Hybrid Backup

The snapshot is dead, long live the snapshot!

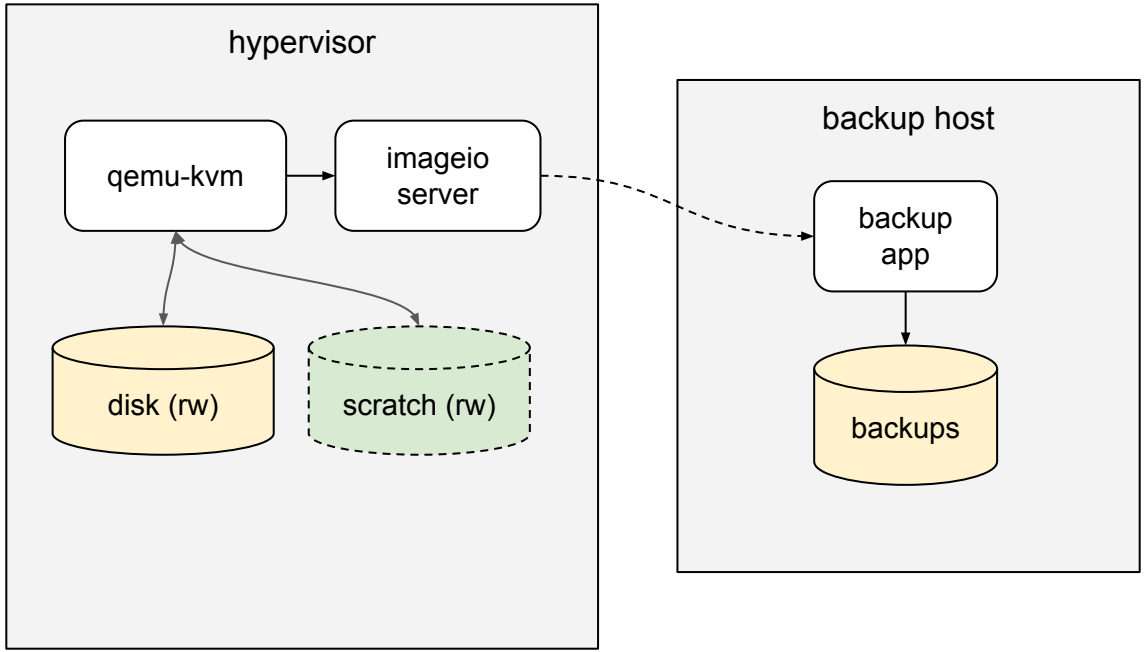
Nir Soffer
Principal Software Engineer
nsoffer@redhat.com

Benny Zlotnik
Principal Software Engineer
bzlotnik@redhat.com

RHV Demo, March 2022

Why?

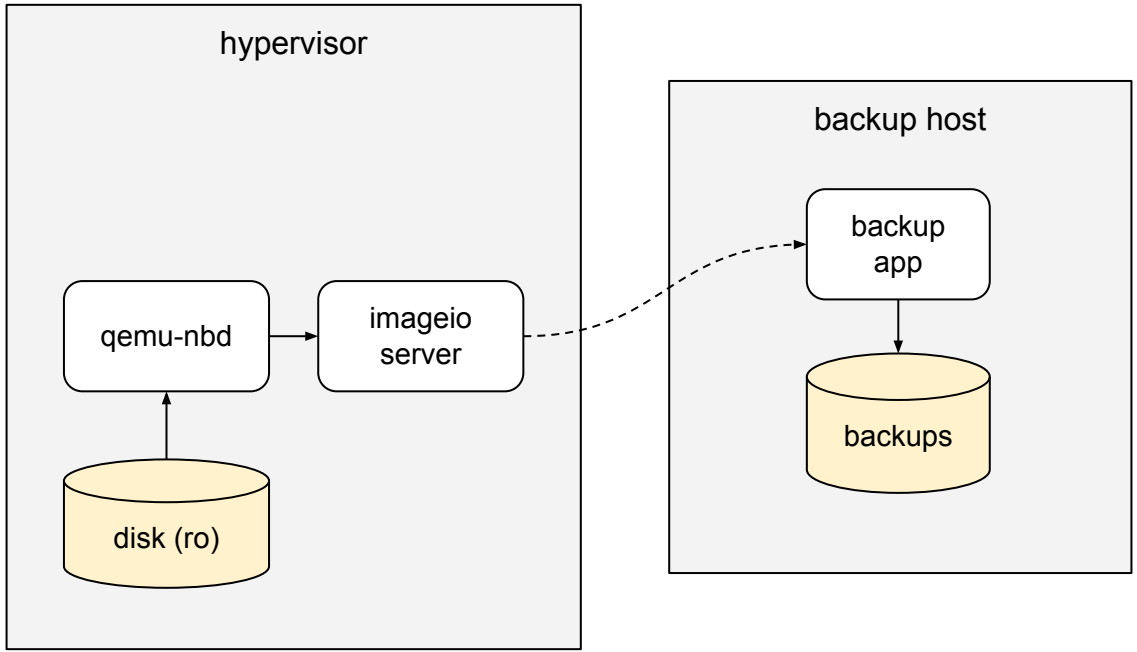
Live backup


hypervisor
qemu-kvm
imageio server
disk (rw)
scratch (rw)
backup host
backup app
backups

Live backup problems

- ✗ Can start only when VM is running
- ✗ Cannot power down the VM during backup
- ✗ Cannot migrate a VM during backup
- ✗ If VM is stopped via API (force), backup is aborted
- ✗ If VM is powered down within the guest, backup is aborted
- ✗ If the scratch disk becomes full, VM pauses
- ✗ Backup I/O degrades guest I/O
- ✗ Guest I/O degrades backup I/O
- ✗ Complex flow on engine

Cold backup


hypervisor
qemu-nbd
imageio server
disk (ro)
backup host
backup app
backups

Cold backup problems

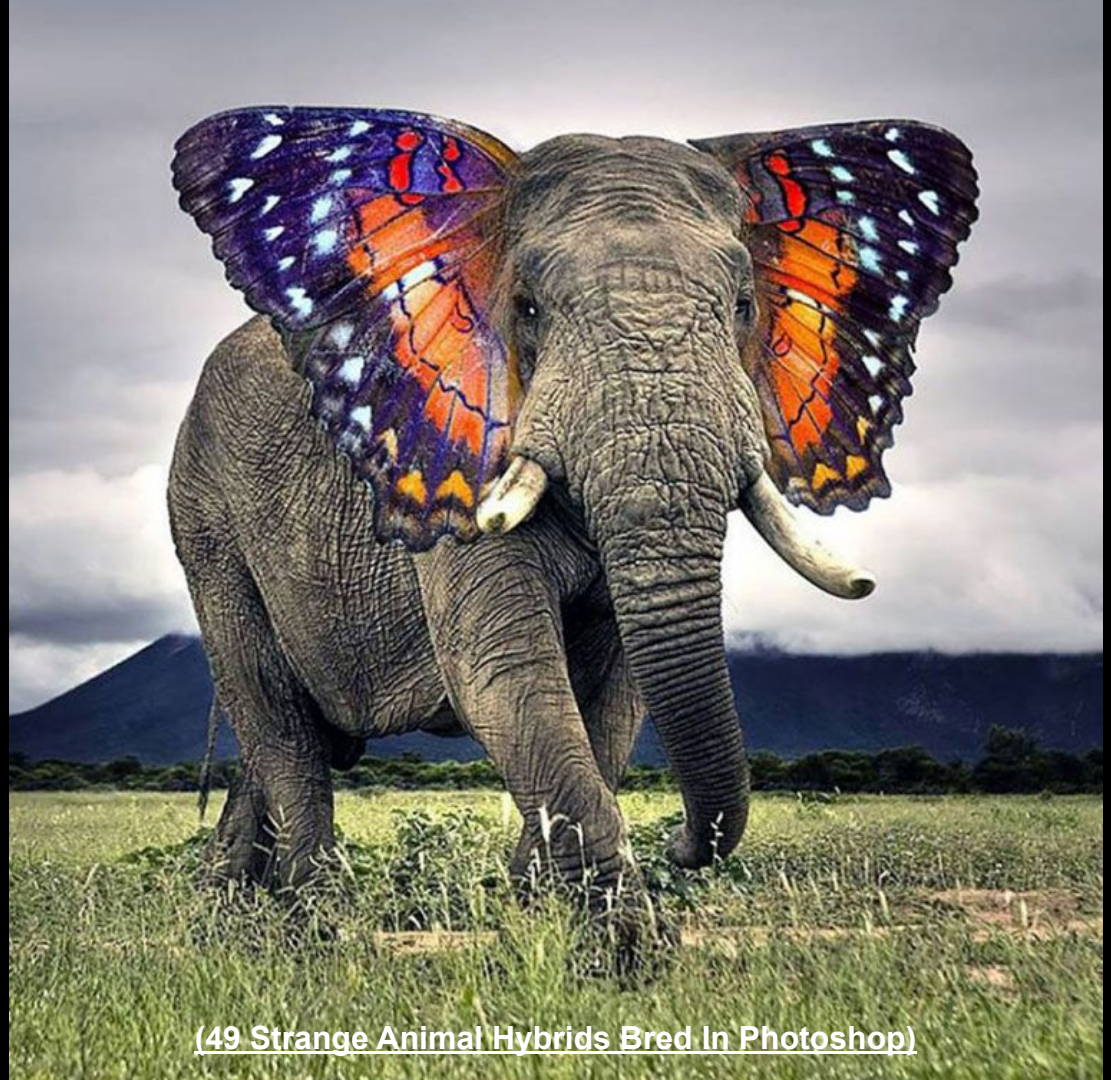
- ✗ Can start only when the VM is down
- ✗ Cannot power up the VM during the backup

What if we could...

- Start and stop a VM during backup
- Start a backup in any VM state
- Migrate a VM during backup
- Take a snapshot during a backup
- Backup on one host, run the VM on another
- Have only one kind of backup

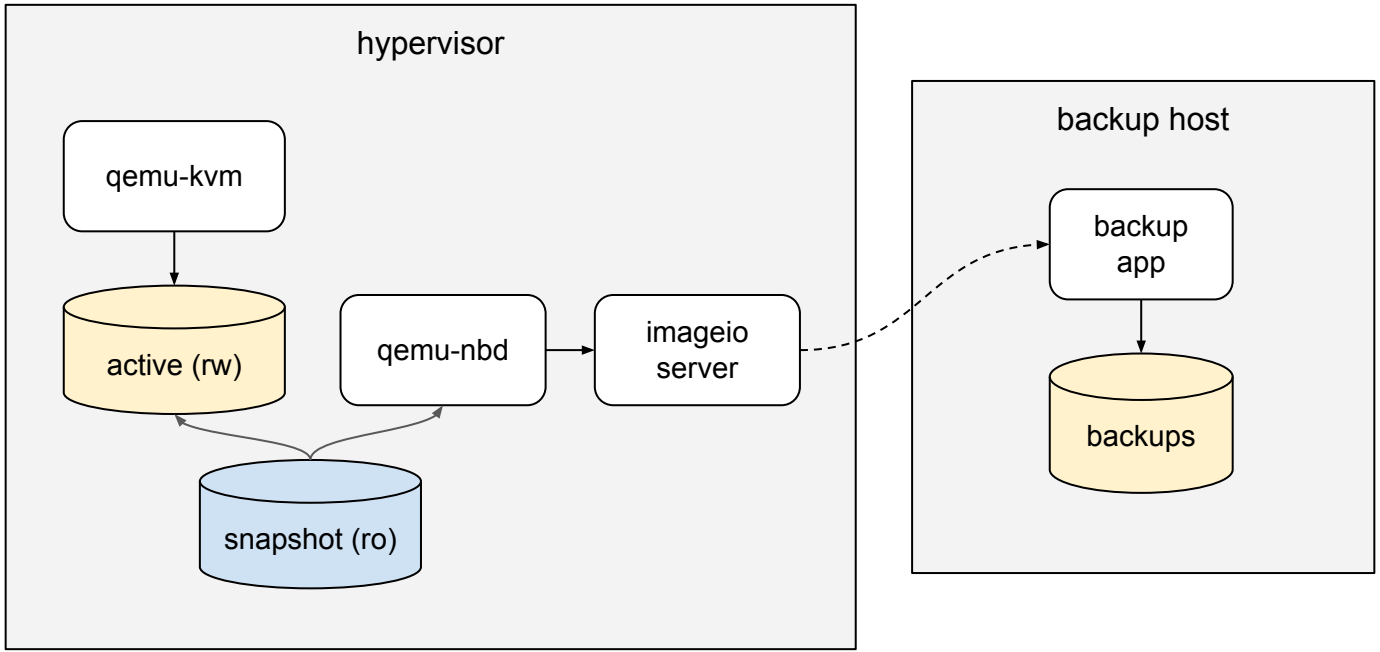
Can we decouple backup from VM life cycle?

Backup a snapshot!

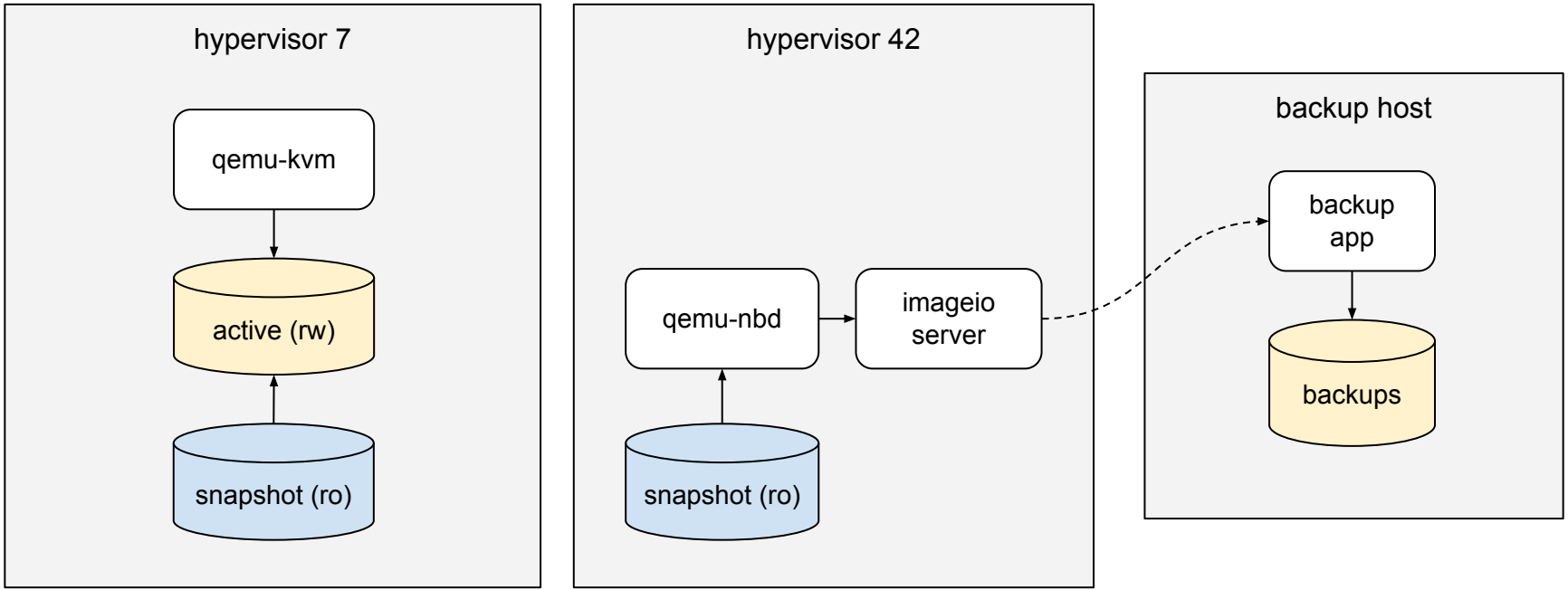


(49 Strange Animal Hybrids Bred In Photoshop)

Hybrid backup (same host)



Hybrid backup (multiple hosts)



Hybrid backup pros and cons

- ✓ Can start, stop, migrate, snapshot a VM during backup
- ✓ Can start backup in any VM state
- ✓ Have only one kind of backup
- ✓ Backup I/O does not affect guest I/O
- ✓ Guest I/O does not affect backup I/O
- ✓ No scratch disks, no pauses
- ✓ Simpler flow on engine side
- ✓ Does not interfere with user snapshots like the old snapshot based backup
- ✗ Need to create and delete a snapshot

Disabling hybrid backup globally

Hybrid backup is enabled by default in in oVirt 4.5. To disable it globally:

```
# engine-config -s UseHybridBackup=false
# systemctl restart ovirt-engine
```

Disabling hybrid backup for a single backup

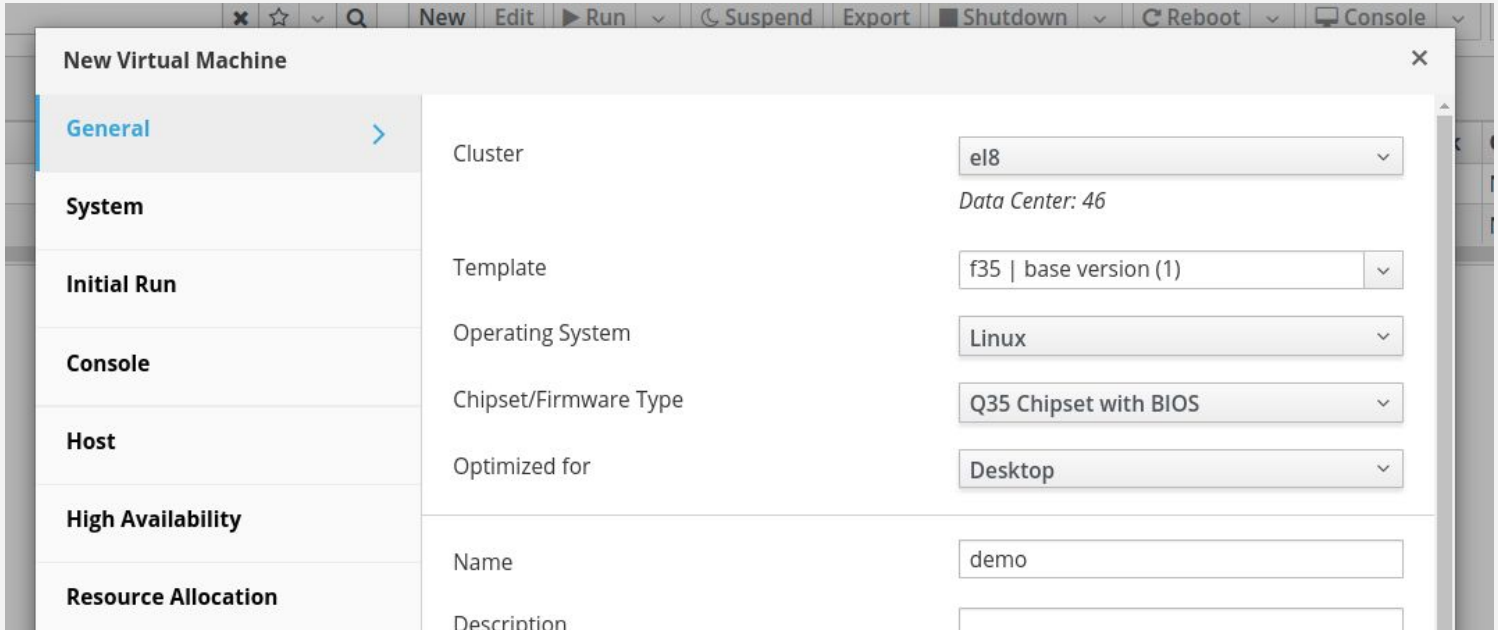
To disable hybrid backup for single backup use the new use_active option:

```
backup = backups_service.add(
  types.Backup(
    disks=disks,
    from_checkpoint_id="d377519f-c0e8-48ea-8731-92db4ab83fba",
  ),
  use_active=True,
)
```

Hybrid Demo

1. Create a new VM

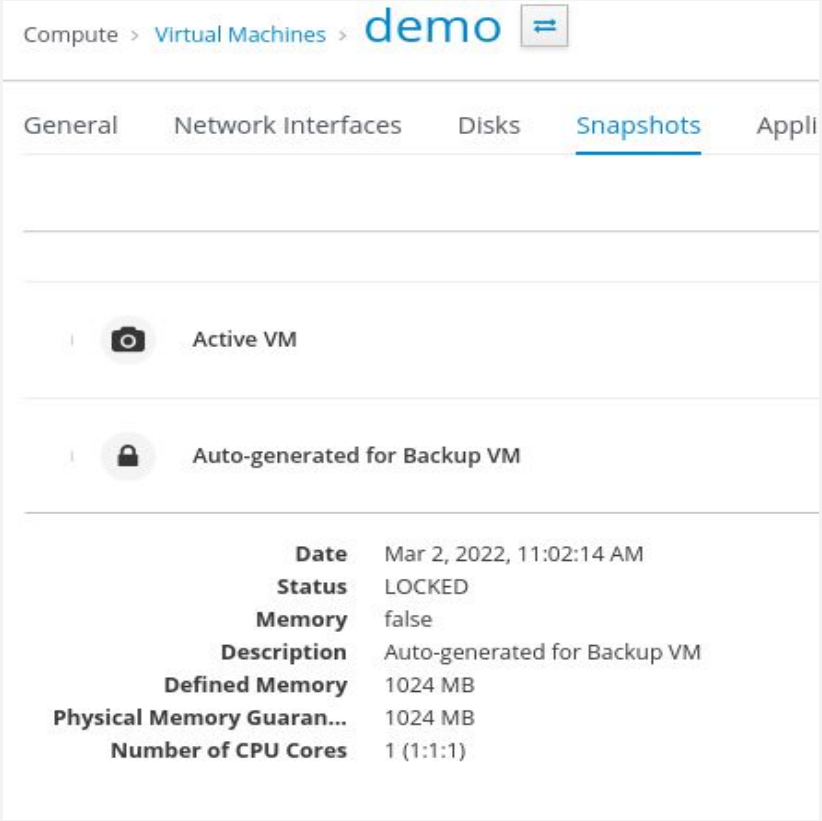
Status: Down



2. Start full backup

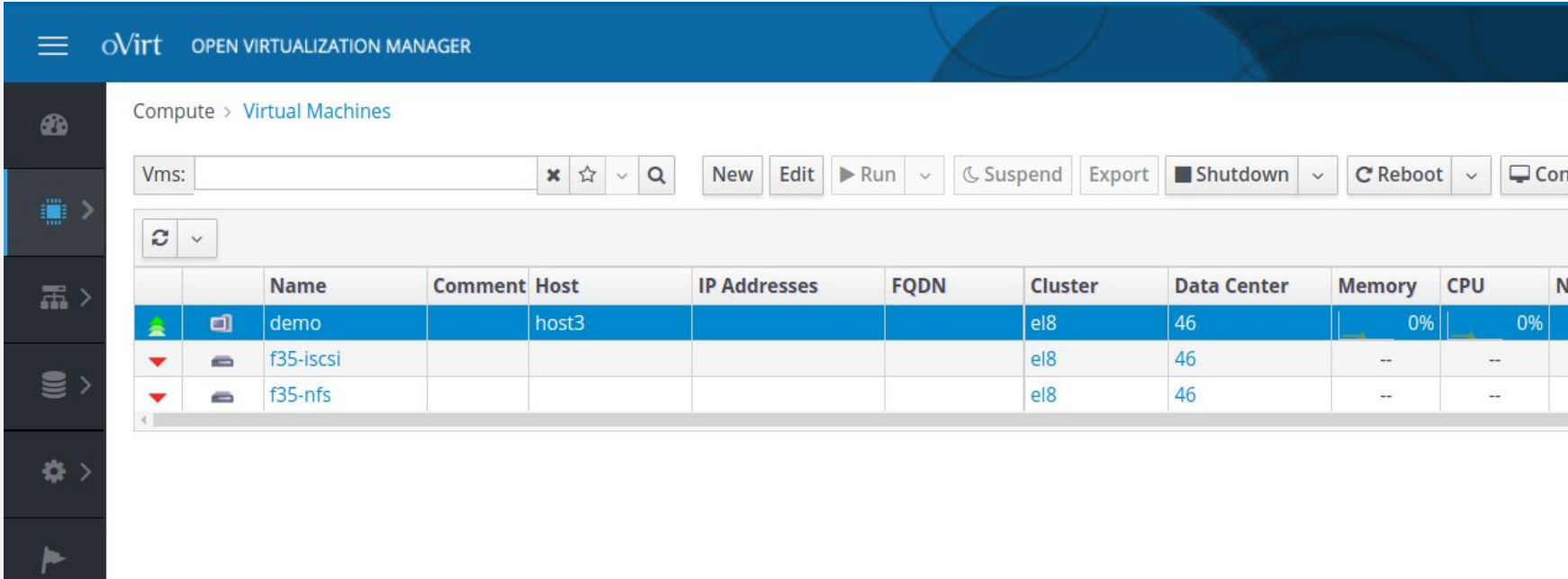
```
$ ./backup_vm.py -c engine start e0bf8ae0-3360-4e0b-9fd8-12cf5fe9839c
[   0.0 ] Starting full backup for VM 'e0bf8ae0-3360-4e0b-9fd8-12cf5fe9839c'
[   0.5 ] Waiting until backup '46d3be69-1ef3-4da2-974d-90bc5d58c42b' is ready
[  11.6 ] Created checkpoint 'a8deec5a-f08c-4cc3-af21-78d2917278c6'
[  11.7 ] Backup '46d3be69-1ef3-4da2-974d-90bc5d58c42b' is ready
```

3. Auto-generated snapshot created



4. Start the VM

Status: `Powering Up`

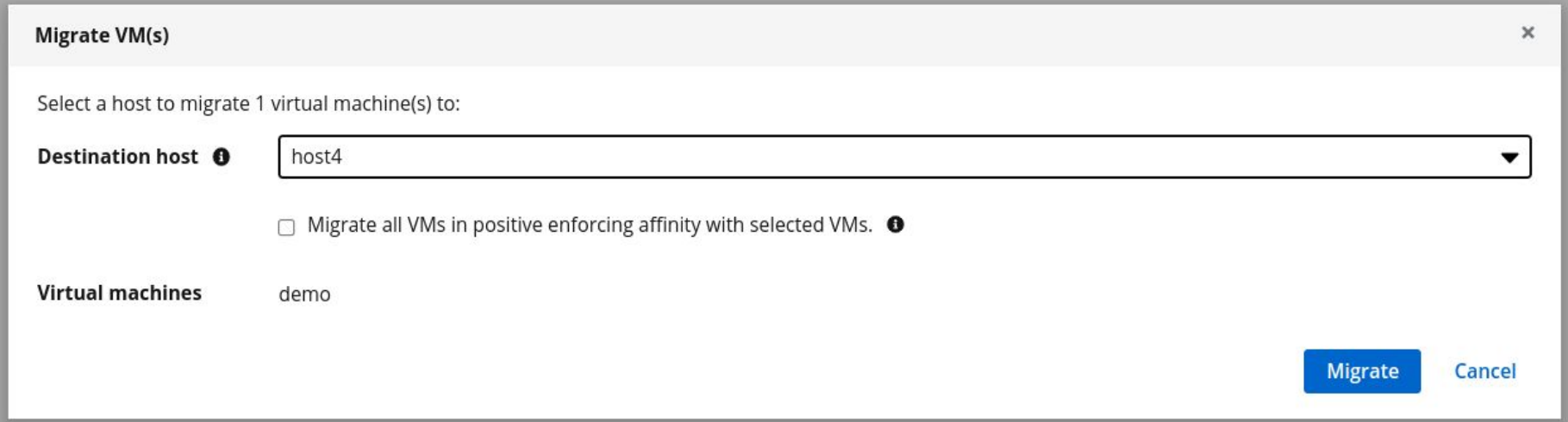


5. Start downloading full backup...

```
$ ./backup_vm.py -c engine download e0bf8ae0-3360-4e0b-9fd8-12cf5fe9839c \
    --backup-dir /data/scratch/backups \
    --backup-uuid 46d3be69-1ef3-4da2-974d-90bc5d58c42b

[   0.0 ] Downloading VM 'e0bf8ae0-3360-4e0b-9fd8-12cf5fe9839c' disks

[   0.1 ] Downloading full backup for disk
'56a62431-624d-4309-af8b-b9b2407dbce5'

[   2.0 ] Image transfer '0eae849f-7451-451a-9727-d8b2d0ac7834' is ready

...
```

6. Migrate VM during backup

Status: `Migrating From`



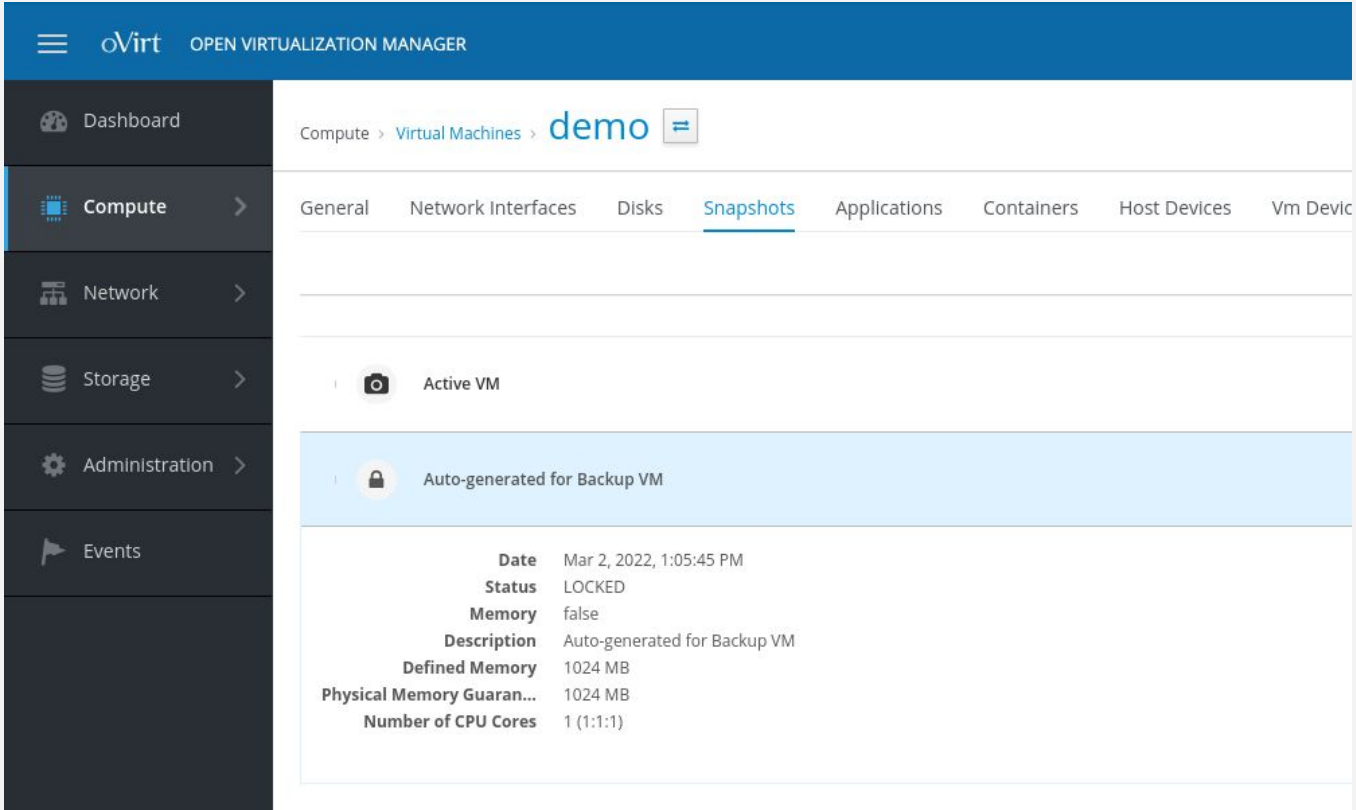
7. Download backup finished

```
...
[ 100.00% ] 6.00 GiB, 3.38 seconds, 1.77 GiB/s
[   5.3 ] Finalizing image transfer
[   8.4 ] Download completed successfully
[   8.4 ] Finished downloading disks
```

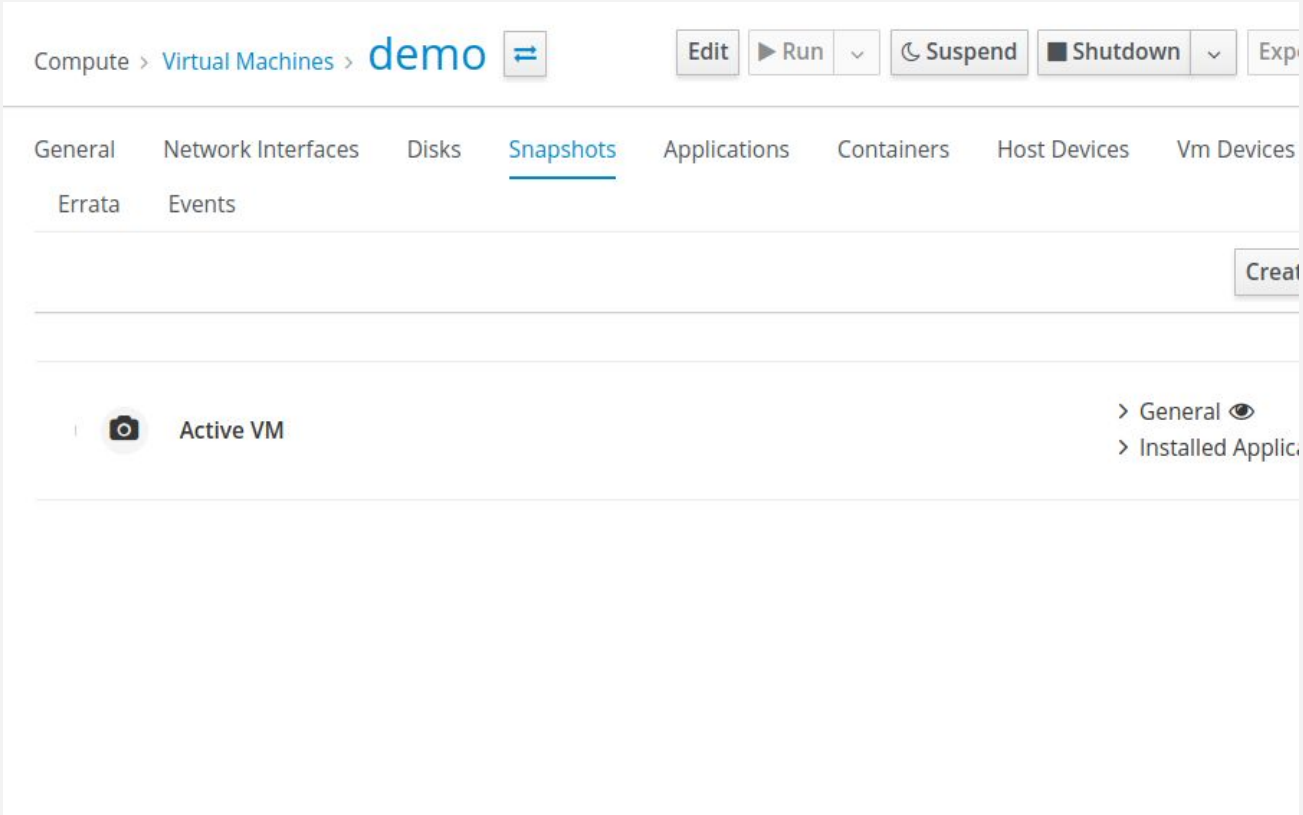
8. Stop the backup

```
$ ./backup_vm.py -c engine stop e0bf8ae0-3360-4e0b-9fd8-12cf5fe9839c \
    46d3be69-1ef3-4da2-974d-90bc5d58c42b

[   0.0 ] Finalizing backup '46d3be69-1ef3-4da2-974d-90bc5d58c42b'

[   7.3 ] Backup '46d3be69-1ef3-4da2-974d-90bc5d58c42b' completed successfully
```

9. Auto-generated snapshot deletion started...



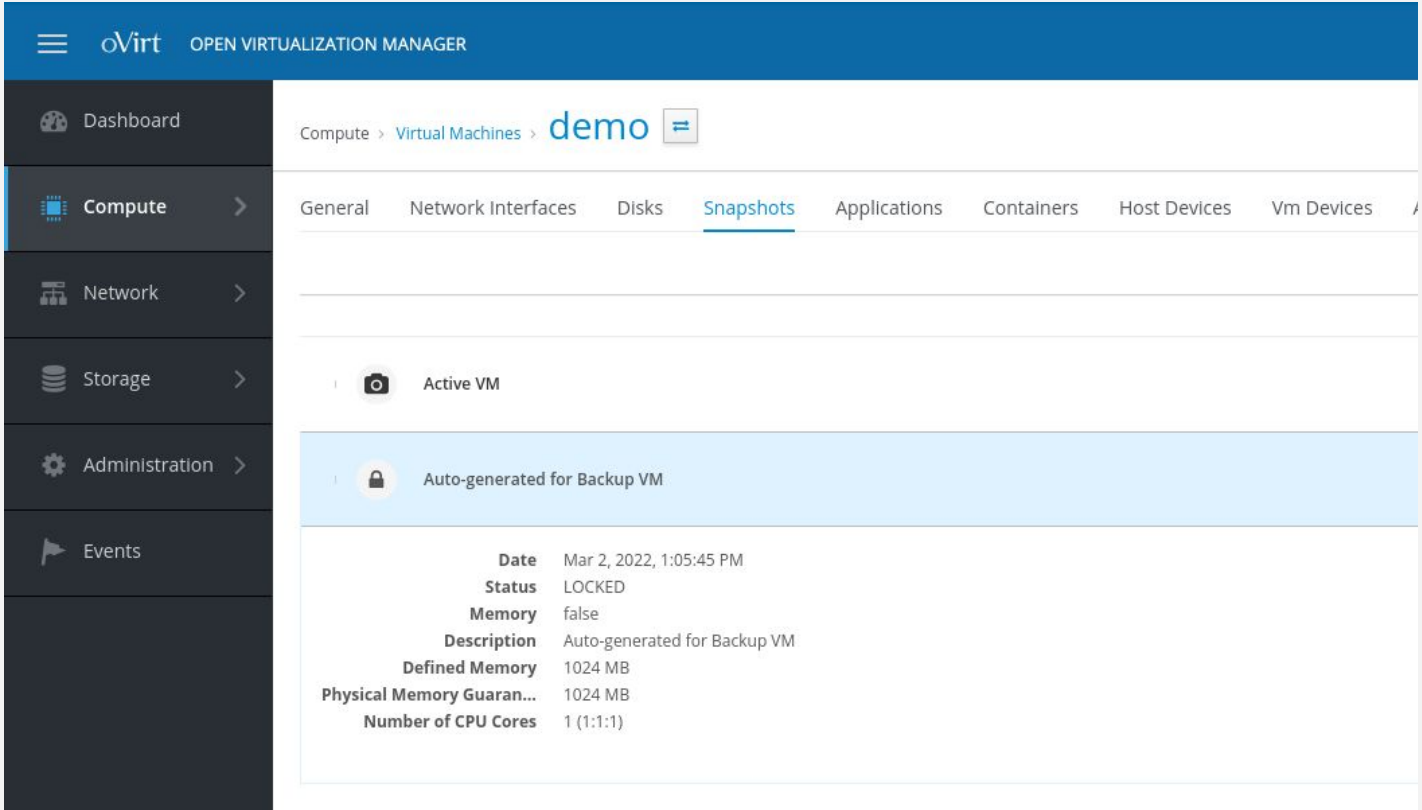
10. Auto-generated snapshot deleted



11. Start incremental backup

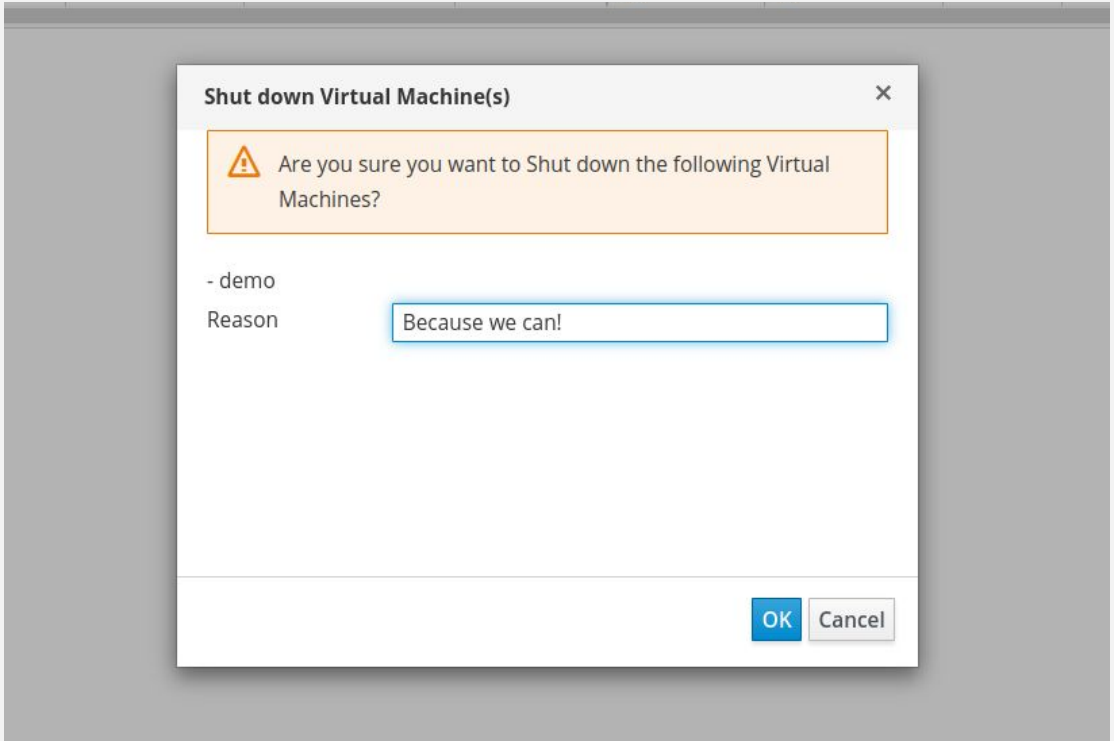
```
$ ./backup_vm.py -c engine start e0bf8ae0-3360-4e0b-9fd8-12cf5fe9839c \
    --from-checkpoint-uuid a8deec5a-f08c-4cc3-af21-78d2917278c6

[   0.0 ] Starting incremental backup since checkpoint
          'a8deec5a-f08c-4cc3-af21-78d2917278c6' for VM
          'e0bf8ae0-3360-4e0b-9fd8-12cf5fe9839c'

[   0.4 ] Waiting until backup '2f27c101-d118-46f9-a82a-735f5a587441' is ready

[   9.5 ] Created checkpoint 'c73f3985-19ff-45bb-a288-4e9bf265c815'

[   9.6 ] Backup '2f27c101-d118-46f9-a82a-735f5a587441' is ready
```

12. Auto-generated snapshot created



13. Stop the VM

Status: `Powering Down`

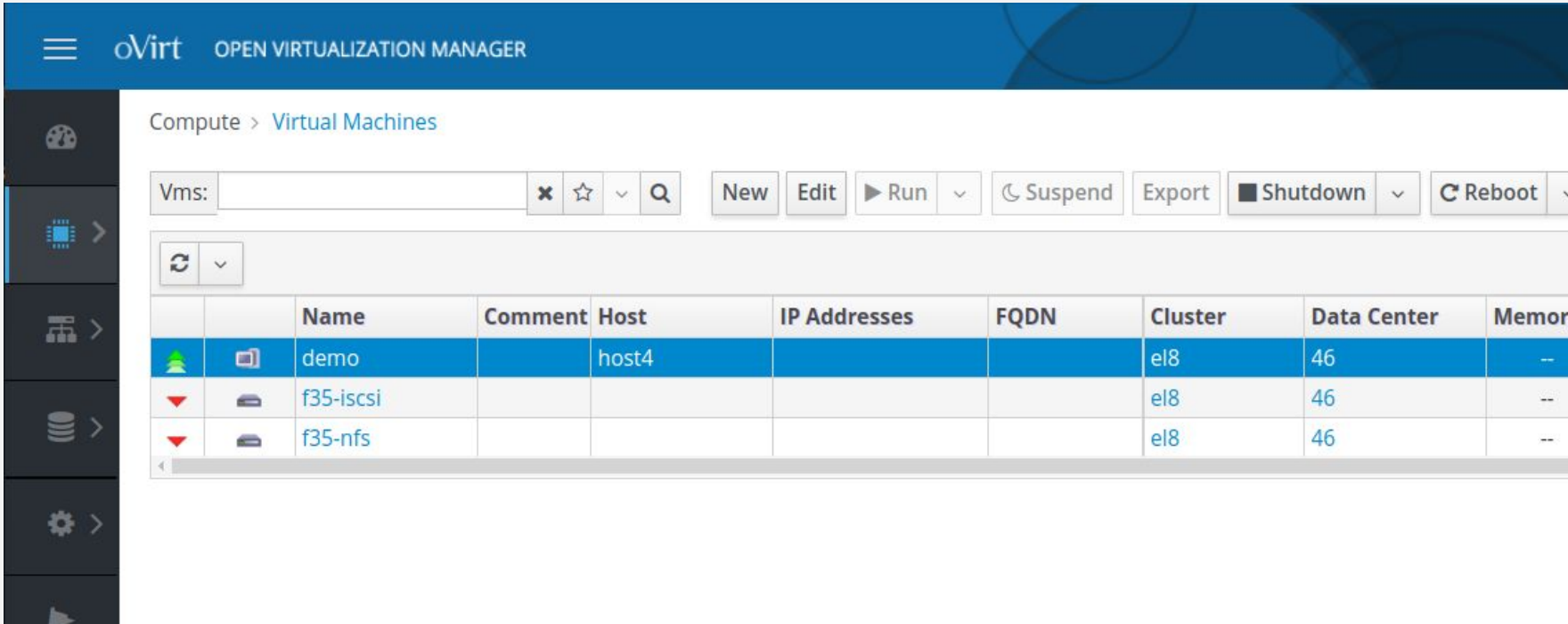


14. Download incremental backup

```
$ ./backup_vm.py -c engine download e0bf8ae0-3360-4e0b-9fd8-12cf5fe9839c \
    --backup-dir /data/scratch/backups \
    --backup-uuid 2f27c101-d118-46f9-a82a-735f5a587441 \
    --incremental

[   0.0 ] Downloading VM 'e0bf8ae0-3360-4e0b-9fd8-12cf5fe9839c' disks

[   0.1 ] Downloading incremental backup for disk '56a62431-624d-4309-af8b-b9b2407dbce5'

[   1.9 ] Image transfer 'b10b448b-0bc2-43be-b683-9c2b91318b31' is ready

[ 100.00% ] 6.00 GiB, 0.62 seconds, 9.69 GiB/s

[   2.5 ] Finalizing image transfer

[   4.5 ] Download completed successfully

[   4.6 ] Finished downloading disks
```

15. Start the VM

Status: `Powering Up`



16. Stop the backup

```
$ ./backup_vm.py -c engine stop e0bf8ae0-3360-4e0b-9fd8-12cf5fe9839c \
    2f27c101-d118-46f9-a82a-735f5a587441

[   0.0 ] Finalizing backup '2f27c101-d118-46f9-a82a-735f5a587441'

[   5.2 ] Backup '2f27c101-d118-46f9-a82a-735f5a587441' completed successfully
```

No virtual machines were harmed

More info

- [RFE] Allow changing vm powerstate during backup operation without interrupting the backup
  https://bugzilla.redhat.com/2053669
- [CBT] [RFE] Extend backup scratch disk as needed
  https://bugzilla.redhat.com/1913387
- [CBT] Provide a way to control backup mode per request
  https://bugzilla.redhat.com/2071744