

Oracle VM3: Backup und Recovery

Martin Bracher
Trivadis AG
Zürich

Schlüsselworte

OracleVM, XEN, Backup, Recovery, OVMCLI

Einleitung

OracleVM, im Folgenden oft OVM genannt, ist die Virtualisierungslösung von Oracle, mit welcher Oracle in Konkurrenz zu etwa VMWare ESX tritt. OracleVM basiert auf einem Enterprise Linux mit der XEN Technologie zur Virtualisierung. OracleVM bietet zwei grosse Vorteile:

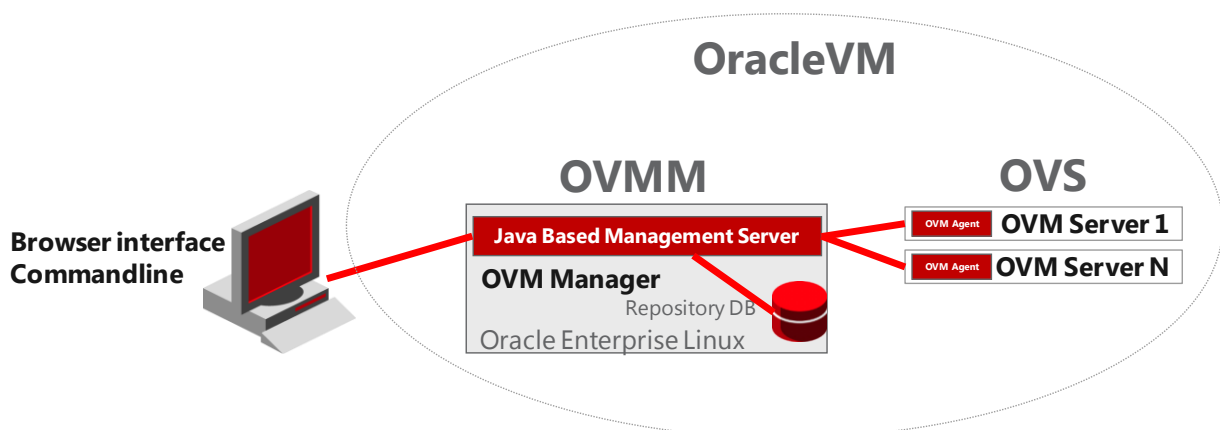
- Zertifizierung und Support für die Oracle Produkte
- Hard-Partitioning: Die Möglichkeit, nur bestimmte CPU-Cores den VM's zuzuordnen, und dann auch nur diese lizenzieren zu müssen

Wie bei jeder anderen Umgebung auch, muss man sich bei OracleVM auch um eine Backup-Strategie kümmern. Die Dokumentation von Oracle liefern hierzu nur sehr rudimentäre Informationen.

Wichtig ist, dass man den Überblick hat, was man eigentlich alles sichern muss. Einerseits natürlich die OVM Infrastruktur selbst (OVM Server, OVM Manager). Auf der anderen Seite die VM's selbst. Diese kann man wie einen physischen Server sichern, aber wir werden in diesem Vortrag auch anschauen, welche zusätzlichen Backup-Möglichkeiten wir mit OVM bekommen.

Übersicht der zu sichernden Komponenten der OVM Infrastruktur

Oracle VM besteht aus zwei Komponenten. Einerseits dem OVM Server, dem Host für die virtuellen Maschinen, und andererseits dem OVM Manager, der Komponente zur Verwaltung des OVM Servers.



Infrastruktur, auf welcher die VM's drauf laufen

- OVM Server Betriebssystem Installation (Disk)

- Pool Filesystem
- Repository Filesystem
- Optional: in VM's verwendete LUN's

Der OVM Manager

- Lokale Disks (OS/Software)
- Repository Datenbank

Die Virtuellen Maschinen (VM's)

- Definition
- Disk-Files
- Daten in der VM

Eine „Komponente“ geht jedoch meist vergessen: Die Installation, bzw. deren Nachvollziehbarkeit. Diese sollte so dokumentiert sein, dass man jederzeit die Umgebung wieder gleich aufbauen kann. Beispielsweise im Fall von Hardware-Defekt, oder Verlust des gesamten Rechenzentrums ist dies sehr hilfreich. Ich empfehle deshalb, den Setup per Skript (via dem OVM Commandline Interface OVMCLI) zu erstellen und die Skripts aufzubewahren.

OVM Server

Hier muss der Host selbst gesichert werden, also das Betriebssystem, welches sich normalerweise auf lokalen Disks befindet.

Es ist jedoch nicht unterstützt, zusätzliche Software auf einem OVM Server zu installieren. Man kann aber lokal vorhandene Tools (tar, gzip, cp, scp, ...) verwenden.

Diesen Backup kann man per scp wegekopieren. Oder auf ein Repository Filesystem ablegen, dieses per NFS freigeben und dann von einer anderen Maschine aus (myBackupHost) abholen. Die Freigabe erfolgt entweder über das OVM Manager Web-Interface, oder per OVMCLI:

```
create RepositoryExport clientHostName=myBackupHost name="MyExport"
repository=MyRepository options="rw, async, no_root_squash" on Server
name=MyOVMServer1
```

Nun kann man den Backup der Filesysteme auf das Storage-Repository „MyRepository“ speichern:

```
BACKUPDIR=/OVS/repository/<uuid>/.backup
tar --one-file-system -zcvf /$BACKUPDIR/root_${HOSTNAME}.tgz /
tar --one-file-system -zcvf $BACKUPDIR/boot_${HOSTNAME}.tgz /boot
```

Praktisch ist, wenn man auch noch den Master Boot Record sichert und den Disklayout speichert:

```
dd if=/dev/sda of=$BACKUPDIR/mbr_${server}-${date +%u}.raw bs=512 count=1
fdisk -l /dev/sda > $BACKUPDIR/fdisk_${server}-${date +%u}.txt
```

Im Problemfall ist es oft von Vorteil, möglichst viel über die Maschine zu wissen. Den Output folgender Tools in einem File zu speichern, kann somit nicht schaden:

```
multipath -ll; lspci; lspci -vvv; df; mount; ifconfig -a
```

Einen zeitgesteuerten Backup muss man z.B. vom OVM Manager aus via ssh starten, da auf dem OVM Server kein cron installiert ist.

Man kann aber auch die Strategie verfolgen, keinen Backup des Servers zu machen, und ihn im Problemfall einfach aus der Konfiguration zu löschen und danach neu zu installieren und wieder in den Pool einzufügen. Wie Anfangs erwähnt, sind hier die Skripts vom initialen Setup sehr hilfreich.

Pool Filesystem

Dies ist das gemeinsame Filesystem eines Cluster-Pools, und ist somit ein Shared Filesystem mit OCFS2. Es befindet sich auf einem SAN, iSCSI oder auf NFS. Wenn es sich auf NFS befindet, ist es eine etwa 12GB grosse Datei, die ein OCFS2 Filesystem enthält und auf den Servern Loopback gemounted ist.

Das OCFS2 Filesystem beinhaltet die Informationen zum Cluster und wird zum Handling der Node-Membership benötigt, sowie zum Speichern, welche VM wo laufen sollte und ein paar weitere Infos. In der Terminologie von Clusterware/RAC hat es die Funktionalität der Cluster-Registry und der Voting Disks.

Dieses Filesystem können wir initial nach der Installation sichern (bei gestopptem Cluster). Bei NFS kopieren und komprimieren der erwähnten Datei, ansonsten mit dem „dd“ Befehl die entsprechende LUN wegekopieren und ebenfalls komprimieren (es bleiben erfahrungsgemäss 10-15 MB übrig).

Stattdessen kann man auch die Erstellungsinformationen des Filesystems festhalten (Befehl 1zeilig).

```
Dev=`mount |grep poolfsmnt|awk '{print $1}'`  
tunefs.ocfs2 -Q "mkfs.ocfs2 -b %B -C %T -J size=4M -N %N -L %V -U %U  
--global-heartbeat --cluster-stack=o2cb --cluster-name=`grep cluster  
/etc/ocfs2/cluster.conf |head -n 1 |awk '{print $3}'`  
--force $Dev\n" $Dev > $BACKUPDIR/mkfs_poolfs_${server}.txt
```

Den Inhalt des gemounteten Filesystems ist in regelmässigen Abständen zu sichern:

```
tar --one-file-system -zcvf $BACKUPDIR/poolfs_${server}-${date +%u}.tgz \  
/poolfsmnt/*
```

Wichtig bei einer Wiederherstellung: die WWID der LUN darf sich nicht ändern (multipath -ll), denn diese ist leider in binären Files noch gespeichert.

Repository Filesystem

Auf diesem Filesystem werden die Virtuellen Maschinen abgelegt (Definition und virtuelle Disks). Es ist ebenfalls ein shared Filesystem zwischen den Servern und ist entweder ein NFS Share, oder ein OCFS2 Filesystem auf LUN oder iSCSI.

Beim Storage-Repository sind folgende Daten zu sichern:

- Informationen zur Erstellung des Filesystems (bei OCFS2)
- Verzeichnis-Struktur

- Statische Files (.ovsrepo, vm.cfg)
- Disk-Files der VM's (konsistente Sicherung!; siehe Sicherung von VM's)

Informationen zur Erstellung des Filesystems:

```
for Dev in `mount |grep /OVS/Repositories/ |awk '{print $1}'`; do
  printf "# " && mount |grep $Dev
  tuneefs.ocfs2 \
    -Q "mkfs.ocfs2 -J block64 -b %B -L %V -U %U -T vmstore -N %N $Dev\n" \
    $Dev
done > $BACKUPDIR/mkfs_repofs_${server}-${date +%u}.txt
```

Regelmässig sollten nun die statischen Dateien gesichert werden: die Definitionsfiles in den Verzeichnissen VirtualMachines und Templates.

Dateien in Assemblies und ISOs sind zwar auch statisch, aber meist sehr gross und sind in der Regel anderweitig wieder herstellbar, so dass wir sie in diesem Backup-Konzept nicht weiter betrachten.

Die Dateien in VirtualDisks (Disk-Images der VM's) sind in der Regel durch die VM's geöffnet und werden laufend beschrieben, so dass hierfür ein spezieller Backup erforderlich ist. Siehe in einem späteren Kapitel.

```
tar --no-recursion -zcf $BACKUPDIR/poolfs_${server}-`basename $mountpoint`-
`date +%u`.tgz `find -type d` .ovsrepo VirtualMachines/*/vm.cfg
Templates/*/vm.cfg
```

Bei Verlust des Filesystems muss dieses also neu erzeugt werden. Es muss auf einer LUN mit derselben WWID wie zuvor erzeugt werden. Danach ist der Restore der Directory-Struktur sowie der (statischen) Files durchzuführen. Zum Schluss sind die Snapshots (siehe später) wieder an den Platz der original Diskdateien zu kopieren.

OVM Manager

Der OVM Manager wird auf einem separaten Server mit RedHat oder Oracle Enterprise Linux installiert.

Es sollte ein Backup der ganzen Filesysteme erstellt werden wie für jeden produktiven Server auch. Bei Verlust können diese Dateien vom Backup zurückgespielt werden.

Sollte kein funktionierender Backup vorhanden sein, kann der Manager (mit Angabe der alten UUID) neu installiert werden, und die Daten durch Re-Discover von den Servern wieder gewonnen werden.

OVM Manager verwendet MySQL als Repository-Datenbank. Oracle erstellt von dieser Repository Datenbank automatisch einen Backup (der dann auch in oben erwähntem Filesystem-Backup enthalten ist).

Virtuelle Maschinen

Eine VM besteht aus 2 Teilen. Erstens dem Storage, also (virtuellen) Disks. Dies kann eine Datei oder ein Block-Device auf dem OVM Server sein oder per Netzwerk ein externer NFS Share oder eine iSCSI LUN.

Eine VM besteht zweitens aus der virtuellen „Hardware“. Diese ist ein Konfigurationsfile gespeichert, in welchem die CPU, Netzwerkkarten, Memory usw. definiert sind. Diese Konfigurationsinformationen befinden sich aber auch in der Repository-Datenbank des OVM Managers. Das Konfigurationsfile wird vom Manager neu geschrieben, falls eine Änderung vorgenommen wird (jedoch nur, wenn die vm.cfg Datei physisch auf dem Server vorhanden ist).

Beim Backup müssen wir unterscheiden zwischen einem Backup der gesamten VM (zwecks „Bare-Metal“ Recovery) , und einem Backup des Inhalts der (virtuellen) Disks (zwecks Restore einzelner Filesysteme/Dateien).

Backup der gesamten VM

Für die virtuelle Hardware müssen wir die Datei <repo-fs>/VirtualMachines/<vm-uuid>/vm.cfg sichern. Diese Datei wird, wie erwähnt, aus den Informationen im Repository erzeugt, aber sie wird zum Restore des Repository-Filesystems benötigt.

Seit OVM ein vernünftiges Commandline Interface zur Verfügung stellt, sollten wir die Befehle zur Erstellung der VM ebenfalls speichern:

```
create Vm name=slot011 repository=slotreposan01 domainType=XEN_PVM
osType=OL_5 bootOrder=DISK cpuCount=2 highAvailability=yes memory=2048 on
ServerPool name=slot
```

```
create Vnic name=00:21:f6:01:00:0b network=vlanpublic
```

```
add Vnic name=00:21:f6:01:00:0b to Vm name=slot011
```

```
create VmDiskMapping slot=0 storageDevice=slot011_xvda name=xvda on Vm
name=slot011
```

Dieses Script würde man brauchen, wenn man die gesamte Umgebung neu aufbauen müsste.

Falls die Disks eine LUN auf dem OVM Server sind, können diese bei gestoppter VM mit dem „dd“ Befehl gesichert werden. Falls der Storage eine Möglichkeit für Snapshots bietet, könnte dies sogar online erfolgen.

Wenn die virtuellen Disks ein File auf dem Storage Repository sind, kann dieses File gesichert werden. Wenn es sich um SAN/iSCSI Storage mit einem OCFS2 Filesystem handelt, können diese Files online durch einen Snapshot gesichert werden, der anschliessend auf ein anderes Medium wegekopiert wird. Der Snapshot selbst ist noch kein Backup, denn er befindet sich immer noch auf demselben Storage.

Wie erstellt man nun einen solchen Snapshot? Man kann aus der VM ein Template klonen, und danach die (nicht geöffneten) Files des Klons wegsichern. OVM bietet die Möglichkeit, das Storage-Repository per NFS zu exportieren. Man erinnere sich: es ist nicht zertifiziert, Backup-Software auf dem OVM Server zu installieren. Aber wir können die Dateien via NFS von einem anderen Server (oder VM) mit installierter Backup-Software aus sichern.

Der Restore einer VM erfolgt dann so, dass man diese stoppt, und danach die Diskfiles der VM durch die gesicherten Snapshot-Files des Klons ersetzt. Danach ist die VM neu zu starten und sie enthält wieder den Zustand vom Zeitpunkt des Snapshots. Dieser Austausch erfolgt direkt auf dem Repository-Filesystem mit Betriebssystem-Kommandos. Alternativ können wir auch die bestehende VM löschen, aus dem Template (unserem Snapshot) eine neue VM klonen, und danach die Netzwerkkarten mit den alten MAC-Adressen wieder hinzufügen.

Backup des Inhalts einer VM

Zum Sichern einzelner Filesysteme oder einzelner Files, sowie von Datenbanken innerhalb der VM unterscheidet sich das Backup-Konzept nicht von dem eines physischen Hosts. Man installiert innerhalb der VM entsprechende Backup-Software und sichert die Dateien auf Storage oder direkt via Netzwerk auf ein Tape.

Ein Problem ist, dass Backup-Clients häufig pro Hostname lizenziert sind. Mit etwas Scripting-Aufwand kann man diese Kosten umgehen: Man erstellt einen Snapshot der Disk-Datei und hängt diesen dann an eine VM mit lizenziertem Backup-Client und macht den Backup von dort aus.

Wenn man wie oben beschrieben jeweils die gesamte VM mit Snapshots sichert, kann man sich den Filesystem-Backup auch sparen und bei Bedarf den Snapshot als zusätzliche Disk temporär hinzufügen und die Dateien von dort zurückkopieren.

```
cloneVdToRepo VirtualDisk name=MyDisk target=MyRepo cloneType=THIN_CLONE  
create VmDiskMapping slot=3 VirtualDisk="MyDisk (1)" name=xvdc on vm  
name=vm001
```

Im Oracle-Bereich können wir Snapshots auch nutzen. Das Zeitaufwändige beim Wiederherstellen einer Datenbank ist der Restore, das anschliessende Recovery ist verhältnismässig schnell. Wenn also die verlorenen/korrupten Datafiles auf einem virtuellen Disk liegen, von dem wir einen Snapshot haben, können wir diesen Disk durch den Snapshot austauschen und danach das Recovery laufen lassen.

Fazit

Für die OVM Server steht uns keine fertige Backup-Lösung bereit, aber mit ein paar Scripts kommen wir zu einer schlanken und effizienten Lösung.

Im Prinzip müssen sich die Backup-Konzepte von physischen Servern und VM's unter OVM praktisch nicht unterscheiden. Der (virtuelle) Server wird mit klassischen Backup-Clients gesichert. Wenn man nun einen physischen Server oder eine VM verliert, besorgt man sich einen neuen physischen/virtuellen Server und installiert diesen neu bzw. macht einen Restore des Backups.

OVM bietet uns aber einige zusätzliche Möglichkeiten, um das Backup-Konzept zu optimieren oder zu erweitern, wie beispielsweise die Möglichkeit für Snapshots ganzer Diskfiles.

Der Verlust des OVM Managers ist kein grosses Problem. Die Server und VM's laufen unabhängig von diesem weiter, und der Manager kann aufgrund der Informationen auf den Servern weitgehend wieder neu aufgebaut werden.

Viel Erfolg beim Einsatz von Trivadis-Know-how wünscht Ihnen

Kontaktadresse:

Martin Bracher
Trivadis AG
Sägereistrasse 29
CH-8152 Glattbrugg

Telefon: +41 58 459 56 56
Fax: +41 58 459 56 66
E-Mail: martin.bracher@trivadis.com
Internet: www.trivadis.com