**Download and Install latest version of Oracle VirtualBox**

<https://www.oracle.com/technetwork/server-storage/virtualbox/downloads/index.html>

**Download latest version of Oracle Linux:**

<http://edelivery.oracle.com> Search for: **REL: Oracle Linux 7**

**Create a new machine**

Memory Size: 2 - 4 GiB

Virtual Hard Disk Size: 16 - 24 GiB

**Note:** The minimal sizes will work for running the OS, but if you will be installing Databases memory may need to be higher and I would go with the higher disk size, and then add additional virtual disks to support software or databases.

**Update the following after the new VM is created but before you start the machine:**

General > Advanced > Shared Clipboard: Bidirectional

General > Advanced > Drag'n'Drop: Bidirectional

System > Motherboard > Pointing Device: USB Tablet

Network > Adapter 1 > Attached to: Bridged Adapter

Shared Folders > Folder Path: C:\share

Shared Folders > Auto-mount: Yes

Shared Folders > Mount point: /share

Shared Folders > Permanent: Yes

**Note:** If you are running the VM while Connected to the Oracle Corp VPN, then you need to use the Bridged Adapter to be able to connect to the software repositories used for OS packages.

**Start the new VM:**

You will be prompted to Select start-up disk.

Browse to find your Oracle Linux 7.6 ISO file.

**Install Oracle Linux 7.6**

Software Selection > Base Environment: Server with GUI

Network & Host Name > Ethernet: On

Network & Host Name > Host name: olvm1 *(for example)*

Begin Installation:

Root Password > Root Password: XXXXXXXX

User Creation > Make this user administrator: Yes

**Update the OS after install**

# Login using your personal account, open a terminal and become root

su -

visudo

# Make sudo not require a password for member of the wheel group

Change:

## Allows people in group wheel to run all commands

%wheel ALL=(ALL) ALL

## Same thing without a password

# %wheel ALL=(ALL) NOPASSWD: ALL

To look like::

## Allows people in group wheel to run all commands

# %wheel ALL=(ALL) ALL

## Same thing without a password

%wheel ALL=(ALL) NOPASSWD: ALL

# Update the OS, this will take some time, get a cup of coffee or two

systemctl stop packagekit

yum update -y

# Enable ol7\_developer and ol7\_developer\_EPEL

vi /etc/yum.repos.d/public-yum-ol7.repo

# Install packages required by the Guest Additions

yum install -y kernel-uek-devel kernel-devel kernel-headers gcc dkms

# If you want to use the VM to connect to the OSC VPN then install the OpenConnect VPN Client

yum install -y openconnect

# Reboot the VM

shutdown -r now

**Install the VirtualBox Guest Additions**

# From the Window that the VM is running in:

VirtualBox -> Devices -> Insert Guest Additions CD image…

# A window should pop up prompting to Cancel or Run, you want to Run it.

**Add user to the VirtualBox Shared Filesystem Group:**

For any user you want to be able to copy files to and from the shared filesystem you can make them a member of the vboxsf Linux Group:

sudo usermod -a -G vboxsf oracle

**Launch OpenConnect**

This window will need to stay open

sudo openconnect --quiet --usergroup OSC-VPN sca-osc-vpn.oraclevpn.com

The following document provides notes on setting up the VM's Network to get into the OSC VPN:

[VirtualBox-VM-Network-OSC.docx](https://stbeehive.oracle.com/content/dav/st/North%20American%20Infrastructure%20&%20Architecture%20Work-Group/Documents/VirtualBox%20Image/VirtualBox-VM-Network-OSC.docx)

**Shared filesystem:**

If the shared filesystem is not mounted, use can use the following to mount it:

sudo mount -t vboxsf share /share

**Copy & Paste friendly Terminal Emulator**

The GNOME GTK Apps don't handle the X11 xterm style of select to clipboard correctly, so if you want to be able to just highlight text in a terminal window and paste it into a Word document then I suggest installing the KDE konsole terminal emulator as it can be configured to do this.

sudo yum install -y konsole

Update the default profile:

konsole -> Settings -> Edit Current Profile -> Mouse -> Copy & Paste

X Copy on select

X Paste from selection

**GNOME Setup**

Open a terminal and run the following commands to make the desktop friendlier to use by:

* Change the screen saver to longer idle time, and disable the screen lock
* Change the desktop icons to be smaller
* Disable the auto maximize window feature and edge tiling

gsettings set org.gnome.desktop.session idle-delay 14400

gsettings set org.gnome.desktop.screensaver lock-enabled false

gsettings set org.gnome.nautilus.icon-view default-zoom-level small

gsettings set org.gnome.mutter auto-maximize false

gsettings set org.gnome.desktop.wm.preferences focus-new-windows 'smart'

dconf write /org/gnome/shell/extensions/classic-overrides/edge-tiling false

**Adding a New File System on a New Disk**

1. Shutdown the VM, Hot add is available using command line but not in the GUI
2. Settings -> Storage > Controller: SATA > Add Hard Disk  > Create new disk > VDI > Dynamically allocated > vdisk2, size GB > Create > Ok
3. Startup the VM
4. If this is the second device, it should show up as /dev/sdb

$ sudo /usr/bin/ -lbp

NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT

/dev/sdb 8:16 0 17179869184 0 disk

/dev/sr0 11:0 1 1073741312 0 rom

/dev/sda 8:0 0 25769803776 0 disk

/dev/sda2 8:2 0 24695013376 0 part

/dev/mapper/ol\_olvm1-swap 252:1 0 2147483648 0 lvm [SWAP]

/dev/mapper/ol\_olvm1-root 252:0 0 22540189696 0 lvm /

/dev/sda1 8:1 0 1073741824 0 part /boot

1. Create a Physical Volume on the device and verify it's there:

$ sudo /usr/sbin/pvcreate **/dev/sdb**

Physical volume "/dev/sdb" successfully created.

$ sudo /usr/sbin/pvscan

PV /dev/sda2 VG ol\_olvm1 lvm2 [<23.00 GiB / 4.00 MiB free]

PV /dev/sdb lvm2 [16.00 GiB]

Total: 2 [<39.00 GiB] / in use: 1 [<23.00 GiB] / in no VG: 1 [16.00 GiB]

1. Create a Volume Group using the Physical Volume and verify is usable space:

$ sudo /usr/sbin/vgcreate **vg01 /dev/sdb**

Volume group "vg01" successfully created

$ sudo /usr/sbin/vgs --units=m

VG #PV #LV #SN Attr VSize VFree

ol\_olvm1 1 2 0 wz--n- 23548.00m 4.00m

vg01 1 0 0 wz--n- 16380.00m 16380.00m

1. Create a Logical Volume in the new Volume Group using the size available above:

$ sudo /usr/sbin/lvcreate -L **16380m** -n **u01\_lv vg01**

Logical volume "u01\_lv" created.

1. Create a File System on the Logical Volume:

$ sudo /usr/sbin/mkfs -t xfs **/dev/vg01/u01\_lv**

meta-data=/dev/vg01/u01\_lv isize=256 agcount=4, agsize=1048320 blks

= sectsz=512 attr=2, projid32bit=1

= crc=0 finobt=0, sparse=0, rmapbt=0, reflink=0

data = bsize=4096 blocks=4193280, imaxpct=25

= sunit=0 swidth=0 blks

naming =version 2 bsize=4096 ascii-ci=0 ftype=1

log =internal log bsize=4096 blocks=2560, version=2

= sectsz=512 sunit=0 blks, lazy-count=1

realtime =none extsz=4096 blocks=0, rtextents=0

1. Create the File System Mount Point and update the File System Table so it will be mounted on reboot:

$ sudo mkdir **/u01**

$ sudo chmod 755 **/u01**

$ sudo bash -c 'echo "**/dev/vg01/u01\_lv /u01** xfs defaults 1 2" >> /etc/fstab'

$ tail -2 /etc/fstab

/dev/mapper/ol\_olvm1-swap swap swap defaults 0 0

/dev/vg01/u01\_lv /u01 xfs defaults 1 2

1. Manually Mount the File System and verify its size, etc.:

$ sudo mount **/u01**

$ df -hP **/u01**

Filesystem Size Used Avail Use% Mounted on

/dev/mapper/vg01-u01\_lv 16G 33M 16G 1% /u01

**Growing an Existing File System by adding a New Virtual Disk**

1. Shutdown the VM, Hot add is available using command line but not in the GUI
2. Settings -> Storage > Controller: SATA > Add Hard Disk  > Create new disk > VDI > Dynamically allocated > vdisk2, size GB > Create > Ok
3. Startup the VM
4. If this is the second device, it should show up as /dev/sdb

$ lsblk -lbp

NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT

/dev/sdb 8:16 0 4294967296 0 disk

/dev/sr0 11:0 1 1073741312 0 rom

/dev/sda 8:0 0 17179869184 0 disk

/dev/sda2 8:2 0 11810111488 0 part

/dev/mapper/ol-swap 252:1 0 1287651328 0 lvm [SWAP]

/dev/mapper/ol-root 252:0 0 10515120128 0 lvm /

/dev/sda1 8:1 0 1073741824 0 part /boot

1. Create a Physical Volume on the device and verify it's there:

$ sudo /sbin/pvcreate /dev/sdb

Physical volume "/dev/sdb" successfully created.

1. Add the Physical Volume to the Volume Group and verify is usable space:

$ sudo /sbin/vgs --units=m

VG #PV #LV #SN Attr VSize VFree

ol 1 2 0 wz--n- 11260.00m 4.00m

$ sudo /sbin/vgextend ol /dev/sdb

Volume group "ol" successfully extended

$ sudo /sbin/vgs --units=m

VG #PV #LV #SN Attr VSize VFree

ol 2 2 0 wz--n- 15352.00m 4096.00m

1. Extend the Logical Volume and verify its new size:

$ sudo /sbin/lvs --units=m

LV VG Attr LSize Pool Origin Data% Meta% Move Log Cpy%Sync Convert

root ol -wi-ao---- 10028.00m

swap ol -wi-ao---- 1228.00m

$ sudo /sbin/lvextend -L+4096m /dev/ol/root

Size of logical volume ol/root changed from 9.79 GiB (2507 extents) to 13.79 GiB (3531 extents).

Logical volume ol/root successfully resized.

$ sudo /sbin/lvs --units=m

LV VG Attr LSize Pool Origin Data% Meta% Move Log Cpy%Sync Convert

root ol -wi-ao---- 14124.00m

swap ol -wi-ao---- 1228.00m

1. Grow the file system to consume the free space on the resized Logical Volume and verify its new size:

$ df -kP /

Filesystem 1024-blocks Used Available Capacity Mounted on

/dev/mapper/ol-root 10258432 5501628 4756804 54% /

$ sudo /sbin/xfs\_growfs /

meta-data=/dev/mapper/ol-root isize=256 agcount=4, agsize=641792 blks

= sectsz=512 attr=2, projid32bit=1

= crc=0 finobt=0 spinodes=0 rmapbt=0

= reflink=0

data = bsize=4096 blocks=2567168, imaxpct=25

= sunit=0 swidth=0 blks

naming =version 2 bsize=4096 ascii-ci=0 ftype=1

log =internal bsize=4096 blocks=2560, version=2

= sectsz=512 sunit=0 blks, lazy-count=1

realtime =none extsz=4096 blocks=0, rtextents=0

data blocks changed from 2567168 to 3615744

$ df -kP /

Filesystem 1024-blocks Used Available Capacity Mounted on

/dev/mapper/ol-root 14452736 5501724 8951012 39% /

**Note:** This was an XFS filesystem so xfs\_growfs was used, if the file system was ext4 then the correct command would have been: /sbin/resize2fs /dev/ol/root