

# Table of Contents

	OpenVZ / LXC / Zones	KVM
CPU Performance	native	high (with CPU support)
CPU Allocation	flexible (FSS + "bursting")	fixed to VCPU limit
I/O Throughput	high (no overhead)	low or medium (with virtio)
I/O Latency	low (native)	some (I/O proxy overhead)
Memory Access Overhead	none	some (EPT/NPT or shadow page tables)
Memory Loss	none	some (extra kernels; page tables)
Memory Allocation	flexible (unused memory used for file system cache)	fixed (and possible double-caching)
Resource Controls	many (depends on OS)	most (OS + hypervisor)
Observability from the host	highest (see everything)	medium (resource usage, hypervisor statistics, OS inspection of hypervisor)
Observability: from the guest	medium (see everything permitted, incl. some physical resource stats)	low (guest only)
Hypervisor Complexity	low (OS partitions)	medium
Different OS Guests	no	yes

Networking (Container vs KVM) aprox. 5x slower.  
 HDD IO (Container vs KVM) aprox. 4x slower.

```

/*
 *
 *          GUEST                                #          QEMU
 * #####
 *
 * +-----+
 * | start_ | (1)
 * | xmit() |
 * +-----+
 *
 *     ||
 *     ||          +-----+
 *     ||----->|free_old_ | (2)
 *     ||----->|xmit_skbs()|
 *     ||          +-----+
 *     \/\          (3)
 * +-----+          +-----+          + - #--- PIO write to VNIC
 * | xmit_ |----->|virtqueue_add| | #   PCI config space (6)
 * | skb() |----->|_buf_gfp() | | #
 * +-----+          +-----+          | #
 *     ||              |          | # +- VM exit
 *     ||          +- iff interrupts | # | KVM driver exit (7)
 *     \/\          | unmasked (4) | # |
 * +-----+          |          +-----+(5) | # | +-----+
 * |virtqueue|----*---->|vp_notify()|----*---#-*->| handle | (8)
 * |_kick() |----*---->|          |----*---#-*->|PIO write|
 * +-----+          +-----+          # +-----+
 *     ||              |          | #          ||
 *     || (13)          |          | #          ||
 *     **-----+ iff avail ring |          | #          \/\          (9)
 *     ||          capacity < 20 |          | # +-----+
 *     ||          else return    |          | # |virtio_net_handle|
 *     ||              |          | # |tx_timer() |
 *     \/\          (14)          |          | # +-----+
 * +-----+          |          | # ||
 * |netif_stop|          |          | # ||          (10)
 * |_queue() |          |          | # ||          +-----+
 * +-----+          |          | # ||-->|qemu_mod_|
 *     ||              |          | # ||-->|timer() |
 *     || (15)          |          | # ||          +-----+
 * +-----+          +-----+          | # ||
 * |virtqueue_enable|---->|unmask | | # ||          (11)
 * |_cb_delayed() |---->|interrupts| | # ||          +-----+
    
```

```

* +-----+ +-----+ # |+->|virtio_ |
* || || # +-->|queue_set_ |
* || (18) || (17) # |notification|
* || +-return +-----+ # +-----+
* || | iff ---->|check if the number| # |
* **--+ is false |of unprocessed used| # | disable host
* || |ring entries is > | # +- interrupts
* || |3/4s of the avail | # (12)
* \/\ (19) |ring index - the | #
* +-----+ |last freed used | #
* |free_old_ | |ring index | #
* |xmit_skbs()| +-----+ #
* +-----+ #
* || #
* || (20) #
* **-----+ iff avail ring #
* || capacity is #
* || now > 20 #
* \/\ #
* +-----+ #
* |netif_start| (21) #
* |_queue() | #
* +-----+ #
* || #
* || #
* \/\ (22) (23) #
* +-----+ +-----+ #
* |virtqueue_ |----->|mask | #
* |disable_cb()|----->|interrupts| #
* +-----+ +-----+ #
* #
* #
*/

```

Figure II: Guest / Host Packet TX Part 1</code>

From: <https://wiki.janforman.com/> - **wiki.janforman.com**

Permanent link: <https://wiki.janforman.com/virtualization>

Last update: **2019/07/09 12:42**

