Backporting Issues with Multi-subsystem Devices

Don Dutile

RHEL RDMA Maintainer
ddutile@redhat.com
September 13, 2017

Linux Plumbers, RDMA Mini-conf

Agenda

- Why is RDMA Subsystem updated in RHEL ?
- What comprises RDMA in Linux (and RHEL)
- What a RHEL RDMA Update Entails
- Kernel Subsystem Dependencies
- Sounds simple …
- Some RDMA backporting stats
- Partners
- Testing
- Summary

Why is RDMA Subsystem Updated in RHEL ?

- Large churn upstream
- (+) Drivers, core and userspace in lock-step
- (+) RDMA customers & partners want latest upstream
  - Few/any(?) want it to 'stay stable, dont change it'
- Updated every minor release
  - Until end of 'Production Phase 1' of RHEL
  - Typically two major RHEL releases in Phase 1 in time
- No kabi, no OOB drivers supported
  - But many other parts of RHEL have kabi
  - RDMA customers want latest & no OOB drivers, e.g., NASA COTS testing for Mars !

What Comprises RDMA in Linux (and RHEL)

- drivers/infiniband/core
  - Needs renaming to drivers/rdma
- drivers/infiniband/ulp's
  - IPoIB: let the IP dependencies begin…
  - SRP: SCSI
  - ISER: add 'i(p)' to 'SCSI'
  - Host & target for SRP & ISER
- drivers/infiniband/sw – rdmavt, rxe; soft-iWARP coming
- Drivers dependent on above:
  - Drivers/infiniband/hw – easiest part to maintain (with core)
  - Drivers/net/ethernet/<vendor>/<driver>: core, RoCE, iWARP – typically hardest to maintain; coordination w/CNB

What Comprises RDMA in Linux (continued)

- Drivers (continued)
  - 17 drivers in RHEL-7 (some deprecated upstream)
  - Six (active) partners … more on them later
- NFSoRDMA (kudos Chuck Lever; net/sunrpc/xprtrdma)
- New additions:
  - Cgroups
  - SELinux
  - NVMEoF
- Plus 40+ userspace packages
  - In lock-setp w/kernel (rdma-core, jwilson; others...)

What a RHEL RDMA Update Entails

- Git-log-based script to generate upstream commits
  - Subsys & files from previous slides
  - Extract per-upstream version, i.e., v4.11, v4.12,…
  - Seconds to perform
- Parse upstream commit list
  - Drop kabi breakers ('tree-wide', mm, cpu, sched, 'cleanup')
  - Previous backports
  - Patch series others want to do (partners, other driver owners)
  - git-backport: tool to generate **quilt** series patch set
  - Hours, sometimes days to perform

What a RHEL RDMA Update Entails (cont.)

- quilt (phase)
  - Speed up over c-p+rebase when doing 100's patches
  - 30 min git rebase; 15→30sec quilt pop/push+git-am for 300→400 commits
  - Man dependency resolutions, additional backports
    - 3days →4 weeks
    - First v4.x update longer due to additional dependencies
      - Subsequent v4.x+n faster (1.5wks; 1 wk; 3 days)
    - Approx 50 local builds, dozens all-arch builds to flush out missing dependencies and/or compilation errors
- Patchreview
  – Compare RHEL patches to upstream (tcamuso)

What a RHEL RDMA Update Entails (cont.)

- Typically starts with 400→500 patches per v4.x
  - Parse down to high-200/high 300 count
  - **Typically end with 50→150 more dependency backports**
    - Nfs, scsi, target, block, net, dma, arch/* ….
- Repeat for each v4.x version
- Have to make patch set reviewable, testable
  - Echo of Corbet's comments on reviews:
    RHEL requires 3 ack's for a patch to be included
  - RHEL X.Y every 6→8 mos == 3 upstream releases
    - So 3 →4 'RDMA patch bombs' per minor release
    - Majority of work done in 3 month window, bug fix as progress

What is RHEL RDMA Update Entails (cont.)

- Bisect check
  - Request by RHEL kernel maintainer
    - == Linus for RHEL → make him happy!
- Unique kernel version per patch bomb released internally
  e.g., kernel-3.10-674.el7.<arch>

Kernel Subsystem Dependencies

- Net
  - RoCE, iWARP
  - IPoIB (IP over RDMA (it's over OPA)
    - 'new IPoIB, aka, device-specific accelerators
  - Pull out multi-LA dictionary: IPv4,IPv6, VLAN, VxLAN, IPSec, GRE, TSO,…
  - Core: flow dissector, netsched, ip, ethtool, …
  - Numerous RHEL net-hooks to enable new functionality and maintain kabi
  - CNB: Common Network Backport → critical to success in this area (see stats)

Kernel Subsystem Dependencies (cont.)

- NFS
  - Some dependencies, but centric to net/sunrpc/xprtrdma
  - RHEL NFS team updates frequently
  - Good upstream NFSoRDMA maintainer
- (i)scsi, target, block:
  - Significant upstream churn in this area
    - [“Hellwig!” == “Neumann!”]
    - Bart & Nicholas get honorable mention here too!
      - But Bar's churn usually drivers/infiniband-centric: no kabi!
  - Far less degrees of built-in kabi freedom (vs net)
  - older then net-core; no CSB/CTB in each RHEL release
- cgroups and SELinux → the more the merrier!

Sounds Simple ...

- RHEL has a kabi – major blockage for backporting, unless you make a RHEL-only change in many places
- Upstream patches dependent on… rest of upstream
  - (Corbet): Yeah, 'upstream-first' best for all
  - RDMA updates means min update of other subsys
  - Need to stay close to upstream functionality
    - Bugs: is it upstream or RHEL-specific?
    - De-Frankenstein-ing RHEL (v4.13, v4.14)
- In-line funtions
  - Important to inline 'dma_supported(), 'dma_[set,get]_mask()
  - Perf ops **may** benefit; cache pressure affects ?
  - Subject: x86: Deinline dma_[alloc,free]_attrs
    - 68K+ bytes of the kernel

Sounds Simple (continued)

- Example v4.11 RDMA backport:
  - 551199aca Subject: lib/dma-virt: Add dma_virt_ops
    - Req'd 29 previous, dma_[map_]ops patches
      - Kabi: many functions are inlines in h files
    - Initial backport broke 400 kabi interfaces
      - Dont mess with struct device !
      - 2 days to find patch that resolved 398
      - Modifying 2 other patches resolved 2 other breaks
    - (Corbet) No one owns/maintains 'dma interface'
      – upstream & RHEL

Sounds Simple (continued)

- Example v4.11 RDMA backport b42057ab1 Subject: ib_srpt: Convert to target_alloc_session
  - Required 22 drivers/target -centric patches
  - Took two weeks to mangle into compile-able state
  - 30 kabi failures
  - Fixed by modifying 1 patch, drop 1 patch that broke kabi
  - RHEL drivers/target basically at v4.1 level
- Other RHEL subsystems follow (conflicting) “don't make changes, just bug fix-it” mentality
  - RDMA mindshare in RHEL needs equivalent highlighting as in upstream! :-p

Some RHEL RDMA Backporting Stats

- Each RHEL X.Y release consists of 8K→10K patches
- RDMA:
  - RHEL-7.3 rebases: 996 patches
  - RHEL-7.4 rebases: 1256 patches
  - Post-rebase bug fixes: 10→20 patches (see testing)
- CNB: (ivecera)
  - RHEL 7.1: 10; 7.2: 51 ; 7.3: 195; 7.4: 766
  - RHEL-7.4 breakdown:
    - Net-sched : 265
    - Bridge : 232
    - Switchdev : 117
    - Net core : 70
    - Ipv4/6/fib : 22
    - Devlink : 10
    - Ethtoool : 8
    - Misc : 25

Some RHEL RDMA Backporting Stats (cont)

- RDMA + CNB account for 10% → 20% of RHEL X.Y
- Platform Enablement:
  - RDMA, PCI, IOMMU, USB, platform, ACPI, UEFI, net-drivers, ALSA
  - 70 → 80% RHEL X.Y release
    - New functionality + don't update unless needed for fix or new functionality

Partners

- It takes a village ….
- RHEL has 'partner engineers' that add patches, fix (update) bugs, and most important: **test updates**
- Six (RDMA) partners over 17 drivers
  - RHEL has 10 year life cycle
  - Driver deprecation/removal makes support difficult
    - dma_map_ops in v4.11 (ehca, ipath)
  - Major RHEL release is deprecation point (go RHEL-8!)
- Partner engineers have access to internal git trees
  - RDMA rebase git tree
  - Partner collaboration critical
  - Access to nightly repos, pre-public (Alpha, Beta)
- Coordination with partners large time commitment

TESTING

- Each v4.x backport tested by each affected partner before internal RKML posting
  - RH RDMA QE does regression testing, debug new tests
  - Target: complete all sets before Alpha kernel
    - Alpha kernel approx 3 mos before GA
    - If miss Alpha, must be done before Beta (function freeze)
    - Partners & RH add bug-fixes to rebases
  - If partner can't commit to testing, new functionality dropped
  - Early v4.x backports have longer test times then latter ones

TESTING (continued)

- Westford RDMA Cluster
  - Pandora's box of systems, devices, switches
    - Dell, HPE, Intel-white, PPC, ARM64
    - Mellanox, Dell, Intel switches
    - All systems connected via switch: no back-to-back
  - Used by RHEL (devel +QE) + Upstream(Doug) + partners
    - Others: net, NFS, virt, storage
  - Expanded from 3 racks to 4 racks in July
    - 4 weeks to tear down & put Humpty-Dumpty back together
    - Now adding more servers & devices
  - Partners test configs RH doesnt have

How Backports Could Be Made Simpler

- Conditionally configure new features
  - (net) flow dissector, tc-offload
  - XDP
  - Separate modules help
    - Improvement: conditional inclusion/stubbing of init ops
- Refactor once!
  - Scsi mq is the classic (bad) case (RHEL-7.3)
    - Corollary: design better the first time
      - If >1 upstream release, highlight completion point, ref start point
      - RHEL: now scan 1→2 upstream releases: 'skip to end' patches
- Get new functionality upstream sooner
  - Doug/linux-rdma no longer accepting rdma-next dependencies from net-next; net dependencies must be in previous v4.x release – light at end of tunnel?

How Backports Could Be Made Simpler (cont.)

- Keep older drivers maintained
  - RHEL has 10 year cycle for a reason: customer use
- Don't submit patches with build warnings
- Identify refactoring, new functionality dependencies in patch posting
  - Helped ib_dma_map_ops → common dma_map_ops
- RHEL
  - Get other subsystems to update more often
  - Better kabi hooks in scsi, target, block

Questions?