

KVM

CPU model

[QEMU / KVM CPU model configuration](#)

AES

Enable AES in CPU flags. Default KVM64 CPU doesn't expose AES flag. Simple openssl benchmark:

```
openssl speed -evp aes-128-cbc aes-256-cbc aes-256-ecb

# Without AES
type          16 bytes      64 bytes      256 bytes    1024 bytes    8192
bytes 16384 bytes
aes-256 cbc    185216.65k    190818.37k    191588.35k    193247.23k
193489.58k    193353.05k
aes-128-cbc    220375.57k    245515.09k    249103.70k    254411.43k
255770.62k    255393.79k

# With AES
type          16 bytes      64 bytes      256 bytes    1024 bytes    8192
bytes 16384 bytes
aes-256 cbc    183729.40k    191020.12k    190906.71k    193176.58k
193333.93k    194065.47k
aes-128-cbc    587427.84k    1272103.38k    1317980.93k    1329665.71k
1332060.16k    1332663.64k
```

DRAFT

Fastest method: create Proxmox machine and import vbox hard discs to newly created Proxmox VM.

No VirtualBox preparation, no IDE drivers. Just import discs to new machine.

- Create new Proxmox machine with similar parameters to old one
 - [Windows 10 guest best practices](#)
 - Do not remove created Hard Disc (scsi0) - it will be used to trigger Windows to install SCSI drivers.
- Import disc images (repeat for all VDI discs)
 - As RAW images with all benefits (compression, thin provisioning, snapshots) of local-zfs storage:

```
qm importdisk 701 WIN7_C.vdi local-zfs
```

- 21743 MB VDI compacted file occupied 15,2G on ZFS with compression ratio 1.24x
- As QCOW2 files on every file storage:

```
qm importdisk 701 WIN7_C.vdi local -format qcow2
```

- Connect new discs into machine:

```
qm set 701 --scsi<N> local-zfs:vm-701-disk-<N>,discard=on,size=64G
```

- NOTE: discard=on should be enabled for thin provisioned storage to reclaim unused space
- Edit config file to change just connected discs from SCSI to IDE. Remember to update CDROM IDE number to do not be in conflict with new discs.

```
sed -i 's/scsi/ide/g' /etc/pve/qemu-server/702.conf
```

- [/etc/pve/qemu-server/702.conf](#)

```
ide0: local-zfs:vm-702-disk-1,discard=on,size=64G
ide1: local-zfs:vm-702-disk-2,discard=on,size=64G
ide2: local-zfs:vm-702-disk-3,discard=on,size=8G
ide3: nas326-ISOs:iso/virtio-
win-0.1.173.iso,media=cdrom,size=384670K
scsi0: local-zfs:vm-702-disk-0,discard=on,size=32G
scsihw: virtio-scsi-pci
```

- Setup correct boot order
- Start Windows 10
 - Windows should load correctly
 - Go to Device Manager and install missing drivers. See [Virtio drivers](#)
 - Shutdown system
- Edit config file to change IDE discs to SCSI
- Setup correct boot order
- Start Windows 10
- ERROR: Windows stuck during booting when VirtIO SCSI driver.
 - It is known KVM issue with VirtIO SCSI driver when more than 1 CPU core is used!
 - https://bugzilla.redhat.com/show_bug.cgi?id=1670673
 - Driver version 61.77.104.17200 2019-06-07 from virtio iso v 1.1.172
 - Update to latest 61.80.104.17300 2019-08-12 (not stable) virtio iso v1.1.173 doesn't help
 - Changing CPU to 1 core solves problem.
 - It works with odd number of cores (1,3,5)
- Windows will crash (because storage driver is not supported). Windows will reboot into repair mode.
- Provide virtio drivers in virtual cd-rom (latest 1.1.173 was used)
- Select Start-up Repair
- Switch SCSI to IDE:

```
sed -i 's/scsi/ide/g' /etc/pve/qemu-server/701.conf
```

- Manually correct numbering conflict with CD-ROM ide drive
- Setup correct boot order
- Add one dummy (small size) hard disc to SCSI controller
- Enable QEMU Guest Agent
- Boot Windows, go to Device Manager and install missing drivers:

...

- Shutdown machine
- Switch disc controller from IDE to SCSI. Edit /etc/pve/qemu-server/701.conf
 - Setup correct boot order

```
sed -i 's/ide/scsi/g' /etc/pve/qemu-server/701.conf
```

From:

<https://niziak.spox.org/wiki/> - **niziak.spox.org**

Permanent link:

<https://niziak.spox.org/wiki/vm:proxmox:kvm>

Last update: **2022/12/31 17:25**

