

Hyper-V (VHDX) → KVM (QCOW2) Ubuntu VM Recovery & Migration — FULL COMMAND PLAYBOOK

This document records every command required to migrate a Hyper-V Gen2 Ubuntu VM to KVM/libvirt, including all intermediary recovery and repair steps encountered in practice.

0. Host Preparation

```
sudo apt update
sudo apt install -y qemu-kvm libvirt-daemon-system libvirt-clients virtinst virt-manager
qemu-utils libguestfs-tools ovmf
sudo systemctl enable --now libvirtd
```

1. Repair and Convert VHDX

```
qemu-img check -r all source.vhdx
qemu-img convert -p -O qcows2 source.vhdx vm.qcows2
```

2. Inspect Guest Disk Layout

```
virt-filesystems -a vm.qcows2 --all --long -h
```

3. Create VM (Import Mode)

```
sudo virt-install --import \
--name vm-name \
--memory 4096 \
--vcpus 2 \
--cpu host \
--disk path=/srv/vms/vm.qcows2,format=qcows2,bus=virtio \
--os-variant ubuntu22.04 \
--network network=default,model=virtio \
--graphics none
```

4. Force UEFI Firmware (OVMF)

```
sudo virsh edit vm-name
<os>
<type arch='x86_64' machine='pc-q35-7.2'>hvm</type>
<loader readonly='yes' type='pflash'>/usr/share/OVMF/OVMF_CODE_4M.fd</loader>
<nvram>/var/lib/libvirt/qemu/nvram/vm-name_VARS.fd</nvram>
</os>
```

5. Add Temporary VNC

```
<graphics type='vnc' port='1' listen='127.0.0.1' />
<video><model type='vga' /></video>
sudo virsh start vm-name
sudo virsh vncdisplay vm-name
```

6. Expected UEFI Boot Loop

Boot Manager → ubuntu (loops)

```
Boot Manager → EFI Shell
map -r
fs0:
cd EFI\ubuntu
grubx64.efi # loops (expected)
```

7. Attach Ubuntu Live ISO (cold attach)

```
sudo virsh destroy vm-name
sudo virsh attach-disk vm-name ubuntu-22.04-live-server-amd64.iso sda --type cdrom
--targetbus sata --config
sudo virsh start vm-name
```

8. Boot ISO via UEFI

```
Boot Manager → UEFI DVD/CD
Select 'Rescue a broken system'
Drop to shell
```

9. Mount Installed System

```
lsblk
mkdir -p /mnt/boot/efi
mount /dev/ubuntu-vg/ubuntu-lv /mnt
mount /dev/vda2 /mnt/boot
mount /dev/vda1 /mnt/boot/efi
mount --bind /dev /mnt/dev
mount --bind /proc /mnt/proc
mount --bind /sys /mnt/sys
chroot /mnt
```

10. EFI Vars + GRUB Reinstall

```
mount -t efivars efivars /sys/firmware/efi/efivars
grub-install --target=x86_64-efi --efi-directory=/boot/efi --bootloader-id=ubuntu --recheck
update-grub
```

11. Exit and Reboot

```
exit
/bin/umount -R /mnt
reboot
```

12. Remove ISO (host)

```
sudo virsh destroy vm-name
sudo virsh detach-disk vm-name sda --config
sudo virsh start vm-name
```

13. Fix Networking (Netplan rename)

```
ip link
sudo nano /etc/netplan/*.yaml
network:
```

```
version: 2
ethernets:
enp1s0:
dhcp4: true
sudo netplan apply
```

14. Final Services

```
sudo systemctl enable serial-getty@ttyS0.service
sudo apt install -y qemu-guest-agent openssh-server
sudo systemctl enable --now qemu-guest-agent ssh
```

15. Host Access Test

```
sudo virsh domifaddr vm-name
ssh user@<vm-ip>
```