

Criação de Máquinas Virtuais utilizando o VirtualBox para Acesso Remoto

Fernando Henrique Gielow*

14 de fevereiro de 2011

Sumário

1	Instalando o VirtualBox	2
1.1	Com apt-get	2
1.2	Com o yum	2
2	Máquina Virtual para Acesso Remoto	3
2.1	Criando a Máquina Virtual	3
2.2	Criando Réplicas da Máquina Virtual	4
2.3	Iniciando e Parando a Máquina Virtual	5
3	Acesso Remoto através do rdesktop	6
3.1	Acesso com a Porta Bloqueada	6

*Para contato, email: fhg07@inf.ufpr.br

1 Instalando o VirtualBox

Esta Seção detalha a instalação da versão *non-free* do VirtualBox [1]. Note que na máquina *baco* do laboratório do NR2, o VirtualBox já está instalado corretamente.

1.1 Com apt-get

Adicione à `/etc/apt/sources.list` uma das seguintes linhas, que corresponda à sua distribuição:

```
deb http://download.virtualbox.org/virtualbox/debian karmic non-free
deb http://download.virtualbox.org/virtualbox/debian jaunty non-free
deb http://download.virtualbox.org/virtualbox/debian hardy non-free
deb http://download.virtualbox.org/virtualbox/debian gutsy non-free
```

Para registrar o repositório, utilize:

```
wget http://download.virtualbox.org/virtualbox/debian/sun_vbox.asc;
sudo apt-key add sun_vbox.asc;
```

Então atualize, e instale a versão mais recente do VirtualBox do repositório (no caso, está sendo exemplificado com a versão 3.1):

```
sudo apt-get update;
aptitude search VirtualBox;
sudo apt-get install virtualbox-3.1;
```

1.2 Com o yum

Basta adicionar o repositório *non-free*, e instalar a versão mais recente do VirtualBox. No caso, vamos exemplificar com a versão 3.1.x86_64:

```
wget http://download.virtualbox.org/virtualbox/rpm/fedora/virtualbox.repo;
sudo mv virtualbox.repo /etc/yum.repos.d/;
yum search virtualbox;
sudo yum install VirtualBox-3.1.x86_64;
```

2 Máquina Virtual para Acesso Remoto

2.1 Criando a Máquina Virtual

No exemplo abaixo, uma máquina virtual de nome *minha_maquina*, com 1024MBs de RAM, 10000MBs de HD virtual será criada. Inicialmente, a máquina virtual é registrada, em seguida, um HD para ela é criado e "conectado" à máquina virtual. A máquina bootará com a iso cujo caminho está especificado em *\$iso_address*. Esta deve ser a iso de instalação bootável do sistema operacional a ser utilizado. Além disso, *\$access_port* será a porta utilizada para o acesso remoto. Utilize portas livres, de preferência com valor alto. Você pode utilizar o comando `netstat` para saber que conjuntos de portas não estão sendo utilizadas. Após a configuração da máquina, o último comando iniciará ela para que a instalação do SO seja feita. Prossiga normalmente com a instalação, e desligue a máquina após ela concluída.

```
export nome="minha_maquina"; #nome pelo qual sua máquina será identificada
export mem="1024"; #memoria ram da máquina hospedeira a ser alocada para a MV
export hd_cap="10000"; #quantos MBs serão utilizados pelo hd virtual da MV
export access_port="10111"; #por que porta a MV será acessada remotamente
export iso_address="/home/fernando/ubuntu-9.10-desktop-i386.iso"; # caminho \
    # para a iso de instalação do sistema operacional a ser sado na MV

VBoxManage createvm --name "$nome" --register;
VBoxManage createhd --filename "$nome.vdi" --size $hd_cap --remember;
VBoxManage storagectl "$nome" --name "ide controller" --add ide;
VBoxManage storageattach "$nome" --storagectl "ide controller" --port 0 \
    --device 0 --type hdd --medium "$nome.vdi";
VBoxManage storageattach "$nome" --storagectl "ide controller" --port 1 \
    --device 0 --type dvddrive --medium $iso_address
VBoxManage modifyvm "$nome" --memory "$mem" --acpi on --boot1 dvd \
    --nic1 nat --vrdp on --vrdpport $access_port;

VBoxManage startvm "$nome";
```

Realizada a instalação do SO, é necessário também instalar a ferramenta VBoxGuestAdditions, que integra de maneira melhor o ambiente virtual criado à ferramentas de visualização remota (protocolo rdp), por exemplo. Para isso, vamos emular a iso contida em *\$path*, e iniciar a máquina, para que a instalação das ferramentas seja realizada. Após isso, desligue a máquina novamente.

```
export nome="minha_maquina"; #nome pelo qual sua máquina será identificada
export path="/usr/share/virtualbox/VBoxGuestAdditions.iso"; #caminho ate a iso
#das ferramentas adicionais da VirtualBox (o caminho já está certo, para o
#caso da maioria das distribuições linux)
```

```
VBoxManage storageattach "$nome" --storagectl "ide controller" --port 1 \
--device 0 --type dvddrive --medium $path;
VBoxManage startvm "$nome";
```

Após a máquina iniciada, é só acessar o conteúdo da .iso emulada, rodando o binário do instalador correto para o sistema operacional sendo utilizado. Abaixo temos um exemplo para um sistema operacional linux de 32 bits.

```
cd /media/cdrom;
sudo ./VBoxLinuxAdditions-x86.run;
sudo shutdown -h 0;
```

2.2 Criando Réplicas da Máquina Virtual

Para não precisar realizar a instalação e devida configuração do sistema operacional diversas vezes, é possível clonar¹ o "HD" de uma instalação válida. Para isso, os passos de criação da máquina virtual podem ser repetidos, com a diferença de que não é criado um "HD" novo, mas sim, clonado. Note que as máquinas virtuais **devem** possuir tanto "HDs" quanto portas de acesso diferentes. Utilize números de portas de acesso diferentes para cada máquina virtual! Abaixo temos um exemplo de um script que realizará 4 réplicas da máquina virtual de nome *minha_maquina*. As réplicas terão nome *minha_maquina-clone.\$i*, aonde *\$i* é o índice da réplica. As portas de acesso utilizadas pelas réplicas serão a porta de acesso utilizada pela máquina original, incrementando o índice da réplica. O código-exemplo se encontra na próxima página.

¹**não utilizar** o comando *cp*, uma cópia normal não funcionará!

```

export nome="minha_maquina"; #nome pelo qual sua máquina será identificada
export mem="1024"; #memoria ram da máquina hospedeira a ser alocada para a MV
export hd_cap="10000"; #quantos MBs serão utilizados pelo hd virtual da MV
export access_port="10111"; #por que porta a MV será acessada remotamente
export iso_address="/home/fernando/ubuntu-9.10-desktop-i386.iso"; # caminho \
    # para a iso de instalação do sistema operacional a ser sado na MV
export nome_CL="minha_maquina-clone"; #nome dos clones a serem gerados

for i in {0..3} #quantos clones? trocar o 3 pela quantidade desejada
do
    access_port_CL=$(echo "$access_port + $i" | bc -l);
    VBoxManage createvm --name "$nome_CL.$i" --register;
    VBoxManage clonehd "$nome.vdi" "$nome_CL.$i.vdi"; #clonar HD
    VBoxManage storagectl "$nome_CL.$i" --name "ide controller" --add ide;
    VBoxManage storageattach "$nome_CL.$i" --storagectl "ide controller" --port 0 \
        --device 0 --type hdd --medium "$nome_CL.$i.vdi";
    VBoxManage storageattach "$nome_CL.$i" --storagectl "ide controller" --port 1 \
        --device 0 --type dvddrive --medium $iso_address;
    VBoxManage modifyvm "$nome_CL.$i" --memory "$mem" --acpi on --boot1 dvd \
        --nic1 nat --vrdp on --vrdpport $access_port_CL;
done

```

2.3 Iniciando e Parando a Máquina Virtual

Para iniciar a máquina virtual em modo Headless (em background, sem que nenhuma interface apareça diretamente) para acesso remoto, utilizar:

```
VBoxManage startvm "$nome" -type vrdp;
```

O recomendável para terminar a execução da máquina virtual é desligar ela normalmente. Porém, caso se deseje forçar o término da máquina virtual, utilizar:

```
VBoxManage controlvm "$nome" poweroff;
```

3 Acesso Remoto através do rdesktop

Para realizar o acesso remoto, utilizaremos o cliente **rdesktop** [2], da seguinte forma:

```
rdesktop -x -z IP_SERVER_MVS:PORTA_ACESSO
```

Caso se deseja fazer um acesso local, é só utilizar IP_SERVER_MVS como sendo *localhost*.

3.1 Acesso com a Porta Bloqueada

É comum que a porta de acesso especificada para o acesso remoto esteja fechada em algum roteador ou firewall. Como normalmente os firewalls e roteadores mantêm uma porta para SSH livre, é possível redirecionar a porta que deveria ser utilizada pelo SSH, através de um "túnel". Para isso:

```
server=IP_SERVER_MVS; #IP do servidor aonde o VirtualBox roda
ssh_port=22; #alterar caso outra porta seja utilizada
access_port=PORTA_ACESSO; #porta de acesso para o uso da MV
user=USUARIO_LOGIN; #usuário de acesso no servidor
ssh -L localhost:$access_port:$server:$access_port \
    -p $ssh_port $user@$server;
```

Obs.: Note que este SSH apenas redireciona a PORTA_ACESSO do servidor para a PORTA_ACESSO do *localhost*. Ou seja, para realizar o acesso após o SSH realizado, utilize o seguinte comando em um terminal **local**:

```
rdesktop -x -z localhost:PORTA_ACESSO
```

Obs2.: Caso o uso das máquinas virtuais locais seja para pessoas externas e se tenha que criar um túnel SSH, é bom criar um usuário diferente e sem nenhum privilégio exclusivo para este fim na máquina servidora, aonde as máquinas virtuais são hospedadas.

Referências

- [1] VirtualBox, April 2010. [Online]. Available: <http://www.virtualbox.org/>
- [2] rdesktop, April 2010. [Online]. Available: <http://www.rdesktop.org/>