

Xen

Cheatsheet & quick tips

- [Create NFS Share on Xen8](#)
- [Create a Xen pool \(cluster\)](#)
- [Xen ISO folder](#)
- [Xen New VM](#)
- [Lan Switch](#)
- [Snapshots And Templates](#)

Create NFS Share on Xen8

Add Drive that will be used as NFS share in fstab

```
vi /etc/fstab
```

```
## UUID="XXXXXX" /data01 xfs defaults 0 0
```

Mount it

```
mount -a
```

Add NFS share permissions in exports file

```
vi /etc/exports
```

```
## /data01 10.10.10.0/24(rw,sync,no_root_squash)
```

Edit portmap

```
vi /etc/sysconfig/portmap
```

```
## PMAP_ARGS=""
```

Enable and start services

```
systemctl enable rpcbind
systemctl enable nfs-server
systemctl enable nfs-lock
systemctl enable nfs-idmap

systemctl restart rpcbind
systemctl restart nfs-server
```

```
systemctl restart nfs-lock  
systemctl restart nfs-idmap
```

Allow port

```
vi /etc/sysconfig/iptables
```

```
## -A RH-Firewall-1-INPUT -p tcp -m tcp --dport 111 -j ACCEPT  
-A RH-Firewall-1-INPUT -p tcp -m tcp --dport 2049 -j ACCEPT  
-A RH-Firewall-1-INPUT -p tcp -m tcp --dport 975 -j ACCEPT
```

Restart service

```
systemctl restart iptables
```

Test connection on remote host

```
mount 10.10.10.X:/data01 /mnt/
```

Connect NFS on share SR

```
xe sr-create content-type=user \  
  name-label="data01" shared=true \  
  device-config:server=10.X.X.X device-config:serverpath=/data01 type=nfs \  
  nfsversion="3", "4"
```

Create a Xen pool (cluster)

Connect on master

```
xe pool-list --minimal
```

Create pool with UUID

```
xe pool-param-set name-label="pool.myhypervisor.ca" uuid=<UUID>
```

Edit hostfile with all FQDN's on all nodes

```
vim /etc/hosts
```

```
## masternode.myhypervisor.ca 10.X.X.11
   othernode.myhypervisor.ca 10.X.X.12
```

Connect on other nodes

```
xe pool-join master-address=masternode.myhypervisor.ca master-username=root master-
password=PASSWORD
```

wait 30 sec and check host list

```
xe host-list
```

Xen ISO folder

Find server UUID:

```
xe host-list
```

Create ISO folder

```
mkdir -p /var/opt/xen/ISO
```

Create SR for ISO

```
xe sr-create name-label="Local ISO" type=iso host-uuid=<host-uuid> device-config:location=/var/opt/xen/ISO  
device-config:legacy_mode=true content-type=iso
```

Rescan after downloading a new ISO

```
xe sr-scan uuid=<sr_uuid>
```

check for ISO

```
xe cd-list
```

To remove the SR

Find SR PDB UUID

```
xe pbd-list sr-uuid=<UUID of SR>
```

Unplug SR

```
xe pbd-unplug uuid=<UUID of PBD>
```

Forget SR

```
xe sr-forget uuid=<UUID of SR>
```

Xen New VM

Find ISO SR UUID

```
xe sr-list
```

```
## uuid ( RO ) : XXXXXXXXX-XXXXX-XXX-XXXX-XXXXXXXXXX
name-label ( RW): data01
host ( RO): <shared>
type ( RO): nfs
content-type ( RO): user
```

Find name of template

```
xe template-list
```

```
xe vm-install template="<template_name>" new-name-label=<name_for_vm> sr-
uuid=<storage_repository_uuid>
```

Example

```
xe vm-install template="Ubuntu Bionic Beaver 18.04" new-name-label=test.myhypervisor.ca sr-
uuid=<storage_repository_uuid>
```

Find UUID for VM

```
xe vm-list
```

```
## uuid ( RO ) : XXXXXXXXXXXXXXXXXXXXXXXX
name-label ( RW): test.myhypervisor.ca
power-state ( RO): halted
```

Find ISO UUID

```
xe cd-list
```

```
uuid ( RO) : XXXXXXXXXXXXXXXXXXXXXXXXXXXX
name-label ( RW): ubuntu-18.04.2-live-server-amd64.iso
```

Find network UUID

```
xe network-list bridge=xenbr0 --minimal
```

List virtual drive

```
xe vm-disk-list vm=<vm-uuid>
```

VDB is the ISO assigned on the VM, VDI is the virtual drive

```
“ Disk 0 VBD:
  uuid ( RO) : XXXXXXXXXXXXXXXXXXXXXXXxx
  vm-name-label ( RO): test.myhypervisor.ca
  userdevice ( RW): 0
```

```

Disk 0 VDI:
  uuid ( RO) : XXXXXXXXXXXXXXXXXXXXXXXxx
  name-label ( RW): 0
  sr-name-label ( RO): data01
  virtual-size ( RO): 10737418240
```

Add ISO

```
xe vm-cd-add uuid=<vm-uuid> cd-name=ubuntu-18.04.2-live-server-amd64.iso device=1
xe vm-param-set HVM-boot-policy="BIOS order" uuid=<vm_uuid>
```

Add network on VM

```
xe vif-create vm-uuid=<vm_uuid> network-uuid=<network_uuid> device=0
```

Add VM RAM

```
xe vm-memory-limits-set uuid=<uuid of the valid VM> static-min=<nn>GiB/MiB dynamic-min=<nn>GiB/MiB
dynamic-max=<nn>GiB/MiB static-max=<nn>GiB/MiB
```

Increase VDI


```
xe vdi-resize uuid=<vdi-uuid> disk-size=15GiB
```

Start VM

```
xe vm-start uuid=<vm-uuid>
```

Get id

```
list_domains | grep <vm-uuid>
```

```
3 | XXXXXXXXXXXXXXXX | B H
```

First number is the domid

```
xl console <dom-id>
```

Lan Switch

Create a network for the new interface to be attached to, and returns the network-uuid

```
xe network-create name-label="LAN-VM"
```

Find vm uuid of the VM

```
xe vm-list name-label=<vm's name-label>
```

Create eth1 interface for the VM, and gets the uuid of the new interface

```
xe vif-create vm-uuid=<vm-uuid> network-uuid=<network-uuid> device=1 mac=random
```

You can later list the UUID of the new NIC on that vm:

```
xe vif-list vm-uuid=<vm-uuid>
```

Hot-plug the created interface to the VM

```
xe vif-plug uuid=<new_vif_uuid>
```

To disconnect

```
xe vif-unplug uuid=<vif uuid>  
xe vif-destroy uuid=<vif uuid>
```

To find Mac on a port:

```
xe vm-vif-list params=dhcp.myhypervisor.ca,MAC
```

Snapshots And Templates

Create new snapshot

Find UUID

```
xe vm-list
```

Create snapshot

```
xe vm-snapshot new-name-label="<name>" vm=<vm-uuid>
```

Restore a snapshot

```
xe snapshot-revert snapshot-uuid=<uuid>
```

List snapshots

```
xe snapshot-list
```

Create a template from snapshot

```
xe snapshot-export-to-template snapshot-uuid=<snapshot-uuid> filename="<name>.xva" preserve-power-state=false
```

List SR

```
xe sr-list name-label=<name>
```

Import template in SR

```
xe vm-import filename=<name>.xva sr-uuid=<sr-uuid>
```

Find template UUID

```
xe template-list
```

Create VM from template

```
xe vm-install template-uuid=<template-uuid> new-name-label=<new-name>
```

List template UUID

```
xe template-list name-label="<name>" --minimal
```