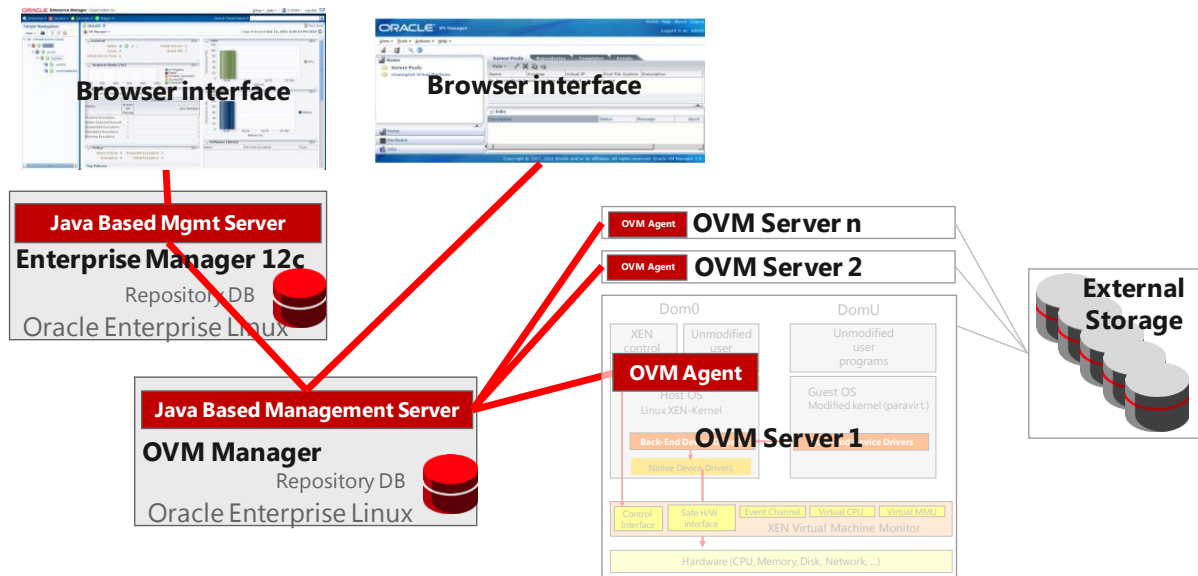


Oracle VM 3 Manager als virtuelle Maschine auf dem Oracle VM 3 Server

Martin Bracher
Senior Consultant
26.12.2011



Oracle VM – die Virtualisierungslösung von Oracle – 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.



Während es für Version 2 von Oracle ein Template für den OVM Manager gab, bietet uns Oracle diese Möglichkeit leider nicht mehr für die Version 3. Die Frage ist nun, wie können wir den Manager wieder als VM auf dem Server laufen lassen?

1. Ausgangslage

Normalerweise installieren wir auf unserem Virtualisierungsserver den OVM Server. Dies ist eine minimale Linux-Distribution, die alle zum Betrieb eines Virtualisierungshosts notwendigen Pakete mitbringt. Bei der Installation lässt sich nur das absolut Notwendige konfigurieren. Die weitere Konfiguration wird dann zentral vom OVM Manager aus erledigt.

Der OVM Manager ist eine in Weblogic integrierte Web-Applikation mit einem Repository in einer Oracle Datenbank. Die Installation erfolgt in ein bestehendes Oracle/RedHat Enterprise Linux System. Bei kleinen Umgebungen, insbesondere wenn wir nur einen einzelnen Virtualisierungsserver haben, ist eine eigene physische Maschine für den Manager wohl etwas übertrieben.

Die Frage ist nun, wie kriegen wir dieses System als VM auf den Server?

Technisch ist es etwas anspruchsvoller als früher, da nicht mehr vorgesehen ist, dass man den Server von Hand administriert, sondern nur noch in Kombination mit dem Manager. Also



wieder ein bisschen das "Henne-Ei"-Problem: wir müssen zuerst den Server mit dem Manager konfigurieren, bevor wir VM's benutzen können, aber die VM muss ja schon laufen. Was können wir also tun? Hier gibt es verschiedene Möglichkeiten:

- Verwendung einer zweiten, physischen Maschine, und diese nach der Konfiguration in eine VM konvertieren.
- Verwendung einer VM auf einem schon bestehenden (alten) OVM Server.
- Temporäre Verwendung einer VM auf dem neuen OVM Server.

Im weiteren Verlauf werden wir uns die letzte Variante anschauen.

2. Implementierung

Da auf OVM3 Server kein Tool zur Erstellung einer VM mehr vorhanden ist, verwenden wir am Einfachsten ein fertiges Template von Oracle (edelivery.oracle.com/linux), beispielsweise das OVM_OL5U6_X86_64_PVM_10GB, welches vom Download her nur etwa 700MB ist, aber nach dem Entpacken eine Disk von 10GB bietet, was für unsere Zwecke ausreichen sollte. Platz auf dem Server haben wir ja wahrscheinlich ausreichend, da Oracle nur wenige GB der Boot-Disk konfiguriert und den Rest frei lässt. Formatieren und Mouten wir uns vorübergehend diesen Platz

```
fdisk #add partition for the unused space, usually partition 4
mkfs.ext3 /dev/sdaX #X=number of new partition
mount /dev/sdaX /mnt
```

und kopieren per scp das Template dorthin und entpacken es. Ebenso kopieren wir das ISO-Image vom OVM Manager dorthin.

```
cd /mnt/
scp user@downloadhost:OVM_OL5U6_X86_64_PVM_10GB.tgz .
scp user@downloadhost:OracleVM-Manager-3.0.2.iso .
tar zxvf OVM_OL5U6_X86_64_PVM_10GB.tgz
```

Falls demnächst ein Template für Enterprise Manger 12c erhältlich ist, werden Sie auch dieses verwenden können für die ovm Manager Installation. Dieses Template können Sie dann auch gleich für die Integration vom Manger in EM12c verwenden.

Das vm.cfg muss man sich nun sinngemäss von Hand anpassen (wir haben ja noch keinen Manger, mit welchem wir das Template clonen könnten...)

```
bootloader = '/usr/bin/pygrub'
disk = ['file:/mnt/OVM_OL5U6_X86_64_PVM_10GB/System.img,xvda,w'
       , 'file:/mnt/OracleVM-Manager-3.0.2.iso,xvdb,r' ]
vif = ['mac=00:21:f6:00:00:01,bridge=xenbr0']
memory = '4096'
name = 'OVM_OL5U6_X86_64_PVM_10GB'
vcpus = 2
on_crash = 'restart'
on_reboot = 'restart'
#vfb = ['type=vnc,vncunused=1,vnclisten=0.0.0.0']
```

- Die MAC-Adresse sollte mit 00:21:f6 beginnen. Dies ist der Bereich, den der Manager später verwalten kann.
- Das ISO-Image mit der Software stellen wir als /dev/xvdb zum Mouten bereit.
- Die virtuelle Grafikkarte (vfb) können wir deaktivieren, die Installation der Software kann auf der Textkonsole erfolgen.



- Für das Netzwerk verwenden wir die Bridge xenbr0. Um nicht in Konflikt mit der Netzwerk-Konfiguration des Managers zu kommen, verwenden wir temporär ein Dummy-Interface:

```
modprobe dummy  
brctl addbr xenbr0  
brctl addif xenbr0 dummy0  
ifconfig dummy0 192.168.11.1
```

Danach können wir manuell die VM starten, welche uns dann gleich mit der Textkonsole verbindet:

```
xm create -c /mnt/OVM_OL5U6_X86_64_PVM_10GB/vm.cfg extra="console=xvc0"
```

Einloggen mit "root", Passwort "ovsroot".

Falls nicht schon beim Startup die Konfiguration abgefragt wurde, können wir diese nun aufrufen. Als IP verwenden wir in diesem Beispiel 192.168.11.2:

```
oraclevm-template --config --force
```

Die Installation und Konfiguration von OVM Manager machen wir nun von diesem System aus.

```
mount /dev/xvdb /mnt  
/mnt/createOracle.sh  
/mnt/runInstaller.sh
```

Wenn die Installation erfolgreich durchgelaufen ist, können wir die Konfiguration starten. Da unsere virtuelle Maschine nur an diesem lokalen dummy0 Netzwerkinterface hängt, und auf dem OVM Server kein Browser verfügbar ist, müssen wir initial von unserer Arbeitsstation aus via SSH-Tunnel darauf zugreifen (Windows-Clients: PuTTY).

```
ssh -L 7001:192.168.11.2:7001 ovmserver
```

Vom Browser unserer Arbeitsstation aus verbinden wir uns nun mit <http://localhost:7001/ovm/console> und können nun den Manager, als auch den Server wie gewohnt konfigurieren.

Tipp: Wer für einen Standalone-Server den freien Platz der lokalen Disk verwenden möchte, kann diesen über NFS exportieren und so im Manager konfigurieren.

Das nächste Problem ist nun, wie verschieben wir diese VM auf unseren neuen Server und machen dies im Manager bekannt. Um diese Änderung machen zu können, muss der Manager ja laufen. Es ist deshalb ein Trick notwendig. Wir erstellen von unserer OVM Manager VM aus, die sich selbst noch gar nicht kennt, eine neue VM. Diese soll die gleiche virtuelle "Hardware" haben, wie unsere bestehende VM, d.h. dieselbe Anzahl Disks, Memory, CPU, MAC-Adresse usw. Diese VM brauchen wir nur zu erstellen, wir brauchen nichts darin zu installieren.

Stattdessen stoppen wir die VM und ersetzen das leer erstellte Diskfile (OVS/Repositories/*/VirtualDisks/*.img) durch das File unserer bisherigen Manager VM (/mnt/OVM_OL5U6_X86_64_PVM_10GB/System.img). Das Einzige was wir vor dem Stoppen noch tun müssen ist, die Netzwerkkonfiguration anzupassen (/etc/sysconfig/network-scripts/ifcfg-eth0; IP-Adresse statisch setzen, oder DHCP aktivieren).

Jetzt können wir die VM stoppen (init 0), und danach mit der neuen, vom Manager generierten Konfiguration wieder starten. Diese befindet sich unter /OVS/Repositories/*/VirtualMachines/*/vm.cfg; Damit die VM gleich beim Booten des Servers gestartet wird, können wir einen Symlink darauf in /etc/xen/auto/ erstellen.

Der Zugriff auf den Manager erfolgt nun direkt über Netz, der Trick mit dem SSH-Tunnel ist nicht mehr erforderlich.



3. Fazit

Mit geringem Aufwand und etwas Kreativität ist es doch möglich, den Manager wieder auf dem Server selbst laufen zu lassen.

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

Martin Bracher

Trivadis AG

Europa-Strasse 5

CH-8152 Glattbrugg

Internet: www.trivadis.com

Tel: +41-31 928 09 60

Fax: +41-31 928 09 64

Mail: info@trivadis.com

Literatur und Links...

www.trivadis.com

edelivery.oracle.com/linux