BUILDING A BLOCK STORAGE APPLICATION ON OFED - CHALLENGES

Subhojit Roy, Tej Parkash, Lokesh Arora, Storage Engineering

[May 26th, 2017]
Introduction

- Setting the Context (SVC as Storage Virtualizer)
- SVC Software Architecture overview
- iSER: Confluence of iSCSI and RDMA
- Performance: iSER v/s Fibre Channel

Challenges

- Queue Pair states
- RDMA disconnect behavior
- RDMA connection management
- Large DMA memory allocation
- Query Device List
- Conclusion
• SVC pools heterogeneous storage and virtualizes it for the host

• iSER Target for Host

• iSER Initiator for Storage Controller (FLASH or HDD)

• Clustered over iSER for high availability

• Supports both RoCE and iWARP

• Supports 10/25/40/50/100G bandwidths
SVC SOFTWARE ARCHITECTURE OVERVIEW

- SVC application runs in user space
- iSER and iSCSI drivers in kernel space
- Lockless architecture (Per CPU port handling)
- Polled mode IO handling
- Supports RoCE and iWARP
- Vendor Independent (Mellanox, Chelsio, Qlogic, Broadcom, Intel etc.)
- Dependence on OFED kernel IB Verbs
iSER: Confluence of iSCSI and RDMA

• iSER is iSCSI with a RDMA data path

• Performance: Low Latency, Low CPU utilization, High Bandwidth

• High Bandwidth: 25Gb, 50Gb, 100Gb and beyond

• No new administration! Leverages existing knowledge of iSCSI administration & ecosystem on servers and storage
## PERFORMANCE: iSER vs FIBRE CHANNEL

<table>
<thead>
<tr>
<th>I/O</th>
<th>iSER (40Gb)</th>
<th>Fibre Channel (16Gb)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read 4KiB</td>
<td>50 (us)</td>
<td>80 (us)</td>
</tr>
<tr>
<td>Write 4KiB</td>
<td>139 (us)</td>
<td>195 (us)</td>
</tr>
<tr>
<td>Read 64KiB</td>
<td>95 (us)</td>
<td>196 (us)</td>
</tr>
<tr>
<td>Write 64KiB</td>
<td>209 (us)</td>
<td>337 (us)</td>
</tr>
</tbody>
</table>

*iSER: Fiber Channel benefits minus the additional costs*
CHALLENGES
QUEUE PAIR STATES

- **Goal**
  - Control number of retries and retry timeout during network outage

- **Actual behavior**
  - State transition differs across RoCE and iWARP e.g. RoCE does not support SQD state

- **Expectation**
  - Transition QP to SQD state to modify QP attributes
  - `ib_modify_qp()` must transition QP states as per state diagram shown
  - All state transition must be supported by both RoCE and iWARP

- **Work Around**
  - No work around found
  - Exploring vendor specific possibilities

Referenced from book “Linux Kernel Networking - Implementation and Theory”
RDMA DISCONNECT BEHAVIOR

- **Goal/Observation**
  - QP cannot be freed before `RDMA_CM_EVENT_DISCONNECTED` event is received
  - There is no control over the timeout period for this event

- **Actual behavior**
  - Link down on peer system causes DISCONNECT event to be received after long delay
    - RoCE: ~100 Sec
    - iWARP: ~70 Sec
  - There is no standard mechanism (verb) to control these timeouts

- **Expectation**
  - RDMA disconnect event must exhibit uniform timeout across RoCE and iWARP
  - Timeout period for disconnect must be configurable

- **Work Around**
  - Evaluating vendor specific mechanism to tune CM timeout
• **Goal**
  - Polled mode data path and Connection Management

• **Current mechanism**
  - No mechanism to poll for CM events. All RDMA CM events are interrupt driven
  - Current implementation involves deferring CM events to Linux workqueues
  - Application has no control over which CPU to POLL CM events from

• **Expectation**
  - Queues for CM event handling

• **Work Around**
  - Usage of locks add to IO latency
LARGE DMA MEMORY ALLOCATION

- **Observation**
  - Allocation of large chunks DMAable memory during session establishment fails
  - SVC reserves majority of physical memory during system initialization for caching

- **Current mechanism**
  - IB Verbs use `kmalloc()` to allocate DMAable memory for all the queues

- **Expectation**
  - IB Verbs must provide a means to allocate DMA-able memory from pre-allocated memory pool. e.g. in the following
    - `ib_alloc_cq()`
    - `ib_create_qp()`

- **Work Around Solutions**
  - Modified iWARP and RoCE driver to use pre-allocated memory pools from SVC

<table>
<thead>
<tr>
<th>Type</th>
<th>Elements</th>
<th>Size</th>
<th>Total Size(KB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SQ</td>
<td>2064</td>
<td>88</td>
<td>~177KB</td>
</tr>
<tr>
<td>RQ</td>
<td>2064</td>
<td>32</td>
<td>~64KB</td>
</tr>
<tr>
<td>CQ</td>
<td>2064</td>
<td>32</td>
<td>~64KB</td>
</tr>
</tbody>
</table>

Single Connection Memory requirement in Linux OFED Stack = ~297KB
- **Observation**
  - No kernel verb to find list of rdma devices on system until RDMA session is established
  - Per device resource allocation during kernel module initialization

- **Current mechanism**
  - RDMA device available only after connection request is established by CM event handler

- **Expectation**
  - Need verb equivalent to `ibv_get_device_list()` in kernel IB Verbs

- **Work Around**
  - Complicates per port resource allocation during initialization
CONCLUSION

- Initial indications of IO performance compared to FC – excellent!
- iSER presents an opportunity for high performance Flash based Ethernet data center
- Error recovery and handling is still evolving
- Mass adoption by storage vendors requires more work in OFED
  - IB Verbs is not completely protocol independent
  - Proper documentation of RoCE vs iWARP specific difference
  - Definitive resource allocation timeout values (R_A_TOV equivalent in FC)
- Same requirements applicable to NVMef
THANK YOU

subhojit.roy@in.ibm.com, tprakash@in.ibm.com, loharora@in.ibm.com

[May 26th, 2017]