: 5 PID: 12438 Comm: sensors-detect Not tainted 5.0.0-0.rc4.git3.1.fc30.x86_64 #1
      Hardware name: LENOVO 20EQS64N17/20EQS64N17, BIOS N1EET74W (1.47 ) 11/21/2017
      RIP: 0010:ioread32+0x2b/0x30
      Code: 81 ff ff ff 03 00 77 20 48 81 ff 00 00 01 00 76 05 0f b7 d7 ed c3
      48 c7 c6 e1 0c 36 96 e8 2d ff ff ff b8 ff ff ff ff c3 8b 07 <c3> 0f 1f
      40 00 49 89 f0 48 81 fe ff ff 03 00 76 04 40 88 3e c3 48
      RSP: 0018:ffffaac3c5007b48 EFLAGS: 00000292 ORIG_RAX: ffffffffffffff13
      RAX: 0000000001111000 RBX: 0000000001111000 RCX: 0000043017a97186
      RDX: 0000000000000aaa RSI: 0000000000000005 RDI: ffffaac3c400e4e4
      RBP: ffff9e6443902c00 R08: ffffaac3c400e4e4 R09: ffffaac3c5007be7
      R10: 0000000000000004 R11: 0000000000000001 R12: ffff9e6445dd0000
      R13: 000000000000e4e4 R14: 00000000000003c4 R15: 0000000000000000
      FS:  00007f253155a740(0000) GS:ffff9e644f600000(0000) knlGS:0000000000000000
      CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
      CR2: 00005630d1500358 CR3: 0000000417c44006 CR4: 00000000003606e0
      DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000
      DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400
      Call Trace:
       g94_i2c_aux_xfer+0x326/0x850 [nouveau]
       nvkm_i2c_aux_i2c_xfer+0x9e/0x140 [nouveau]
       __i2c_transfer+0x14b/0x620
       i2c_smbus_xfer_emulated+0x159/0x680
       ? _raw_spin_unlock_irqrestore+0x1/0x60
       ? rt_mutex_slowlock.constprop.0+0x13d/0x1e0
       ? __lock_is_held+0x59/0xa0
       __i2c_smbus_xfer+0x138/0x5a0
       i2c_smbus_xfer+0x4f/0x80
       i2cdev_ioctl_smbus+0x162/0x2d0 [i2c_dev]
       i2cdev_ioctl+0x1db/0x2c0 [i2c_dev]
       do_vfs_ioctl+0x408/0x750
       ksys_ioctl+0x5e/0x90
       __x64_sys_ioctl+0x16/0x20
       do_syscall_64+0x60/0x1e0
       entry_SYSCALL_64_after_hwframe+0x49/0xbe
      RIP: 0033:0x7f25317f546b
      Code: 0f 1e fa 48 8b 05 1d da 0c 00 64 c7 00 26 00 00 00 48 c7 c0 ff ff
      ff ff c3 66 0f 1f 44 00 00 f3 0f 1e fa b8 10 00 00 00 0f 05 <48> 3d 01
      f0 ff ff 73 01 c3 48 8b 0d ed d9 0c 00 f7 d8 64 89 01 48
```

```
       RSP: 002b:00007ffc88caab68 EFLAGS: 00000246 ORIG_RAX: 0000000000000010
       RAX: ffffffffffffffda RBX: 00005630d0fe7260 RCX: 00007f25317f546b
       RDX: 00005630d1598e80 RSI: 0000000000000720 RDI: 0000000000000003
       RBP: 00005630d155b968 R08: 0000000000000001 R09: 00005630d15a1da0
       R10: 0000000000000070 R11: 0000000000000246 R12: 00005630d1598e80
       R13: 00005630d12f3d28 R14: 0000000000000720 R15: 00005630d12f3ce0
       watchdog: BUG: soft lockup - CPU#5 stuck for 23s! [sensors-detect:12438]

    Yikes! While I wanted to try to make it so that accessing an i2c bus on
    nouveau would wake up the GPU as needed, airlied pointed out that pretty
    much any usecase for userspace accessing an i2c bus on a GPU (mainly for
    the DDC brightness control that some displays have) is going to only be
    useful while there's at least one display enabled on the GPU anyway, and
    the GPU never sleeps while there's displays running.

    Since teaching the i2c bus to wake up the GPU on userspace accesses is a
    good deal more difficult than it might seem, mostly due to the fact that
    we have to use the i2c bus during runtime resume of the GPU, we instead
    opt for the easiest solution: don't let userspace access i2c busses on
    the GPU at all while it's in runtime suspend.

    Changes since v1:
    * Also disable i2c busses that run over DP AUX

    Signed-off-by: Lyude Paul <lyude@redhat.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Ben Skeggs <bskeggs@redhat.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 1f41c93ad28072726cb7e98a10c105a824bb63dd
Author: Thomas Huth <thuth@redhat.com>
Date:   Thu May 23 18:43:08 2019 +0200

    KVM: s390: Do not report unusabled IDs via KVM_CAP_MAX_VCPU_ID

    commit a86cb413f4bf273a9d341a3ab2c2ca44e12eb317 upstream.

    KVM_CAP_MAX_VCPU_ID is currently always reporting KVM_MAX_VCPU_ID on all
    architectures. However, on s390x, the amount of usable CPUs is determined
    during runtime - it is depending on the features of the machine the code
    is running on. Since we are using the vcpu_id as an index into the SCA
    structures that are defined by the hardware (see e.g. the sca_add_vcpu()
    function), it is not only the amount of CPUs that is limited by the hard-
    ware, but also the range of IDs that we can use.
    Thus KVM_CAP_MAX_VCPU_ID must be determined during runtime on s390x, too.
    So the handling of KVM_CAP_MAX_VCPU_ID has to be moved from the common
    code into the architecture specific code, and on s390x we have to return
    the same value here as for KVM_CAP_MAX_VCPUS.
    This problem has been discovered with the kvm_create_max_vcpus selftest.
    With this change applied, the selftest now passes on s390x, too.

    Reviewed-by: Andrew Jones <drjones@redhat.com>
    Reviewed-by: Cornelia Huck <cohuck@redhat.com>
    Reviewed-by: David Hildenbrand <david@redhat.com>
    Signed-off-by: Thomas Huth <thuth@redhat.com>
    Message-Id: <20190523164309.13345-9-thuth@redhat.com>
    Cc: stable@vger.kernel.org
    Signed-off-by: Christian Borntraeger <borntraeger@de.ibm.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 8f3b8eeaea4227d1335645b71187b78368424aae
Author: Hui Wang <hui.wang@canonical.com>
Date:   Wed May 29 12:41:38 2019 +0800

    ALSA: hda/realtek - Improve the headset mic for Acer Aspire laptops
```

```
    commit 9cb40eb184c4220d244a532bd940c6345ad9dbd9 upstream.

    We met another Acer Aspire laptop which has the problem on the
    headset-mic, the Pin 0x19 is not set the corret configuration for a
    mic and the pin presence can't be detected too after plugging a
    headset. Kailang suggested that we should set the coeff to enable the
    mic and apply the ALC269_FIXUP_LIFEBOOK_EXTMIC. After doing that,
    both headset-mic presence and headset-mic work well.

    The existing ALC255_FIXUP_ACER_MIC_NO_PRESENCE set the headset-mic
    jack to be a phantom jack. Now since the jack can support presence
    unsol event, let us imporve it to set the jack to be a normal jack.

    https://bugs.launchpad.net/bugs/1821269
    Fixes: 5824ce8de7b1c ("ALSA: hda/realtek - Add support for Acer Aspire E5-475 headset
mic")
    Cc: Chris Chiu <chiu@endlessm.com>
    CC: Daniel Drake <drake@endlessm.com>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Kailang Yang <kailang@realtek.com>
    Signed-off-by: Hui Wang <hui.wang@canonical.com>
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 0031bae95a86c369ff9b5b8f301f932d0c1b5fc3
Author: Kailang Yang <kailang@realtek.com>
Date:   Thu May 23 14:43:04 2019 +0800

    ALSA: hda/realtek - Set default power save node to 0

    commit 317d9313925cd8388304286c0d3c8dda7f060a2d upstream.

    I measured power consumption between power_save_node=1 and power_save_node=0.
    It's almost the same.
    Codec will enter to runtime suspend and suspend.
    That pin also will enter to D3. Don't need to enter to D3 by single pin.
    So, Disable power_save_node as default. It will avoid more issues.
    Windows Driver also has not this option at runtime PM.

    Signed-off-by: Kailang Yang <kailang@realtek.com>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit cf02204f3c7ee56bece26630d67805fcc0891874
Author: Takashi Iwai <tiwai@suse.de>
Date:   Tue May 28 08:39:44 2019 +0200

    ALSA: line6: Assure canceling delayed work at disconnection

    commit 0b074ab7fc0d575247b9cc9f93bb7e007ca38840 upstream.

    The current code performs the cancel of a delayed work at the late
    stage of disconnection procedure, which may lead to the access to the
    already cleared state.

    This patch assures to call cancel_delayed_work_sync() at the beginning
    of the disconnection procedure for avoiding that race.  The delayed
    work object is now assigned in the common line6 object instead of its
    derivative, so that we can call cancel_delayed_work_sync().

    Along with the change, the startup function is called via the new
    callback instead.  This will make it easier to port other LINE6
    drivers to use the delayed work for startup in later patches.
```

```
    Reported-by: syzbot+5255458d5e0a2b10bbb9@syzkaller.appspotmail.com
    Fixes: 7f84ff68be05 ("ALSA: line6: toneport: Fix broken usage of timer for delayed
execution")
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Takashi Iwai <tiwai@suse.de>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e1801d51d673e1c9bbe0ded03bc9285f6d5a858d
Author: Thiago Jung Bauermann <bauerman@linux.ibm.com>
Date:   Wed May 22 19:01:58 2019 -0300

    powerpc/kexec: Fix loading of kernel + initramfs with kexec_file_load()

    commit 8b909e3548706cbebc0a676067b81aadda57f47e upstream.

    Commit b6664ba42f14 ("s390, kexec_file: drop arch_kexec_mem_walk()")
    changed kexec_add_buffer() to skip searching for a memory location if
    kexec_buf.mem is already set, and use the address that is there.

    In powerpc code we reuse a kexec_buf variable for loading both the
    kernel and the initramfs by resetting some of the fields between those
    uses, but not mem. This causes kexec_add_buffer() to try to load the
    kernel at the same address where initramfs will be loaded, which is
    naturally rejected:

      # kexec -s -l --initrd initramfs vmlinuz
      kexec_file_load failed: Invalid argument

    Setting the mem field before every call to kexec_add_buffer() fixes
    this regression.

    Fixes: b6664ba42f14 ("s390, kexec_file: drop arch_kexec_mem_walk()")
    Cc: stable@vger.kernel.org # v5.0+
    Signed-off-by: Thiago Jung Bauermann <bauerman@linux.ibm.com>
    Reviewed-by: Dave Young <dyoung@redhat.com>
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 7229404941796558c387eee6f43989d750773cb2
Author: Ravi Bangoria <ravi.bangoria@linux.ibm.com>
Date:   Sat May 11 08:12:17 2019 +0530

    powerpc/perf: Fix MMCRA corruption by bhrb_filter

    commit 3202e35ec1c8fc19cea24253ff83edf702a60a02 upstream.

    Consider a scenario where user creates two events:

      1st event:
        attr.sample_type |= PERF_SAMPLE_BRANCH_STACK;
        attr.branch_sample_type = PERF_SAMPLE_BRANCH_ANY;
        fd = perf_event_open(attr, 0, 1, -1, 0);

      This sets cpuhw->bhrb_filter to 0 and returns valid fd.

      2nd event:
        attr.sample_type |= PERF_SAMPLE_BRANCH_STACK;
        attr.branch_sample_type = PERF_SAMPLE_BRANCH_CALL;
        fd = perf_event_open(attr, 0, 1, -1, 0);

      It overrides cpuhw->bhrb_filter to -1 and returns with error.

    Now if power_pmu_enable() gets called by any path other than
    power_pmu_add(), ppmu->config_bhrb(-1) will set MMCRA to -1.
```

```
    Fixes: 3925f46bb590 ("powerpc/perf: Enable branch stack sampling framework")
    Cc: stable@vger.kernel.org # v3.10+
    Signed-off-by: Ravi Bangoria <ravi.bangoria@linux.ibm.com>
    Reviewed-by: Madhavan Srinivasan <maddy@linux.vnet.ibm.com>
    Signed-off-by: Michael Ellerman <mpe@ellerman.id.au>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit effe3c9c8a466ccb6183c9db97a33a3716118659
Author: Suraj Jitindar Singh <sjitindarsingh@gmail.com>
Date:   Thu May 30 12:17:18 2019 +1000

    KVM: PPC: Book3S HV: Restore SPRG3 in kvmhv_p9_guest_entry()

    commit d724c9e54939a597592de3659541da11fc7aa112 upstream.

    The sprgs are a set of 4 general purpose sprs provided for software use.
    SPRG3 is special in that it can also be read from userspace. Thus it is
    used on linux to store the cpu and numa id of the process to speed up
    syscall access to this information.

    This register is overwritten with the guest value on kvm guest entry,
    and so needs to be restored on exit again. Thus restore the value on
    the guest exit path in kvmhv_p9_guest_entry().

    Cc: stable@vger.kernel.org # v4.20+
    Fixes: 95a6432ce9038 ("KVM: PPC: Book3S HV: Streamlined guest entry/exit path on P9 for
radix guests")

    Signed-off-by: Suraj Jitindar Singh <sjitindarsingh@gmail.com>
    Signed-off-by: Paul Mackerras <paulus@ozlabs.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b297e5b0eb613eb311fec3bb7c5b0de17c3f1818
Author: Paul Mackerras <paulus@ozlabs.org>
Date:   Tue May 28 15:01:59 2019 +1000

    KVM: PPC: Book3S HV: Fix lockdep warning when entering guest on POWER9

    commit 1b28d5531e446a87bbefa5ced191c4cbd316576c upstream.

    Commit 3309bec85e60 ("KVM: PPC: Book3S HV: Fix lockdep warning when
    entering the guest") moved calls to trace_hardirqs_{on,off} in the
    entry path used for HPT guests.  Similar code exists in the new
    streamlined entry path used for radix guests on POWER9.  This makes
    the same change there, so as to avoid lockdep warnings such as this:

    [  228.686461] DEBUG_LOCKS_WARN_ON(current->hardirqs_enabled)
    [  228.686480] WARNING: CPU: 116 PID: 3803 at ../kernel/locking/lockdep.c:4219
check_flags.part.23+0x21c/0x270
    [  228.686544] Modules linked in: vhost_net vhost xt_CHECKSUM iptable_mangle
xt_MASQUERADE iptable_nat nf_nat
    +xt_conntrack nf_conntrack nf_defrag_ipv6 nf_defrag_ipv4 ipt_REJECT nf_reject_ipv4 tun
bridge stp llc ebtable_filter
    +ebtables ip6table_filter ip6_tables iptable_filter fuse kvm_hv kvm at24 ipmi_powernv
regmap_i2c ipmi_devintf
    +uio_pdrv_genirq ofpart ipmi_msghandler uio powernv_flash mtd ibmpowernv opal_prd
ip_tables ext4 mbcache jbd2 btrfs
    +zstd_decompress zstd_compress raid10 raid456 async_raid6_recov async_memcpy async_pq
async_xor async_tx libcrc32c xor
    +raid6_pq raid1 raid0 ses sd_mod enclosure scsi_transport_sas ast i2c_opal i2c_algo_bit
drm_kms_helper syscopyarea
    +sysfillrect sysimgblt fb_sys_fops ttm drm i40e e1000e cxl aacraid tg3
drm_panel_orientation_quirks i2c_core
    [  228.686859] CPU: 116 PID: 3803 Comm: qemu-system-ppc Kdump: loaded Not tainted 5.2.0-
```

```
 rc1-xive+ #42
     [  228.686911] NIP:  c0000000001b394c LR: c0000000001b3948 CTR: c000000000bfad20
     [  228.686963] REGS: c000200cdb50f570 TRAP: 0700   Not tainted  (5.2.0-rc1-xive+)
     [  228.687001] MSR:  9000000002823033 <SF,HV,VEC,VSX,FP,ME,IR,DR,RI,LE>  CR: 48222222
 XER: 20040000
     [  228.687060] CFAR: c000000000116db0 IRQMASK: 1
     [  228.687060] GPR00: c0000000001b3948 c000200cdb50f800 c0000000015e7600
 000000000000002e
     [  228.687060] GPR04: 0000000000000001 c0000000001c71a0 000000006e655f73
 72727563284e4f5f
     [  228.687060] GPR08: 0000200e60680000 0000000000000000 c000200cdb486180
 0000000000000000
     [  228.687060] GPR12: 0000000000002000 c000200fff61a680 0000000000000000
 00007fffb75c0000
     [  228.687060] GPR16: 0000000000000000 0000000000000000 c0000000017d6900
 c000000001124900
     [  228.687060] GPR20: 0000000000000074 c008000006916f68 0000000000000074
 0000000000000074
     [  228.687060] GPR24: ffffffffffffffff ffffffffffffffff 0000000000000003
 c000200d4b600000
     [  228.687060] GPR28: c000000001627e58 c000000001489908 c000000001627e58
 c000000002304de0
     [  228.687377] NIP [c0000000001b394c] check_flags.part.23+0x21c/0x270
     [  228.687415] LR [c0000000001b3948] check_flags.part.23+0x218/0x270
     [  228.687466] Call Trace:
     [  228.687488] [c000200cdb50f800] [c0000000001b3948] check_flags.part.23+0x218/0x270
 (unreliable)
     [  228.687542] [c000200cdb50f870] [c0000000001b6548] lock_is_held_type+0x188/0x1c0
     [  228.687595] [c000200cdb50f8d0] [c0000000001d939c] rcu_read_lock_sched_held+0xdc/0x100
     [  228.687646] [c000200cdb50f900] [c0000000001dd704] rcu_note_context_switch+0x304/0x340
     [  228.687701] [c000200cdb50f940] [c0080000068fcc58] kvmhv_run_single_vcpu+0xdb0/0x1120
 [kvm_hv]
     [  228.687756] [c000200cdb50fa20] [c0080000068fd5b0] kvmppc_vcpu_run_hv+0x5e8/0xe40
 [kvm_hv]
     [  228.687816] [c000200cdb50faf0] [c0080000071797dc] kvmppc_vcpu_run+0x34/0x48 [kvm]
     [  228.687863] [c000200cdb50fb10] [c0080000071755dc] kvm_arch_vcpu_ioctl_run+0x244/0x420
 [kvm]
     [  228.687916] [c000200cdb50fba0] [c008000007165ccc] kvm_vcpu_ioctl+0x424/0x838 [kvm]
     [  228.687957] [c000200cdb50fd10] [c000000000433a24] do_vfs_ioctl+0xd4/0xcd0
     [  228.687995] [c000200cdb50fdb0] [c000000000434724] ksys_ioctl+0x104/0x120
     [  228.688033] [c000200cdb50fe00] [c000000000434768] sys_ioctl+0x28/0x80
     [  228.688072] [c000200cdb50fe20] [c00000000000b888] system_call+0x5c/0x70
     [  228.688109] Instruction dump:
     [  228.688142] 4bf6342d 60000000 0fe00000 e8010080 7c0803a6 4bfffe60 3c82ff87 3c62ff87
     [  228.688196] 388472d0 3863d738 4bf63405 60000000 <0fe00000> 4bffff4c 3c82ff87 3c62ff87
     [  228.688251] irq event stamp: 205
     [  228.688287] hardirqs last  enabled at (205): [<c0080000068fc1b4>]
 kvmhv_run_single_vcpu+0x30c/0x1120 [kvm_hv]
     [  228.688344] hardirqs last disabled at (204): [<c0080000068fbff0>]
 kvmhv_run_single_vcpu+0x148/0x1120 [kvm_hv]
     [  228.688412] softirqs last  enabled at (180): [<c000000000c0b2ac>]
 __do_softirq+0x4ac/0x5d4
     [  228.688464] softirqs last disabled at (169): [<c000000000122aa8>]
 irq_exit+0x1f8/0x210
     [  228.688513] ---[ end trace eb16f6260022a812 ]---
     [  228.688548] possible reason: unannotated irqs-off.
     [  228.688571] irq event stamp: 205
     [  228.688607] hardirqs last  enabled at (205): [<c0080000068fc1b4>]
 kvmhv_run_single_vcpu+0x30c/0x1120 [kvm_hv]
     [  228.688664] hardirqs last disabled at (204): [<c0080000068fbff0>]
 kvmhv_run_single_vcpu+0x148/0x1120 [kvm_hv]
     [  228.688719] softirqs last  enabled at (180): [<c000000000c0b2ac>]
 __do_softirq+0x4ac/0x5d4
     [  228.688758] softirqs last disabled at (169): [<c000000000122aa8>]
 irq_exit+0x1f8/0x210
```

```
    Cc: stable@vger.kernel.org # v4.20+
    Fixes: 95a6432ce903 ("KVM: PPC: Book3S HV: Streamlined guest entry/exit path on P9 for
radix guests")
    Signed-off-by: Paul Mackerras <paulus@ozlabs.org>
    Reviewed-by: Cédric Le Goater <clg@kaod.org>
    Tested-by: Cédric Le Goater <clg@kaod.org>
    Signed-off-by: Paul Mackerras <paulus@ozlabs.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit b625c1586efa5148ffd04b9b76c5a4137182af64
Author: Cédric Le Goater <clg@kaod.org>
Date:   Tue May 28 14:17:15 2019 +0200

    KVM: PPC: Book3S HV: XIVE: Do not clear IRQ data of passthrough interrupts

    commit ef9740204051d0e00f5402fe96cf3a43ddd2bbbf upstream.

    The passthrough interrupts are defined at the host level and their IRQ
    data should not be cleared unless specifically deconfigured (shutdown)
    by the host. They differ from the IPI interrupts which are allocated
    by the XIVE KVM device and reserved to the guest usage only.

    This fixes a host crash when destroying a VM in which a PCI adapter
    was passed-through. In this case, the interrupt is cleared and freed
    by the KVM device and then shutdown by vfio at the host level.

    [ 1007.360265] BUG: Kernel NULL pointer dereference at 0x00000d00
    [ 1007.360285] Faulting instruction address: 0xc00000000009da34
    [ 1007.360296] Oops: Kernel access of bad area, sig: 7 [#1]
    [ 1007.360303] LE PAGE_SIZE=64K MMU=Radix MMU=Hash SMP NR_CPUS=2048 NUMA PowerNV
    [ 1007.360314] Modules linked in: vhost_net vhost iptable_mangle ipt_MASQUERADE
iptable_nat nf_nat xt_conntrack nf_conntrack nf_defrag_ipv4 ipt_REJECT nf_reject_ipv4 tun
bridge stp llc kvm_hv kvm xt_tcpudp iptable_filter squashfs fuse binfmt_misc vmx_crypto
ib_iser rdma_cm iw_cm ib_cm libiscsi scsi_transport_iscsi nfsd ip_tables x_tables autofs4
btrfs zstd_decompress zstd_compress lzo_compress raid10 raid456 async_raid6_recov
async_memcpy async_pq async_xor async_tx xor raid6_pq multipath mlx5_ib ib_uverbs ib_core
crc32c_vpmsum mlx5_core
    [ 1007.360425] CPU: 9 PID: 15576 Comm: CPU 18/KVM Kdump: loaded Not tainted 5.1.0-
gad7e7d0ef #4
    [ 1007.360454] NIP:  c00000000009da34 LR: c00000000009e50c CTR: c00000000009e5d0
    [ 1007.360482] REGS: c000007f24ccf330 TRAP: 0300   Not tainted  (5.1.0-gad7e7d0ef)
    [ 1007.360500] MSR:  900000000280b033 <SF,HV,VEC,VSX,EE,FP,ME,IR,DR,RI,LE>  CR: 24002484
XER: 00000000
    [ 1007.360532] CFAR: c00000000009da10 DAR: 0000000000000d00 DSISR: 00080000 IRQMASK: 1
    [ 1007.360532] GPR00: c00000000009e62c c000007f24ccf5c0 c000000001510600
c000007fe7f947c0
    [ 1007.360532] GPR04: 0000000000000d00 0000000000000000 0000000000000000
c000005eff02d200
    [ 1007.360532] GPR08: 0000000000400000 0000000000000000 0000000000000000
fffffffffffffffd
    [ 1007.360532] GPR12: c00000000009e5d0 c000007fffff7b00 0000000000000031
000000012c345718
    [ 1007.360532] GPR16: 0000000000000000 0000000000000008 0000000000418004
0000000000040100
    [ 1007.360532] GPR20: 0000000000000000 0000000008430000 00000000003c0000
0000000000000027
    [ 1007.360532] GPR24: 00000000000000ff 0000000000000000 00000000000000ff
c000007faa90d98c
    [ 1007.360532] GPR28: c000007faa90da40 00000000000fe040 ffffffffffffffff
c000007fe7f947c0
    [ 1007.360689] NIP [c00000000009da34] xive_esb_read+0x34/0x120
    [ 1007.360706] LR [c00000000009e50c] xive_do_source_set_mask.part.0+0x2c/0x50
    [ 1007.360732] Call Trace:
    [ 1007.360738] [c000007f24ccf5c0] [c000000000a6383c] snooze_loop+0x15c/0x270
```

```
  (unreliable)
      [ 1007.360775] [c000007f24ccf5f0] [c00000000009e62c] xive_irq_shutdown+0x5c/0xe0
      [ 1007.360795] [c000007f24ccf630] [c00000000019e4a0] irq_shutdown+0x60/0xe0
      [ 1007.360813] [c000007f24ccf660] [c000000000198c44] __free_irq+0x3a4/0x420
      [ 1007.360831] [c000007f24ccf700] [c000000000198dc8] free_irq+0x78/0xe0
      [ 1007.360849] [c000007f24ccf730] [c00000000096c5a8]
  vfio_msi_set_vector_signal+0xa8/0x350
      [ 1007.360878] [c000007f24ccf7f0] [c00000000096c938] vfio_msi_set_block+0xe8/0x1e0
      [ 1007.360899] [c000007f24ccf850] [c00000000096cae0] vfio_msi_disable+0xb0/0x110
      [ 1007.360912] [c000007f24ccf8a0] [c00000000096cd04]
  vfio_pci_set_msi_trigger+0x1c4/0x3d0
      [ 1007.360922] [c000007f24ccf910] [c00000000096d910] vfio_pci_set_irqs_ioctl+0xa0/0x170
      [ 1007.360941] [c000007f24ccf930] [c00000000096b400] vfio_pci_disable+0x80/0x5e0
      [ 1007.360963] [c000007f24ccfa10] [c00000000096b9bc] vfio_pci_release+0x5c/0x90
      [ 1007.360991] [c000007f24ccfa40] [c000000000963a9c] vfio_device_fops_release+0x3c/0x70
      [ 1007.361012] [c000007f24ccfa70] [c0000000003b5668] __fput+0xc8/0x2b0
      [ 1007.361040] [c000007f24ccfac0] [c0000000001409b0] task_work_run+0x140/0x1b0
      [ 1007.361059] [c000007f24ccfb20] [c000000000118f8c] do_exit+0x3ac/0xd00
      [ 1007.361076] [c000007f24ccfc00] [c0000000001199b0] do_group_exit+0x60/0x100
      [ 1007.361094] [c000007f24ccfc40] [c00000000012b514] get_signal+0x1a4/0x8f0
      [ 1007.361112] [c000007f24ccfd30] [c000000000021cc8] do_notify_resume+0x1a8/0x430
      [ 1007.361141] [c000007f24ccfe20] [c00000000000e444] ret_from_except_lite+0x70/0x74
      [ 1007.361159] Instruction dump:
      [ 1007.361175] 38422c00 e9230000 712a0004 41820010 548a2036 7d442378 78840020 71290020
      [ 1007.361194] 4082004c e9230010 7c892214 7c0004ac <e9240000> 0c090000 4c00012c 792a0022

      Cc: stable@vger.kernel.org # v4.12+
      Fixes: 5af50993850a ("KVM: PPC: Book3S HV: Native usage of the XIVE interrupt
  controller")
      Signed-off-by: Cédric Le Goater <clg@kaod.org>
      Signed-off-by: Greg Kurz <groug@kaod.org>
      Signed-off-by: Paul Mackerras <paulus@ozlabs.org>
      Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

  commit c0fc8aff6c0884250c0c6622c76f5cb30d51cba7
  Author: Harald Freudenberger <freude@linux.ibm.com>
  Date:   Mon May 27 15:24:20 2019 +0200

      s390/crypto: fix possible sleep during spinlock aquired

      commit 1c2c7029c008922d4d48902cc386250502e73d51 upstream.

      This patch fixes a complain about possible sleep during
      spinlock aquired
      "BUG: sleeping function called from invalid context at
      include/crypto/algapi.h:426"
      for the ctr(aes) and ctr(des) s390 specific ciphers.

      Instead of using a spinlock this patch introduces a mutex
      which is save to be held in sleeping context. Please note
      a deadlock is not possible as mutex_trylock() is used.

      Signed-off-by: Harald Freudenberger <freude@linux.ibm.com>
      Reported-by: Julian Wiedmann <jwi@linux.ibm.com>
      Cc: stable@vger.kernel.org
      Signed-off-by: Heiko Carstens <heiko.carstens@de.ibm.com>
      Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

  commit 0de686f70de09738eaa3ae47100403f5c8f250f5
  Author: Harald Freudenberger <freude@linux.ibm.com>
  Date:   Thu May 23 16:18:25 2019 +0200

      s390/crypto: fix gcm-aes-s390 selftest failures

      commit bef9f0ba300a55d79a69aa172156072182176515 upstream.
```

```
    The current kernel uses improved crypto selftests. These
    tests showed that the current implementation of gcm-aes-s390
    is not able to deal with chunks of output buffers which are
    not a multiple of 16 bytes. This patch introduces a rework
    of the gcm aes s390 scatter walk handling which now is able
    to handle any input and output scatter list chunk sizes
    correctly.

    Code has been verified by the crypto selftests, the tcrypt
    kernel module and additional tests ran via the af_alg interface.

    Cc: <stable@vger.kernel.org>
    Reported-by: Julian Wiedmann <jwi@linux.ibm.com>
    Reviewed-by: Patrick Steuer <steuer@linux.ibm.com>
    Signed-off-by: Harald Freudenberger <freude@linux.ibm.com>
    Signed-off-by: Heiko Carstens <heiko.carstens@de.ibm.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 273ec0ffa4633aebda963daadd99b387dd5fff9c
Author: Sean Nyekjaer <sean@geanix.com>
Date:   Tue May 7 10:23:04 2019 +0200

    iio: adc: ti-ads8688: fix timestamp is not updated in buffer

    commit e6d12298310fa1dc11f1d747e05b168016057fdd upstream.

    When using the hrtimer iio trigger timestamp isn't updated.
    If we use iio_get_time_ns it is updated correctly.

    Fixes: 2a86487786b5c ("iio: adc: ti-ads8688: add trigger and buffer support")
    Signed-off-by: Sean Nyekjaer <sean@geanix.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 fb304841e95845ad131b6762ed2231c4302f8743
Author: Tomer Maimon <tmaimon77@gmail.com>
Date:   Sun Apr 7 11:19:28 2019 +0300

    iio: adc: modify NPCM ADC read reference voltage

    commit 4e63ed6b90803eeb400c392e9ff493200d926b06 upstream.

    Checking if regulator is valid before reading
    NPCM ADC regulator voltage to avoid system crash
    in a case the regulator is not valid.

    Signed-off-by: Tomer Maimon <tmaimon77@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 6b84e38b700bf6c8f4e983ba1e6c01d8b16bf581
Author: Vincent Stehlé <vincent.stehle@laposte.net>
Date:   Sun Mar 31 20:54:23 2019 +0200

    iio: adc: ads124: avoid buffer overflow

    commit 0db8aa49a97e7f40852a64fd35abcc1292a7c365 upstream.

    When initializing the priv->data array starting from index 1, there is one
    less element to consider than when initializing the full array.

    Fixes: e717f8c6dfec8f76 ("iio: adc: Add the TI ads124s08 ADC code")
```

```
    Signed-off-by: Vincent Stehlé <vincent.stehle@laposte.net>
    Reviewed-by: Mukesh Ojha <mojha@codeaurora.org>
    Reviewed-by: Dan Murphy <dmurphy@ti.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 642ec93fe232028f859a090e8a1d8dc4e4367d39
Author: Ruslan Babayev <ruslan@babayev.com>
Date:   Sun May 5 12:24:37 2019 -0700

    iio: dac: ds4422/ds4424 fix chip verification

    commit 60f2208699ec08ff9fdf1f97639a661a92a18f1c upstream.

    The ds4424_get_value function takes channel number as it's 3rd
    argument and translates it internally into I2C address using
    DS4424_DAC_ADDR macro. The caller ds4424_verify_chip was passing an
    already translated I2C address as its last argument.

    Signed-off-by: Ruslan Babayev <ruslan@babayev.com>
    Cc: xe-linux-external@cisco.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 19b65aac22fbcd20d0fd3b178cdf287c6dcf7fdc
Author: Qu Wenruo <wqu@suse.com>
Date:   Wed May 22 16:33:11 2019 +0800

    btrfs: reloc: Also queue orphan reloc tree for cleanup to avoid BUG_ON()

    commit 30d40577e322b670551ad7e2faa9570b6e23eb2b upstream.

    [BUG]
    When a fs has orphan reloc tree along with unfinished balance:
      ...
            item 16 key (TREE_RELOC ROOT_ITEM FS_TREE) itemoff 12090 itemsize 439
                    generation 12 root_dirid 256 bytenr 300400640 level 1 refs 0 <<<
                    lastsnap 8 byte_limit 0 bytes_used 1359872 flags 0x0(none)
                    uuid 7c48d938-33a3-4aae-ab19-6e5c9d406e46
            item 17 key (BALANCE TEMPORARY_ITEM 0) itemoff 11642 itemsize 448
                    temporary item objectid BALANCE offset 0
                    balance status flags 14

    Then at mount time, we can hit the following kernel BUG_ON():
      BTRFS info (device dm-3): relocating block group 298844160 flags metadata|dup
      ------------[ cut here ]------------
      kernel BUG at fs/btrfs/relocation.c:1413!
      invalid opcode: 0000 [#1] PREEMPT SMP NOPTI
      CPU: 1 PID: 897 Comm: btrfs-balance Tainted: G           O      5.2.0-rc1-custom #15
      RIP: 0010:create_reloc_root+0x1eb/0x200 [btrfs]
      Call Trace:
       btrfs_init_reloc_root+0x96/0xb0 [btrfs]
       record_root_in_trans+0xb2/0xe0 [btrfs]
       btrfs_record_root_in_trans+0x55/0x70 [btrfs]
       select_reloc_root+0x7e/0x230 [btrfs]
       do_relocation+0xc4/0x620 [btrfs]
       relocate_tree_blocks+0x592/0x6a0 [btrfs]
       relocate_block_group+0x47b/0x5d0 [btrfs]
       btrfs_relocate_block_group+0x183/0x2f0 [btrfs]
       btrfs_relocate_chunk+0x4e/0xe0 [btrfs]
       btrfs_balance+0x864/0xfa0 [btrfs]
       balance_kthread+0x3b/0x50 [btrfs]
       kthread+0x123/0x140
```

```
        ret_from_fork+0x27/0x50

    [CAUSE]
    In btrfs, reloc trees are used to record swapped tree blocks during
    balance.
    Reloc tree either get merged (replace old tree blocks of its parent
    subvolume) in next transaction if its ref is 1 (fresh).
    Or is already merged and will be cleaned up if its ref is 0 (orphan).

    After commit d2311e698578 ("btrfs: relocation: Delay reloc tree deletion
    after merge_reloc_roots"), reloc tree cleanup is delayed until one block
    group is balanced.

    Since fresh reloc roots are recorded during merge, as long as there
    is no power loss, those orphan reloc roots converted from fresh ones are
    handled without problem.

    However when power loss happens, orphan reloc roots can be recorded
    on-disk, thus at next mount time, we will have orphan reloc roots from
    on-disk data directly, and ignored by clean_dirty_subvols() routine.

    Then when background balance starts to balance another block group, and
    needs to create new reloc root for the same root, btrfs_insert_item()
    returns -EEXIST, and trigger that BUG_ON().

    [FIX]
    For orphan reloc roots, also queue them to rc->dirty_subvol_roots, so
    all reloc roots no matter orphan or not, can be cleaned up properly and
    avoid above BUG_ON().

    And to cooperate with above change, clean_dirty_subvols() will check if
    the queued root is a reloc root or a subvol root.
    For a subvol root, do the old work, and for a orphan reloc root, clean it
    up.

    Fixes: d2311e698578 ("btrfs: relocation: Delay reloc tree deletion after
merge_reloc_roots")
    CC: stable@vger.kernel.org # 5.1
    Signed-off-by: Qu Wenruo <wqu@suse.com>
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 8a7c6f193db66ea0e2b79f76e4b64dd251859fe1
Author: Filipe Manana <fdmanana@suse.com>
Date:   Mon May 20 09:55:42 2019 +0100

    Btrfs: incremental send, fix file corruption when no-holes feature is enabled

    commit 6b1f72e5b82a5c2a4da4d1ebb8cc01913ddbea21 upstream.

    When using the no-holes feature, if we have a file with prealloc extents
    with a start offset beyond the file's eof, doing an incremental send can
    cause corruption of the file due to incorrect hole detection. Such case
    requires that the prealloc extent(s) exist in both the parent and send
    snapshots, and that a hole is punched into the file that covers all its
    extents that do not cross the eof boundary.

    Example reproducer:

      $ mkfs.btrfs -f -O no-holes /dev/sdb
      $ mount /dev/sdb /mnt/sdb

      $ xfs_io -f -c "pwrite -S 0xab 0 500K" /mnt/sdb/foobar
      $ xfs_io -c "falloc -k 1200K 800K" /mnt/sdb/foobar
```

```
        $ btrfs subvolume snapshot -r /mnt/sdb /mnt/sdb/base

        $ btrfs send -f /tmp/base.snap /mnt/sdb/base

        $ xfs_io -c "fpunch 0 500K" /mnt/sdb/foobar

        $ btrfs subvolume snapshot -r /mnt/sdb /mnt/sdb/incr

        $ btrfs send -p /mnt/sdb/base -f /tmp/incr.snap /mnt/sdb/incr

        $ md5sum /mnt/sdb/incr/foobar
        816df6f64deba63b029ca19d880ee10a   /mnt/sdb/incr/foobar

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

        $ btrfs receive -f /tmp/base.snap /mnt/sdc
        $ btrfs receive -f /tmp/incr.snap /mnt/sdc

        $ md5sum /mnt/sdc/incr/foobar
        cf2ef71f4a9e90c2f6013ba3b2257ed2   /mnt/sdc/incr/foobar

          --> Different checksum, because the prealloc extent beyond the
              file's eof confused the hole detection code and it assumed
              a hole starting at offset 0 and ending at the offset of the
              prealloc extent (1200Kb) instead of ending at the offset
              500Kb (the file's size).

    Fix this by ensuring we never cross the file's size when issuing the
    write operations for a hole.

    Fixes: 16e7549f045d33 ("Btrfs: incompatible format change to remove hole extents")
    CC: stable@vger.kernel.org # 3.14+
    Signed-off-by: Filipe Manana <fdmanana@suse.com>
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit afd722785f8ba82e5e280e12dddaf210575ba356
Author: Qu Wenruo <wqu@suse.com>
Date:   Tue May 21 19:28:08 2019 +0800

    btrfs: qgroup: Check bg while resuming relocation to avoid NULL pointer dereference

    commit 57949d033a09c57d77be218b5bec07af6878ab32 upstream.

    [BUG]
    When mounting a fs with reloc tree and has qgroup enabled, it can cause
    NULL pointer dereference at mount time:

      BUG: kernel NULL pointer dereference, address: 00000000000000a8
      #PF: supervisor read access in kernel mode
      #PF: error_code(0x0000) - not-present page
      PGD 0 P4D 0
      Oops: 0000 [#1] PREEMPT SMP NOPTI
      RIP: 0010:btrfs_qgroup_add_swapped_blocks+0x186/0x300 [btrfs]
      Call Trace:
       replace_path.isra.23+0x685/0x900 [btrfs]
       merge_reloc_root+0x26e/0x5f0 [btrfs]
       merge_reloc_roots+0x10a/0x1a0 [btrfs]
       btrfs_recover_relocation+0x3cd/0x420 [btrfs]
       open_ctree+0x1bc8/0x1ed0 [btrfs]
       btrfs_mount_root+0x544/0x680 [btrfs]
       legacy_get_tree+0x34/0x60
       vfs_get_tree+0x2d/0xf0
       fc_mount+0x12/0x40
```

```
        vfs_kern_mount.part.12+0x61/0xa0
        vfs_kern_mount+0x13/0x20
        btrfs_mount+0x16f/0x860 [btrfs]
        legacy_get_tree+0x34/0x60
        vfs_get_tree+0x2d/0xf0
        do_mount+0x81f/0xac0
        ksys_mount+0xbf/0xe0
        __x64_sys_mount+0x25/0x30
        do_syscall_64+0x65/0x240
        entry_SYSCALL_64_after_hwframe+0x49/0xbe

    [CAUSE]
    In btrfs_recover_relocation(), we don't have enough info to determine
    which block group we're relocating, but only to merge existing reloc
    trees.

    Thus in btrfs_recover_relocation(), rc->block_group is NULL.
    btrfs_qgroup_add_swapped_blocks() hasn't taken this into consideration,
    and causes a NULL pointer dereference.

    The bug is introduced by commit 3d0174f78e72 ("btrfs: qgroup: Only trace
    data extents in leaves if we're relocating data block group"), and
    later qgroup refactoring still keeps this optimization.

    [FIX]
    Thankfully in the context of btrfs_recover_relocation(), there is no
    other progress can modify tree blocks, thus those swapped tree blocks
    pair will never affect qgroup numbers, no matter whatever we set for
    block->trace_leaf.

    So we only need to check if @bg is NULL before accessing @bg->flags.

    Reported-by: Juan Erbes <jerbes@gmail.com>
    Link: https://bugzilla.opensuse.org/show_bug.cgi?id=1134806
    Fixes: 3d0174f78e72 ("btrfs: qgroup: Only trace data extents in leaves if we're
relocating data block group")
    CC: stable@vger.kernel.org # 4.20+
    Signed-off-by: Qu Wenruo <wqu@suse.com>
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit c2d03b614443dc864c4ce6a2dad0f50f0505726f
Author: Dennis Zhou <dennis@kernel.org>
Date:   Fri May 17 19:16:26 2019 -0400

    btrfs: correct zstd workspace manager lock to use spin_lock_bh()

    commit fee13fe96529523a709d1fff487f14a5e0d56d34 upstream.

    The btrfs zstd workspace manager uses a background timer to reclaim not
    recently used workspaces. I used spin_lock() from this context which
    should have been caught with lockdep, but was not. This deadlock was
    reported in bugzilla. The fix is to switch the zstd wsm lock to use
    spin_lock_bh() from the softirq context.

    This happened quite relibably on ppc64, unlike on other architectures.

      [  313.402874] ================================
      [  313.402875] WARNING: inconsistent lock state
      [  313.402879] 5.1.0-rc7 #1 Not tainted
      [  313.402880] --------------------------------
      [  313.402882] inconsistent {SOFTIRQ-ON-W} -> {IN-SOFTIRQ-W} usage.
      [  313.402885] swapper/5/0 [HC0[0]:SC1[1]:HE1:SE0] takes:
      [  313.402888] 0000000080d1120c (&(&wsm.lock)->rlock){+.?.}, at:
.zstd_reclaim_timer_fn+0x40/0x230
```

```
       [  313.402895] {SOFTIRQ-ON-W} state was registered at:
       [  313.402899]   .lock_acquire+0xd0/0x240
       [  313.402903]   ._raw_spin_lock+0x34/0x60
       [  313.402906]   .zstd_get_workspace+0xd0/0x360
       [  313.402908]   .end_compressed_bio_read+0x3b8/0x540
       [  313.402911]   .bio_endio+0x174/0x2c0
       [  313.402914]   .end_workqueue_fn+0x4c/0x70
       [  313.402917]   .normal_work_helper+0x138/0x7e0
       [  313.402920]   .process_one_work+0x324/0x790
       [  313.402922]   .worker_thread+0x68/0x570
       [  313.402925]   .kthread+0x19c/0x1b0
       [  313.402928]   .ret_from_kernel_thread+0x58/0x78
       [  313.402930] irq event stamp: 2629216
       [  313.402933] hardirqs last  enabled at (2629216): [<c0000000009da738>]
  ._raw_spin_unlock_irq+0x38/0x60
       [  313.402936] hardirqs last disabled at (2629215): [<c0000000009da4c4>]
  ._raw_spin_lock_irq+0x24/0x70
       [  313.402939] softirqs last  enabled at (2629212): [<c0000000000af9fc>]
  .irq_enter+0x8c/0xd0
       [  313.402942] softirqs last disabled at (2629213): [<c0000000000afb58>]
  .irq_exit+0x118/0x170
       [  313.402944]
                      other info that might help us debug this:
       [  313.402945]  Possible unsafe locking scenario:

       [  313.402947]        CPU0
       [  313.402948]        ----
       [  313.402949]   lock(&(&wsm.lock)->rlock);
       [  313.402951]   <Interrupt>
       [  313.402952]     lock(&(&wsm.lock)->rlock);
       [  313.402954]
                       *** DEADLOCK ***

       [  313.402957] 1 lock held by swapper/5/0:
       [  313.402958]  #0: 000000004b612042 ((&wsm.timer)){+.-.}, at:
  .call_timer_fn+0x0/0x3c0
       [  313.402963]
                      stack backtrace:
       [  313.402967] CPU: 5 PID: 0 Comm: swapper/5 Not tainted 5.1.0-rc7 #1
       [  313.402968] Call Trace:
       [  313.402972] [c0000007fa262e70] [c0000000009b3294] .dump_stack+0xe0/0x15c
  (unreliable)
       [  313.402975] [c0000007fa262f10] [c000000000125548] .print_usage_bug+0x348/0x390
       [  313.402978] [c0000007fa262fd0] [c000000000125cb4] .mark_lock+0x724/0x930
       [  313.402981] [c0000007fa263080] [c000000000126c20] .__lock_acquire+0xc90/0x16a0
       [  313.402984] [c0000007fa2631b0] [c000000000128040] .lock_acquire+0xd0/0x240
       [  313.402987] [c0000007fa263280] [c0000000009da2b4] ._raw_spin_lock+0x34/0x60
       [  313.402990] [c0000007fa263300] [c00000000054b0b0] .zstd_reclaim_timer_fn+0x40/0x230
       [  313.402993] [c0000007fa2633d0] [c000000000158b38] .call_timer_fn+0xc8/0x3c0
       [  313.402996] [c0000007fa2634a0] [c000000000158f74] .expire_timers+0x144/0x260
       [  313.402999] [c0000007fa263550] [c000000000159178] .run_timer_softirq+0xe8/0x230
       [  313.403002] [c0000007fa263680] [c0000000009db288] .__do_softirq+0x188/0x5d4
       [  313.403004] [c0000007fa263790] [c0000000000afb58] .irq_exit+0x118/0x170
       [  313.403008] [c0000007fa263800] [c000000000028d88] .timer_interrupt+0x158/0x430
       [  313.403012] [c0000007fa2638b0] [c0000000000091d4] decrementer_common+0x134/0x140
       [  313.403017] --- interrupt: 901 at replay_interrupt_return+0x0/0x4
                          LR = .arch_local_irq_restore.part.0+0x68/0x80
       [  313.403020] [c0000007fa263bb0] [c00000000001a3ac]
  .arch_local_irq_restore.part.0+0x2c/0x80 (unreliable)
       [  313.403024] [c0000007fa263c30] [c0000000007bbbcc] .cpuidle_enter_state+0xec/0x670
       [  313.403027] [c0000007fa263d00] [c0000000000f5130] .call_cpuidle+0x40/0x90
       [  313.403031] [c0000007fa263d70] [c0000000000f554c] .do_idle+0x2dc/0x3a0
       [  313.403034] [c0000007fa263e30] [c0000000000f59ac] .cpu_startup_entry+0x2c/0x30
       [  313.403037] [c0000007fa263ea0] [c000000000045674] .start_secondary+0x644/0x650
       [  313.403041] [c0000007fa263f90] [c00000000000ad5c] start_secondary_prolog+0x10/0x14
```

```
    Bugzilla: https://bugzilla.kernel.org/show_bug.cgi?id=203517
    Fixes: 3f93aef535c8 ("btrfs: add zstd compression level support")
    CC: stable@vger.kernel.org # 5.1+
    Signed-off-by: Dennis Zhou <dennis@kernel.org>
    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 a1d13d873782b9ac41febecdf241d34cd5a62d70
Author: Filipe Manana <fdmanana@suse.com>
Date:   Thu May 16 15:48:55 2019 +0100

    Btrfs: fix fsync not persisting changed attributes of a directory

    commit 60d9f50308e5df19bc18c2fefab0eba4a843900a upstream.

    While logging an inode we follow its ancestors and for each one we mark
    it as logged in the current transaction, even if we have not logged it.
    As a consequence if we change an attribute of an ancestor, such as the
    UID or GID for example, and then explicitly fsync it, we end up not
    logging the inode at all despite returning success to user space, which
    results in the attribute being lost if a power failure happens after
    the fsync.

    Sample reproducer:

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

      $ mkdir /mnt/dir
      $ chown 6007:6007 /mnt/dir

      $ sync

      $ chown 9003:9003 /mnt/dir
      $ touch /mnt/dir/file
      $ xfs_io -c fsync /mnt/dir/file

      # fsync our directory after fsync'ing the new file, should persist the
      # new values for the uid and gid.
      $ xfs_io -c fsync /mnt/dir

      <power failure>

      $ mount /dev/sdb /mnt
      $ stat -c %u:%g /mnt/dir
      6007:6007

        --> should be 9003:9003, the uid and gid were not persisted, despite
            the explicit fsync on the directory prior to the power failure

    Fix this by not updating the logged_trans field of ancestor inodes when
    logging an inode, since we have not logged them. Let only future calls to
    btrfs_log_inode() to mark inodes as logged.

    This could be triggered by my recent fsync fuzz tester for fstests, for
    which an fstests patch exists titled "fstests: generic, fsync fuzz tester
    with fsstress".

    Fixes: 12fcfd22fe5b ("Btrfs: tree logging unlink/rename fixes")
    CC: stable@vger.kernel.org # 4.4+
    Signed-off-by: Filipe Manana <fdmanana@suse.com>
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

```
commit 1e6d76d45e36ca328ed69ceb1e8cdaece45cf304
Author: Filipe Manana <fdmanana@suse.com>
Date:   Wed May 15 16:03:17 2019 +0100

    Btrfs: fix race updating log root item during fsync

    commit 06989c799f04810f6876900d4760c0edda369cf7 upstream.

    When syncing the log, the final phase of a fsync operation, we need to
    either create a log root's item or update the existing item in the log
    tree of log roots, and that depends on the current value of the log
    root's log_transid - if it's 1 we need to create the log root item,
    otherwise it must exist already and we update it. Since there is no
    synchronization between updating the log_transid and checking it for
    deciding whether the log root's item needs to be created or updated, we
    end up with a tiny race window that results in attempts to update the
    item to fail because the item was not yet created:

                  CPU 1                                         CPU 2

      btrfs_sync_log()

        lock root->log_mutex

        set log root's log_transid to 1

        unlock root->log_mutex

                                                    btrfs_sync_log()

                                                      lock root->log_mutex

                                                      sets log root's
                                                      log_transid to 2

                                                      unlock root->log_mutex

        update_log_root()

          sees log root's log_transid
          with a value of 2

            calls btrfs_update_root(),
            which fails with -EUCLEAN
            and causes transaction abort

    Until recently the race lead to a BUG_ON at btrfs_update_root(), but after
    the recent commit 7ac1e464c4d47 ("btrfs: Don't panic when we can't find a
    root key") we just abort the current transaction.

    A sample trace of the BUG_ON() on a SLE12 kernel:

      ------------[ cut here ]------------
      kernel BUG at ../fs/btrfs/root-tree.c:157!
      Oops: Exception in kernel mode, sig: 5 [#1]
      SMP NR_CPUS=2048 NUMA pSeries
      (...)
      Supported: Yes, External
      CPU: 78 PID: 76303 Comm: rtas_errd Tainted: G                X 4.4.156-94.57-default
#1
      task: c00000ffa906d010 ti: c00000ff42b08000 task.ti: c00000ff42b08000
      NIP: d000000036ae5cdc LR: d000000036ae5cd8 CTR: 0000000000000000
      REGS: c00000ff42b0b860 TRAP: 0700   Tainted: G                X  (4.4.156-94.57-
default)
```

```
       MSR: 8000000002029033 <SF,VEC,EE,ME,IR,DR,RI,LE>  CR: 22444484  XER: 20000000
       CFAR: d000000036aba66c SOFTE: 1
       GPR00: d000000036ae5cd8 c00000ff42b0bae0 d000000036bda220 0000000000000054
       GPR04: 0000000000000001 0000000000000000 c00007ffff8d37c8 0000000000000000
       GPR08: c000000000e19c00 0000000000000000 0000000000000000 3736343438312079
       GPR12: 3930373337303434 c000000007a3a800 00000000007fffff 0000000000000023
       GPR16: c00000ffa9d26028 c00000ffa9d261f8 0000000000000010 c00000ffa9d2ab28
       GPR20: c00000ff42b0bc48 0000000000000001 c00000ff9f0d9888 0000000000000001
       GPR24: c00000ffa9d26000 c00000ffa9d261e8 c00000ffa9d2a800 c00000ff9f0d9888
       GPR28: c00000ffa9d26028 c00000ffa9d2aa98 0000000000000001 c00000ffa98f5b20
       NIP [d000000036ae5cdc] btrfs_update_root+0x25c/0x4e0 [btrfs]
       LR [d000000036ae5cd8] btrfs_update_root+0x258/0x4e0 [btrfs]
       Call Trace:
       [c00000ff42b0bae0] [d000000036ae5cd8] btrfs_update_root+0x258/0x4e0 [btrfs]
 (unreliable)
       [c00000ff42b0bba0] [d000000036b53610] btrfs_sync_log+0x2d0/0xc60 [btrfs]
       [c00000ff42b0bce0] [d000000036b1785c] btrfs_sync_file+0x44c/0x4e0 [btrfs]
       [c00000ff42b0bd80] [c00000000032e300] vfs_fsync_range+0x70/0x120
       [c00000ff42b0bdd0] [c00000000032e44c] do_fsync+0x5c/0xb0
       [c00000ff42b0be10] [c00000000032e8dc] SyS_fdatasync+0x2c/0x40
       [c00000ff42b0be30] [c000000000009488] system_call+0x3c/0x100
       Instruction dump:
       7f43d378 4bffebb9 60000000 88d90008 3d220000 e8b90000 3b390009 e87a01f0
       e8898e08 e8f90000 4bfd48e5 60000000 <0fe00000> e95b0060 39200004 394a0ea0
       ---[ end trace 8f2dc8f919cabab8 ]---

    So fix this by doing the check of log_transid and updating or creating the
    log root's item while holding the root's log_mutex.

    Fixes: 7237f1833601d ("Btrfs: fix tree logs parallel sync")
    CC: stable@vger.kernel.org # 4.4+
    Signed-off-by: Filipe Manana <fdmanana@suse.com>
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a48f8ac4efcd85fdc6038fa4efda4a0f52aafac3
Author: Filipe Manana <fdmanana@suse.com>
Date:   Wed May 15 16:02:47 2019 +0100

    Btrfs: fix wrong ctime and mtime of a directory after log replay

    commit 5338e43abbab13791144d37fd8846847062351c6 upstream.

    When replaying a log that contains a new file or directory name that needs
    to be added to its parent directory, we end up updating the mtime and the
    ctime of the parent directory to the current time after we have set their
    values to the correct ones (set at fsync time), efectivelly losing them.

    Sample reproducer:

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

      $ mkdir /mnt/dir
      $ touch /mnt/dir/file

      # fsync of the directory is optional, not needed
      $ xfs_io -c fsync /mnt/dir
      $ xfs_io -c fsync /mnt/dir/file

      $ stat -c %Y /mnt/dir
      1557856079

      <power failure>
```

```
       $ sleep 3
       $ mount /dev/sdb /mnt
       $ stat -c %Y /mnt/dir
       1557856082

         --> should have been 1557856079, the mtime is updated to the current
             time when replaying the log

    Fix this by not updating the mtime and ctime to the current time at
    btrfs_add_link() when we are replaying a log tree.

    This could be triggered by my recent fsync fuzz tester for fstests, for
    which an fstests patch exists titled "fstests: generic, fsync fuzz tester
    with fsstress".

    Fixes: e02119d5a7b43 ("Btrfs: Add a write ahead tree log to optimize synchronous
operations")
    CC: stable@vger.kernel.org # 4.4+
    Reviewed-by: Nikolay Borisov <nborisov@suse.com>
    Signed-off-by: Filipe Manana <fdmanana@suse.com>
    Signed-off-by: David Sterba <dsterba@suse.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 58abecd5a7da5eddbf261c98cd5977dd2235a695
Author: Tomas Bortoli <tomasbortoli@gmail.com>
Date:   Tue May 28 17:43:38 2019 +0200

    tracing: Avoid memory leak in predicate_parse()

    commit dfb4a6f2191a80c8b790117d0ff592fd712d3296 upstream.

    In case of errors, predicate_parse() goes to the out_free label
    to free memory and to return an error code.

    However, predicate_parse() does not free the predicates of the
    temporary prog_stack array, thence leaking them.

    Link: http://lkml.kernel.org/r/20190528154338.29976-1-tomasbortoli@gmail.com

    Cc: stable@vger.kernel.org
    Fixes: 80765597bc587 ("tracing: Rewrite filter logic to be simpler and faster")
    Reported-by: syzbot+6b8e0fb820e570c59e19@syzkaller.appspotmail.com
    Signed-off-by: Tomas Bortoli <tomasbortoli@gmail.com>
    [ Added protection around freeing prog_stack[i].pred ]
    Signed-off-by: Steven Rostedt (VMware) <rostedt@goodmis.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e0c4ec193040fc696269468aa2cb531a7c23558d
Author: Steffen Maier <maier@linux.ibm.com>
Date:   Thu May 23 15:23:46 2019 +0200

    scsi: zfcp: fix to prevent port_remove with pure auto scan LUNs (only sdevs)

    commit ef4021fe5fd77ced0323cede27979d80a56211ca upstream.

    When the user tries to remove a zfcp port via sysfs, we only rejected it if
    there are zfcp unit children under the port. With purely automatically
    scanned LUNs there are no zfcp units but only SCSI devices. In such cases,
    the port_remove erroneously continued. We close the port and this
    implicitly closes all LUNs under the port. The SCSI devices survive with
    their private zfcp_scsi_dev still holding a reference to the "removed"
    zfcp_port (still allocated but invisible in sysfs) [zfcp_get_port_by_wwpn
    in zfcp_scsi_slave_alloc]. This is not a problem as long as the fc_rport
    stays blocked. Once (auto) port scan brings back the removed port, we
    unblock its fc_rport again by design.  However, there is no mechanism that
```

```
    would recover (open) the LUNs under the port (no "ersfs_3" without
    zfcp_unit [zfcp_erp_strategy_followup_success]).  Any pending or new I/O to
    such LUN leads to repeated:

      Done: NEEDS_RETRY Result: hostbyte=DID_IMM_RETRY driverbyte=DRIVER_OK

    See also v4.10 commit 6f2ce1c6af37 ("scsi: zfcp: fix rport unblock race
    with LUN recovery"). Even a manual LUN recovery
    (echo 0 > /sys/bus/scsi/devices/H:C:T:L/zfcp_failed)
    does not help, as the LUN links to the old "removed" port which remains
    to lack ZFCP_STATUS_COMMON_RUNNING [zfcp_erp_required_act].
    The only workaround is to first ensure that the fc_rport is blocked
    (e.g. port_remove again in case it was re-discovered by (auto) port scan),
    then delete the SCSI devices, and finally re-discover by (auto) port scan.
    The port scan includes an fc_rport unblock, which in turn triggers
    a new scan on the scsi target to freshly get new pure auto scan LUNs.

    Fix this by rejecting port_remove also if there are SCSI devices
    (even without any zfcp_unit) under this port. Re-use mechanics from v3.7
    commit d99b601b6338 ("[SCSI] zfcp: restore refcount check on port_remove").
    However, we have to give up zfcp_sysfs_port_units_mutex earlier in unit_add
    to prevent a deadlock with scsi_host scan taking shost->scan_mutex first
    and then zfcp_sysfs_port_units_mutex now in our zfcp_scsi_slave_alloc().

    Signed-off-by: Steffen Maier <maier@linux.ibm.com>
    Fixes: b62a8d9b45b9 ("[SCSI] zfcp: Use SCSI device data zfcp scsi dev instead of zfcp
unit")
    Fixes: f8210e34887e ("[SCSI] zfcp: Allow midlayer to scan for LUNs when running in NPIV
mode")
    Cc: <stable@vger.kernel.org> #2.6.37+
    Reviewed-by: Benjamin Block <bblock@linux.ibm.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 66ba9b51c8948d87b1ef7603a4cc4746043eb8be
Author: Steffen Maier <maier@linux.ibm.com>
Date:   Thu May 23 15:23:45 2019 +0200

    scsi: zfcp: fix missing zfcp_port reference put on -EBUSY from port_remove

    commit d27e5e07f9c49bf2a6a4ef254ce531c1b4fb5a38 upstream.

    With this early return due to zfcp_unit child(ren), we don't use the
    zfcp_port reference from the earlier zfcp_get_port_by_wwpn() anymore and
    need to put it.

    Signed-off-by: Steffen Maier <maier@linux.ibm.com>
    Fixes: d99b601b6338 ("[SCSI] zfcp: restore refcount check on port_remove")
    Cc: <stable@vger.kernel.org> #3.7+
    Reviewed-by: Jens Remus <jremus@linux.ibm.com>
    Reviewed-by: Benjamin Block <bblock@linux.ibm.com>
    Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 6da1e14bc47f1cec4af12ad88000baaab7c2018e
Author: Piotr Figiel <p.figiel@camlintechnologies.com>
Date:   Mon Mar 4 15:42:52 2019 +0000

    brcmfmac: fix NULL pointer derefence during USB disconnect

    commit 5cdb0ef6144f47440850553579aa923c20a63f23 upstream.

    In case USB disconnect happens at the moment transmitting workqueue is in
    progress the underlying interface may be gone causing a NULL pointer
    dereference. Add synchronization of the workqueue destruction with the
```

```
    detach implementation in core so that the transmitting workqueue is stopped
    during detach before the interfaces are removed.

    Fix following Oops:

    Unable to handle kernel NULL pointer dereference at virtual address 00000008
    pgd = 9e6a802d
    [00000008] *pgd=00000000
    Internal error: Oops: 5 [#1] PREEMPT SMP ARM
    Modules linked in: nf_log_ipv4 nf_log_common xt_LOG xt_limit iptable_mangle
    xt_connmark xt_tcpudp xt_conntrack nf_conntrack nf_defrag_ipv6 nf_defrag_ipv4
    iptable_filter ip_tables x_tables usb_f_mass_storage usb_f_rndis u_ether
    usb_serial_simple usbserial cdc_acm brcmfmac brcmutil smsc95xx usbnet
    ci_hdrc_imx ci_hdrc ulpi usbmisc_imx 8250_exar 8250_pci 8250 8250_base
    libcomposite configfs udc_core
    CPU: 0 PID: 7 Comm: kworker/u8:0 Not tainted 4.19.23-00076-g03740aa-dirty #102
    Hardware name: Freescale i.MX6 Quad/DualLite (Device Tree)
    Workqueue: brcmf_fws_wq brcmf_fws_dequeue_worker [brcmfmac]
    PC is at brcmf_txfinalize+0x34/0x90 [brcmfmac]
    LR is at brcmf_fws_dequeue_worker+0x218/0x33c [brcmfmac]
    pc : [<7f0dee64>]    lr : [<7f0e4140>]    psr: 60010093
    sp : ee8abef0  ip : 00000000  fp : edf38000
    r10: ffffffed  r9 : edf38970  r8 : edf38004
    r7 : edf3e970  r6 : 00000000  r5 : ede69000  r4 : 00000000
    r3 : 00000a97  r2 : 00000000  r1 : 0000888e  r0 : ede69000
    Flags: nZCv  IRQs off  FIQs on  Mode SVC_32  ISA ARM  Segment none
    Control: 10c5387d  Table: 7d03c04a  DAC: 00000051
    Process kworker/u8:0 (pid: 7, stack limit = 0x24ec3e04)
    Stack: (0xee8abef0 to 0xee8ac000)
    bee0:                                     ede69000 00000000 ed56c3e0 7f0e4140
    bf00: 00000001 00000000 edf38004 edf3e99c ed56c3e0 80d03d00 edfea43a edf3e970
    bf20: ee809880 ee804200 ee971100 00000000 edf3e974 00000000 ee804200 80135a70
    bf40: 80d03d00 ee804218 ee809880 ee809894 ee804200 80d03d00 ee804218 ee8aa000
    bf60: 00000088 80135d5c 00000000 ee829f00 ee829dc0 00000000 ee809880 80135d30
    bf80: ee829f1c ee873eac 00000000 8013b1a0 ee829dc0 8013b07c 00000000 00000000
    bfa0: 00000000 00000000 00000000 801010e8 00000000 00000000 00000000 00000000
    bfc0: 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
    bfe0: 00000000 00000000 00000000 00000000 00000013 00000000 00000000 00000000
    [<7f0dee64>] (brcmf_txfinalize [brcmfmac]) from [<7f0e4140>]
(brcmf_fws_dequeue_worker+0x218/0x33c [brcmfmac])
    [<7f0e4140>] (brcmf_fws_dequeue_worker [brcmfmac]) from [<80135a70>]
(process_one_work+0x138/0x3f8)
    [<80135a70>] (process_one_work) from [<80135d5c>] (worker_thread+0x2c/0x554)
    [<80135d5c>] (worker_thread) from [<8013b1a0>] (kthread+0x124/0x154)
    [<8013b1a0>] (kthread) from [<801010e8>] (ret_from_fork+0x14/0x2c)
    Exception stack(0xee8abfb0 to 0xee8abff8)
    bfa0:                                     00000000 00000000 00000000 00000000
    bfc0: 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
    bfe0: 00000000 00000000 00000000 00000000 00000013 00000000
    Code: e1530001 0a000007 e3560000 e1a00005 (05942008)
    ---[ end trace 079239dd31c86e90 ]---

    Signed-off-by: Piotr Figiel <p.figiel@camlintechnologies.com>
    Signed-off-by: Kalle Valo <kvalo@codeaurora.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 2a1ba1ea4d293808051c4d7c2f54131e85af93f4
Author: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
Date:   Fri May 24 10:59:43 2019 -0400

    media: smsusb: better handle optional alignment

    commit a47686636d84eaec5c9c6e84bd5f96bed34d526d upstream.

    Most Siano devices require an alignment for the response.
```

```
    Changeset f3be52b0056a ("media: usb: siano: Fix general protection fault in smsusb")
    changed the logic with gets such aligment, but it now produces a
    sparce warning:

    drivers/media/usb/siano/smsusb.c: In function 'smsusb_init_device':
    drivers/media/usb/siano/smsusb.c:447:37: warning: 'in_maxp' may be used uninitialized in
this function [-Wmaybe-uninitialized]
     447 |   dev->response_alignment = in_maxp - sizeof(struct sms_msg_hdr);
         |                             ~~~~~~~~^~~~~~~~~~~~~~~~~~~~~~~~~~~

    The sparse message itself is bogus, but a broken (or fake) USB
    eeprom could produce a negative value for response_alignment.

    So, change the code in order to check if the result is not
    negative.

    Fixes: 31e0456de5be ("media: usb: siano: Fix general protection fault in smsusb")
    CC: <stable@vger.kernel.org>
    Signed-off-by: Mauro Carvalho Chehab <mchehab+samsung@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 8c28bf8a1ef4af95b22258a1624da9f1f02bc732
Author: Alan Stern <stern@rowland.harvard.edu>
Date:   Tue May 21 11:38:07 2019 -0400

    media: usb: siano: Fix false-positive "uninitialized variable" warning

    commit 45457c01171fd1488a7000d1751c06ed8560ee38 upstream.

    GCC complains about an apparently uninitialized variable recently
    added to smsusb_init_device().  It's a false positive, but to silence
    the warning this patch adds a trivial initialization.

    Signed-off-by: Alan Stern <stern@rowland.harvard.edu>
    Reported-by: kbuild test robot <lkp@intel.com>
    CC: <stable@vger.kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 45b7d1be4bcbc14f01919b739f7b294e4b7d73f1
Author: Alan Stern <stern@rowland.harvard.edu>
Date:   Tue May 7 12:39:47 2019 -0400

    media: usb: siano: Fix general protection fault in smsusb

    commit 31e0456de5be379b10fea0fa94a681057114a96e upstream.

    The syzkaller USB fuzzer found a general-protection-fault bug in the
    smsusb part of the Siano DVB driver.  The fault occurs during probe
    because the driver assumes without checking that the device has both
    IN and OUT endpoints and the IN endpoint is ep1.

    By slightly rearranging the driver's initialization code, we can make
    the appropriate checks early on and thus avoid the problem.  If the
    expected endpoints aren't present, the new code safely returns -ENODEV
    from the probe routine.

    Signed-off-by: Alan Stern <stern@rowland.harvard.edu>
    Reported-and-tested-by: syzbot+53f029db71c19a47325a@syzkaller.appspotmail.com
    CC: <stable@vger.kernel.org>
    Reviewed-by: Johan Hovold <johan@kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 97059f28267753028452ecdf17a4151726945670
Author: Oliver Neukum <oneukum@suse.com>
```

```
Date:   Thu May 9 11:30:59 2019 +0200

    USB: rio500: fix memory leak in close after disconnect

    commit e0feb73428b69322dd5caae90b0207de369b5575 upstream.

    If a disconnected device is closed, rio_close() must free
    the buffers.

    Signed-off-by: Oliver Neukum <oneukum@suse.com>
    Cc: stable <stable@vger.kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 77198d72a4cd6b45222981034c1cf8dc7056c6d5
Author: Oliver Neukum <oneukum@suse.com>
Date:   Thu May 9 11:30:58 2019 +0200

    USB: rio500: refuse more than one device at a time

    commit 3864d33943b4a76c6e64616280e98d2410b1190f upstream.

    This driver is using a global variable. It cannot handle more than
    one device at a time. The issue has been existing since the dawn
    of the driver.

    Signed-off-by: Oliver Neukum <oneukum@suse.com>
    Reported-by: syzbot+35f04d136fc975a70da4@syzkaller.appspotmail.com
    Cc: stable <stable@vger.kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 47991232340128d2a3e7f7515eff9a3e80186e42
Author: Maximilian Luz <luzmaximilian@gmail.com>
Date:   Thu May 16 17:08:31 2019 +0200

    USB: Add LPM quirk for Surface Dock GigE adapter

    commit ea261113385ac0a71c2838185f39e8452d54b152 upstream.

    Without USB_QUIRK_NO_LPM ethernet will not work and rtl8152 will
    complain with

        r8152 <device...>: Stop submitting intr, status -71

    Adding the quirk resolves this. As the dock is externally powered, this
    should not have any drawbacks.

    Signed-off-by: Maximilian Luz <luzmaximilian@gmail.com>
    Cc: stable <stable@vger.kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 90eab08960e71334fd6de55edf1aab60d0b514e1
Author: Oliver Neukum <oneukum@suse.com>
Date:   Thu May 9 14:41:50 2019 +0200

    USB: sisusbvga: fix oops in error path of sisusb_probe

    commit 9a5729f68d3a82786aea110b1bfe610be318f80a upstream.

    The pointer used to log a failure of usb_register_dev() must
    be set before the error is logged.

    v2: fix that minor is not available before registration

    Signed-off-by: oliver Neukum <oneukum@suse.com>
    Reported-by: syzbot+a0cbdbd6d169020c8959@syzkaller.appspotmail.com
```

```
    Fixes: 7b5cd5fefbe02 ("USB: SisUSB2VGA: Convert printk to dev_* macros")
    Cc: stable <stable@vger.kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 7cfb3a00564c6eb6654c604959fef859af2333b6
Author: Alan Stern <stern@rowland.harvard.edu>
Date:   Mon May 13 13:14:29 2019 -0400

    USB: Fix slab-out-of-bounds write in usb_get_bos_descriptor

    commit a03ff54460817c76105f81f3aa8ef655759ccc9a upstream.

    The syzkaller USB fuzzer found a slab-out-of-bounds write bug in the
    USB core, caused by a failure to check the actual size of a BOS
    descriptor.  This patch adds a check to make sure the descriptor is at
    least as large as it is supposed to be, so that the code doesn't
    inadvertently access memory beyond the end of the allocated region
    when assigning to dev->bos->desc->bNumDeviceCaps later on.

    Signed-off-by: Alan Stern <stern@rowland.harvard.edu>
    Reported-and-tested-by: syzbot+71f1e64501a309fcc012@syzkaller.appspotmail.com
    CC: <stable@vger.kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit e0161d24117e8755d99a35a255e7d58f060770b2
Author: Shuah Khan <skhan@linuxfoundation.org>
Date:   Wed May 29 13:46:15 2019 -0600

    usbip: usbip_host: fix stub_dev lock context imbalance regression

    commit 3ea3091f1bd8586125848c62be295910e9802af0 upstream.

    Fix the following sparse context imbalance regression introduced in
    a patch that fixed sleeping function called from invalid context bug.

    kbuild test robot reported on:

    tree/branch: https://git.kernel.org/pub/scm/linux/kernel/git/gregkh/usb.git  usb-linus

    Regressions in current branch:

    drivers/usb/usbip/stub_dev.c:399:9: sparse: sparse: context imbalance in 'stub_probe' -
different lock contexts for basic block
    drivers/usb/usbip/stub_dev.c:418:13: sparse: sparse: context imbalance in
'stub_disconnect' - different lock contexts for basic block
    drivers/usb/usbip/stub_dev.c:464:1-10: second lock on line 476

    Error ids grouped by kconfigs:

    recent_errors
    ├── i386-allmodconfig
    │   └── drivers-usb-usbip-stub_dev.c:second-lock-on-line
    ├── x86_64-allmodconfig
    │   ├── drivers-usb-usbip-stub_dev.c:sparse:sparse:context-imbalance-in-stub_disconnect-
different-lock-contexts-for-basic-block
    │   └── drivers-usb-usbip-stub_dev.c:sparse:sparse:context-imbalance-in-stub_probe-
different-lock-contexts-for-basic-block
    └── x86_64-allyesconfig
        └── drivers-usb-usbip-stub_dev.c:second-lock-on-line

    This is a real problem in an error leg where spin_lock() is called on an
    already held lock.

    Fix the imbalance in stub_probe() and stub_disconnect().
```

```
    Signed-off-by: Shuah Khan <skhan@linuxfoundation.org>
    Fixes: 0c9e8b3cad65 ("usbip: usbip_host: fix BUG: sleeping function called from invalid
 context")
    Cc: stable <stable@vger.kernel.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

 commit 6787013a7e2bd3d79dcba2c7ec55ba8672d24c1c
 Author: Shuah Khan <skhan@linuxfoundation.org>
 Date:   Thu May 2 13:47:02 2019 -0600

    usbip: usbip_host: fix BUG: sleeping function called from invalid context

    commit 0c9e8b3cad654bfc499c10b652fbf8f0b890af8f upstream.

    stub_probe() and stub_disconnect() call functions which could call
    sleeping function in invalid context whil holding busid_lock.

    Fix the problem by refining the lock holds to short critical sections
    to change the busid_priv fields. This fix restructures the code to
    limit the lock holds in stub_probe() and stub_disconnect().

    stub_probe():

    [15217.927028] BUG: sleeping function called from invalid context at mm/slab.h:418
    [15217.927038] in_atomic(): 1, irqs_disabled(): 0, pid: 29087, name: usbip
    [15217.927044] 5 locks held by usbip/29087:
    [15217.927047]  #0: 0000000091647f28 (sb_writers#6){....}, at: vfs_write+0x191/0x1c0
    [15217.927062]  #1: 000000008f9ba75b (&of->mutex){....}, at: kernfs_fop_write+0xf7/0x1b0
    [15217.927072]  #2: 00000000872e5b4b (&dev->mutex){....}, at:
 __device_driver_lock+0x3b/0x50
    [15217.927082]  #3: 00000000e74ececc (&dev->mutex){....}, at:
 __device_driver_lock+0x46/0x50
    [15217.927090]  #4: 00000000b20abbe0 (&(&busid_table[i].busid_lock)->rlock){....}, at:
 get_busid_priv+0x48/0x60 [usbip_host]
    [15217.927103] CPU: 3 PID: 29087 Comm: usbip Tainted: G        W         5.1.0-rc6+ #40
    [15217.927106] Hardware name: Dell Inc. OptiPlex 790/0HY9JP, BIOS A18 09/24/2013
    [15217.927109] Call Trace:
    [15217.927118]  dump_stack+0x63/0x85
    [15217.927127]  ___might_sleep+0xff/0x120
    [15217.927133]  __might_sleep+0x4a/0x80
    [15217.927143]  kmem_cache_alloc_trace+0x1aa/0x210
    [15217.927156]  stub_probe+0xe8/0x440 [usbip_host]
    [15217.927171]  usb_probe_device+0x34/0x70

    stub_disconnect():

    [15279.182478] BUG: sleeping function called from invalid context at
 kernel/locking/mutex.c:908
    [15279.182487] in_atomic(): 1, irqs_disabled(): 0, pid: 29114, name: usbip
    [15279.182492] 5 locks held by usbip/29114:
    [15279.182494]  #0: 0000000091647f28 (sb_writers#6){....}, at: vfs_write+0x191/0x1c0
    [15279.182506]  #1: 00000000702cf0f3 (&of->mutex){....}, at: kernfs_fop_write+0xf7/0x1b0
    [15279.182514]  #2: 00000000872e5b4b (&dev->mutex){....}, at:
 __device_driver_lock+0x3b/0x50
    [15279.182522]  #3: 00000000e74ececc (&dev->mutex){....}, at:
 __device_driver_lock+0x46/0x50
    [15279.182529]  #4: 00000000b20abbe0 (&(&busid_table[i].busid_lock)->rlock){....}, at:
 get_busid_priv+0x48/0x60 [usbip_host]
    [15279.182541] CPU: 0 PID: 29114 Comm: usbip Tainted: G        W         5.1.0-rc6+ #40
    [15279.182543] Hardware name: Dell Inc. OptiPlex 790/0HY9JP, BIOS A18 09/24/2013
    [15279.182546] Call Trace:
    [15279.182554]  dump_stack+0x63/0x85
    [15279.182561]  ___might_sleep+0xff/0x120
    [15279.182566]  __might_sleep+0x4a/0x80
    [15279.182574]  __mutex_lock+0x55/0x950
```

```
     [15279.182582]  ? get_busid_priv+0x48/0x60 [usbip_host]
     [15279.182587]  ? reacquire_held_locks+0xec/0x1a0
     [15279.182591]  ? get_busid_priv+0x48/0x60 [usbip_host]
     [15279.182597]  ? find_held_lock+0x94/0xa0
     [15279.182609]  mutex_lock_nested+0x1b/0x20
     [15279.182614]  ? mutex_lock_nested+0x1b/0x20
     [15279.182618]  kernfs_remove_by_name_ns+0x2a/0x90
     [15279.182625]  sysfs_remove_file_ns+0x15/0x20
     [15279.182629]  device_remove_file+0x19/0x20
     [15279.182634]  stub_disconnect+0x6d/0x180 [usbip_host]
     [15279.182643]  usb_unbind_device+0x27/0x60

     Signed-off-by: Shuah Khan <skhan@linuxfoundation.org>
     Cc: stable <stable@vger.kernel.org>
     Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

  commit 4eff5fb3a19470c5037c44fd08f61ac0186786fa
  Author: Carsten Schmid <carsten_schmid@mentor.com>
  Date:   Wed May 22 14:33:59 2019 +0300

     usb: xhci: avoid null pointer deref when bos field is NULL

     commit 7aa1bb2ffd84d6b9b5f546b079bb15cd0ab6e76e upstream.

     With defective USB sticks we see the following error happen:
     usb 1-3: new high-speed USB device number 6 using xhci_hcd
     usb 1-3: device descriptor read/64, error -71
     usb 1-3: device descriptor read/64, error -71
     usb 1-3: new high-speed USB device number 7 using xhci_hcd
     usb 1-3: device descriptor read/64, error -71
     usb 1-3: unable to get BOS descriptor set
     usb 1-3: New USB device found, idVendor=0781, idProduct=5581
     usb 1-3: New USB device strings: Mfr=1, Product=2, SerialNumber=3
     ...
     BUG: unable to handle kernel NULL pointer dereference at 0000000000000008

     This comes from the following place:
     [ 1660.215380] IP: xhci_set_usb2_hardware_lpm+0xdf/0x3d0 [xhci_hcd]
     [ 1660.222092] PGD 0 P4D 0
     [ 1660.224918] Oops: 0000 [#1] PREEMPT SMP NOPTI
     [ 1660.425520] CPU: 1 PID: 38 Comm: kworker/1:1 Tainted: P     U  W  O    4.14.67-apl #1
     [ 1660.434277] Workqueue: usb_hub_wq hub_event [usbcore]
     [ 1660.439918] task: ffffa295b6ae4c80 task.stack: ffffad4580150000
     [ 1660.446532] RIP: 0010:xhci_set_usb2_hardware_lpm+0xdf/0x3d0 [xhci_hcd]
     [ 1660.453821] RSP: 0018:ffffad4580153c70 EFLAGS: 00010046
     [ 1660.459655] RAX: 0000000000000000 RBX: ffffa295b4d7c000 RCX: 0000000000000002
     [ 1660.467625] RDX: 0000000000000002 RSI: ffffffff984a55b2 RDI: ffffffff984a55b2
     [ 1660.475586] RBP: ffffad4580153cc8 R08: 0000000000d6520a R09: 0000000000000001
     [ 1660.483556] R10: ffffad4580a004a0 R11: 0000000000000286 R12: ffffa295b4d7c000
     [ 1660.491525] R13: 0000000000010648 R14: ffffa295a84e1800 R15: 0000000000000000
     [ 1660.499494] FS:  0000000000000000(0000) GS:ffffa295bfc80000(0000)
  knlGS:0000000000000000
     [ 1660.508530] CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
     [ 1660.514947] CR2: 0000000000000008 CR3: 000000025a114000 CR4: 00000000003406a0
     [ 1660.522917] Call Trace:
     [ 1660.525657]  usb_set_usb2_hardware_lpm+0x3d/0x70 [usbcore]
     [ 1660.531792]  usb_disable_device+0x242/0x260 [usbcore]
     [ 1660.537439]  usb_disconnect+0xc1/0x2b0 [usbcore]
     [ 1660.542600]  hub_event+0x596/0x18f0 [usbcore]
     [ 1660.547467]  ? trace_preempt_on+0xdf/0x100
     [ 1660.552040]  ? process_one_work+0x1c1/0x410
     [ 1660.556708]  process_one_work+0x1d2/0x410
     [ 1660.561184]  ? preempt_count_add.part.3+0x21/0x60
     [ 1660.566436]  worker_thread+0x2d/0x3f0
     [ 1660.570522]  kthread+0x122/0x140
```

```
    [ 1660.574123]  ? process_one_work+0x410/0x410
    [ 1660.578792]  ? kthread_create_on_node+0x60/0x60
    [ 1660.583849]  ret_from_fork+0x3a/0x50
    [ 1660.587839] Code: 00 49 89 c3 49 8b 84 24 50 16 00 00 8d 4a ff 48 8d 04 c8 48 89 ca
 4c 8b 10 45 8b 6a 04 48 8b 00 48 89 45 c0 49 8b 86 80 03 00 00 <48> 8b 40 08 8b 40 03 0f 1f
 44 00 00 45 85 ff 0f 84 81 01 00 00
    [ 1660.608980] RIP: xhci_set_usb2_hardware_lpm+0xdf/0x3d0 [xhci_hcd] RSP:
 ffffad4580153c70
    [ 1660.617921] CR2: 0000000000000008

    Tracking this down shows that udev->bos is NULL in the following code:
    (xhci.c, in xhci_set_usb2_hardware_lpm)
            field = le32_to_cpu(udev->bos->ext_cap->bmAttributes);  <<<<<<< here

            xhci_dbg(xhci, "%s port %d USB2 hardware LPM\n",
                           enable ? "enable" : "disable", port_num + 1);

            if (enable) {
                    /* Host supports BESL timeout instead of HIRD */
                    if (udev->usb2_hw_lpm_besl_capable) {
                            /* if device doesn't have a preferred BESL value use a
                             * default one which works with mixed HIRD and BESL
                             * systems. See XHCI_DEFAULT_BESL definition in xhci.h
                             */
                            if ((field & USB_BESL_SUPPORT) &&
                                (field & USB_BESL_BASELINE_VALID))
                                    hird = USB_GET_BESL_BASELINE(field);
                            else
                                    hird = udev->l1_params.besl;

    The failing case is when disabling LPM. So it is sufficient to avoid
    access to udev->bos by moving the instruction into the "enable" clause.

    Cc: Stable <stable@vger.kernel.org>
    Signed-off-by: Carsten Schmid <carsten_schmid@mentor.com>
    Signed-off-by: Mathias Nyman <mathias.nyman@linux.intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

 commit 239c0a91c61821da7d668788e77906965f986e7a
 Author: Andrey Smirnov <andrew.smirnov@gmail.com>
 Date:   Wed May 22 14:34:01 2019 +0300

    xhci: Convert xhci_handshake() to use readl_poll_timeout_atomic()

    commit f7fac17ca925faa03fc5eb854c081a24075f8bad upstream.

    Xhci_handshake() implements the algorithm already captured by
    readl_poll_timeout_atomic(). Convert the former to use the latter to
    avoid repetition.

    Turned out this patch also fixes a bug on the AMD Stoneyridge platform
    where usleep(1) sometimes takes over 10ms.
    This means a 5 second timeout can easily take over 15 seconds which will
    trigger the watchdog and reboot the system.

    [Add info about patch fixing a bug to commit message -Mathias]
    Signed-off-by: Andrey Smirnov <andrew.smirnov@gmail.com>
    Tested-by: Raul E Rangel <rrangel@chromium.org>
    Reviewed-by: Raul E Rangel <rrangel@chromium.org>
    Cc: <stable@vger.kernel.org>
    Signed-off-by: Mathias Nyman <mathias.nyman@linux.intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

 commit 1f44ae1f98fec0fb8fdeceaf87565da7fdcc39a8
 Author: Fabio Estevam <festevam@gmail.com>
```

```
Date:   Wed May 22 10:35:29 2019 -0300

    xhci: Use %zu for printing size_t type

    commit c1a145a3ed9a40f3b6145feb97789e8eb49c5566 upstream.

    Commit 597c56e372da ("xhci: update bounce buffer with correct sg num")
    caused the following build warnings:

    drivers/usb/host/xhci-ring.c:676:19: warning: format '%ld' expects argument of type
'long int', but argument 3 has type 'size_t {aka unsigned int}' [-Wformat=]

    Use %zu for printing size_t type in order to fix the warnings.

    Fixes: 597c56e372da ("xhci: update bounce buffer with correct sg num")
    Reported-by: kbuild test robot <lkp@intel.com>
    Signed-off-by: Fabio Estevam <festevam@gmail.com>
    Cc: stable <stable@vger.kernel.org>
    Acked-by: Mathias Nyman <mathias.nyman@linux.intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 3983bb9d0bc6197ca606bf5e47d6a276de14b87e
Author: Henry Lin <henryl@nvidia.com>
Date:   Wed May 22 14:33:57 2019 +0300

    xhci: update bounce buffer with correct sg num

    commit 597c56e372dab2c7f79b8d700aad3a5deebf9d1b upstream.

    This change fixes a data corruption issue occurred on USB hard disk for
    the case that bounce buffer is used during transferring data.

    While updating data between sg list and bounce buffer, current
    implementation passes mapped sg number (urb->num_mapped_sgs) to
    sg_pcopy_from_buffer() and sg_pcopy_to_buffer(). This causes data
    not get copied if target buffer is located in the elements after
    mapped sg elements. This change passes sg number for full list to
    fix issue.

    Besides, for copying data from bounce buffer, calling dma_unmap_single()
    on the bounce buffer before copying data to sg list can avoid cache issue.

    Fixes: f9c589e142d0 ("xhci: TD-fragment, align the unsplittable case with a bounce
buffer")
    Cc: <stable@vger.kernel.org> # v4.8+
    Signed-off-by: Henry Lin <henryl@nvidia.com>
    Signed-off-by: Mathias Nyman <mathias.nyman@linux.intel.com>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit a40d0bd2fd4e38fb66fd058cf294947036310d1a
Author: Rasmus Villemoes <linux@rasmusvillemoes.dk>
Date:   Tue May 14 15:43:27 2019 -0700

    include/linux/bitops.h: sanitize rotate primitives

    commit ef4d6f6b275c498f8e5626c99dbeefdc5027f843 upstream.

    The ror32 implementation (word >> shift) | (word << (32 - shift) has
    undefined behaviour if shift is outside the [1, 31] range.  Similarly
    for the 64 bit variants.  Most callers pass a compile-time constant
    (naturally in that range), but there's an UBSAN report that these may
    actually be called with a shift count of 0.

    Instead of special-casing that, we can make them DTRT for all values of
    shift while also avoiding UB.  For some reason, this was already partly
```

```
    done for rol32 (which was well-defined for [0, 31]).  gcc 8 recognizes
    these patterns as rotates, so for example

      __u32 rol32(__u32 word, unsigned int shift)
      {
            return (word << (shift & 31)) | (word >> ((-shift) & 31));
      }

    compiles to

    0000000000000020 <rol32>:
      20:   89 f8                   mov    %edi,%eax
      22:   89 f1                   mov    %esi,%ecx
      24:   d3 c0                   rol    %cl,%eax
      26:   c3                      retq

    Older compilers unfortunately do not do as well, but this only affects
    the small minority of users that don't pass constants.

    Due to integer promotions, ro[lr]8 were already well-defined for shifts
    in [0, 8], and ro[lr]16 were mostly well-defined for shifts in [0, 16]
    (only mostly - u16 gets promoted to _signed_ int, so if bit 15 is set,
    word << 16 is undefined).  For consistency, update those as well.

    Link: http://lkml.kernel.org/r/20190410211906.2190-1-linux@rasmusvillemoes.dk
    Signed-off-by: Rasmus Villemoes <linux@rasmusvillemoes.dk>
    Reported-by: Ido Schimmel <idosch@mellanox.com>
    Tested-by: Ido Schimmel <idosch@mellanox.com>
    Reviewed-by: Will Deacon <will.deacon@arm.com>
    Cc: Vadim Pasternak <vadimp@mellanox.com>
    Cc: Andrey Ryabinin <aryabinin@virtuozzo.com>
    Cc: Jacek Anaszewski <jacek.anaszewski@gmail.com>
    Cc: Pavel Machek <pavel@ucw.cz>
    Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
    Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

commit 42962ec2cd984a82b86fb46a0a634d15dd4a7c67
Author: James Clarke <jrtc27@jrtc27.com>
Date:   Wed May 29 22:31:31 2019 +0100

    sparc64: Fix regression in non-hypervisor TLB flush xcall

    commit d3c976c14ad8af421134c428b0a89ff8dd3bd8f8 upstream.

    Previously, %g2 would end up with the value PAGE_SIZE, but after the
    commit mentioned below it ends up with the value 1 due to being reused
    for a different purpose. We need it to be PAGE_SIZE as we use it to step
    through pages in our demap loop, otherwise we set different flags in the
    low 12 bits of the address written to, thereby doing things other than a
    nucleus page flush.

    Fixes: a74ad5e660a9 ("sparc64: Handle extremely large kernel TLB range flushes more
gracefully.")
    Reported-by: Meelis Roos <mroos@linux.ee>
    Tested-by: Meelis Roos <mroos@linux.ee>
    Signed-off-by: James Clarke <jrtc27@jrtc27.com>
    Signed-off-by: David S. Miller <davem@davemloft.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```