# VMS on KVM in the Cloud: Guided First Steps

Rafael Martins & Kobe Smith
December 9th, 2024



VMS Software

# Agenda

# Presentation Objectives

- This presentation will:
  - Show the process of instance creation.
  - Serve as a quick reference guide for when you try it yourself.
  - Share our learned lessons and roadblocks.
  - Motivate you to try it out yourself!

- This presentation will **not:**
  - Be the ultimate guide to full blown Linux KVM install.
  - Mention details about OCI networking and storage. (Stay tuned for future presentations!)

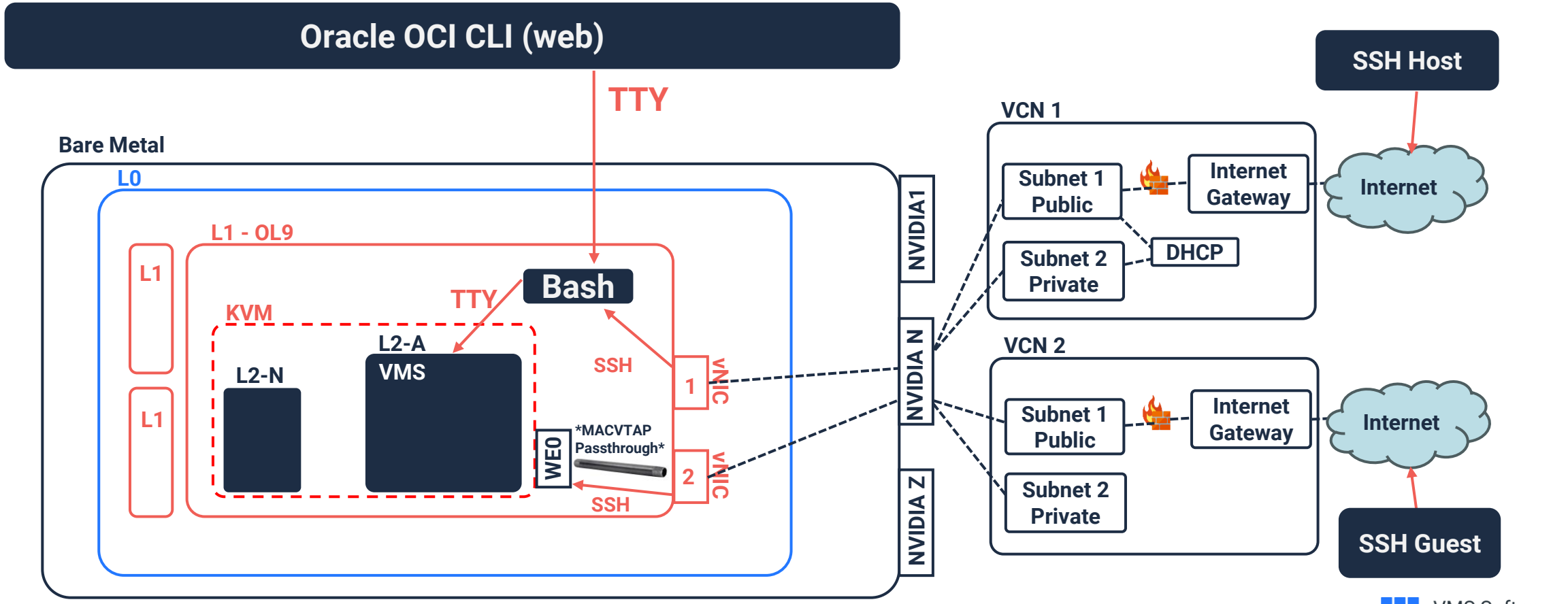
- Let’s get started!

# Why the cloud?

- Many companies prioritize cloud solutions for their businesses, whether that be a cloud-first, cloud-only, or cloud-ready model.
  - The Professional Services team at VMS has made efforts to alleviate these concerns by testing the first OpenVMS system on Oracle Cloud Infrastructure.

- VMS Software and CDL collaborated to create a proof of concept of an existing customer VMS system being replicated in and migrated to the Cloud.

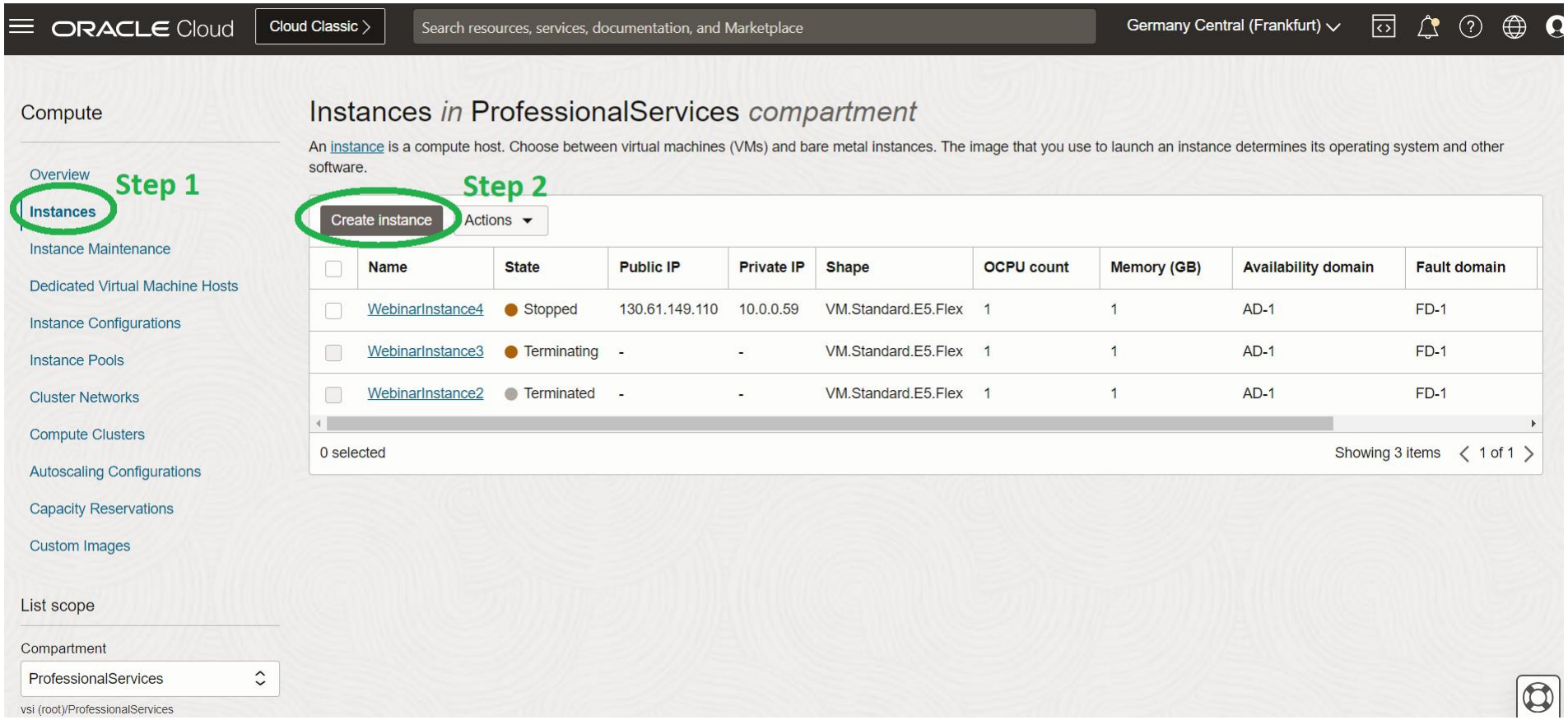
# System Topology




VMS Software

# Manual Creation of an instance

## Hands on demo.




VMS Software

# Scripted Creation of an instance

## Understand underline, adjust **bold**:

```
CIDR_BLOCK_PRIVATE="10.0.0.0/24"

export OCI_COMPARTMENT_OCID=\
"ocid1.compartment.oc1..aaaaaaaa6mtjuvf74dx7dwrxylb7qkfnnowxhydc2hus4aswuxqfqqyacsdq" #PS
compartment

OCI_VCN_OCID=$(oci network vcn create --cidr-block "${CIDR_BLOCK_PRIVATE}" --compartment-id
"${OCI_COMPARTMENT_OCID}" --query data.id --raw-output)

OCI_SUBNETPRIVATE_OCID=$(oci network subnet create --cidr-block "${CIDR_BLOCK_PRIVATE}" --
compartment-id "${OCI_COMPARTMENT_OCID}" --vcn-id "${OCI_VCN_OCID}" --query data.id --raw-output)

export OCI_AVAILABILITY_DOMAIN="lixT:EU-FRANKFURT-1-AD-1"
export OCI_IMAGE_OCID="ocid1.image.oc1.eu-frankfurt-
1.aaaaaaaajnfw4vhjfoenaoqhcymfdf47ep4angkowyuh3b4dzi56pkqcxavq" #Oracle-Linux-9.4-2024.09.30-0

oci compute instance launch --debug \
--display-name "WebinarInstance2" \
--availability-domain "${OCI_AVAILABILITY_DOMAIN}" \
--compartment-id "${OCI_COMPARTMENT_OCID}" \
--image-id "${OCI_IMAGE_OCID}" \
--shape "VM.Standard.E5.Flex" --shape-config '{"ocpus": 1, "memoryInGBs": 1}' \
--subnet-id "${OCI_SUBNETPRIVATE_OCID}" --assign-public-ip true
```

# Creation of an instance – Notes 1

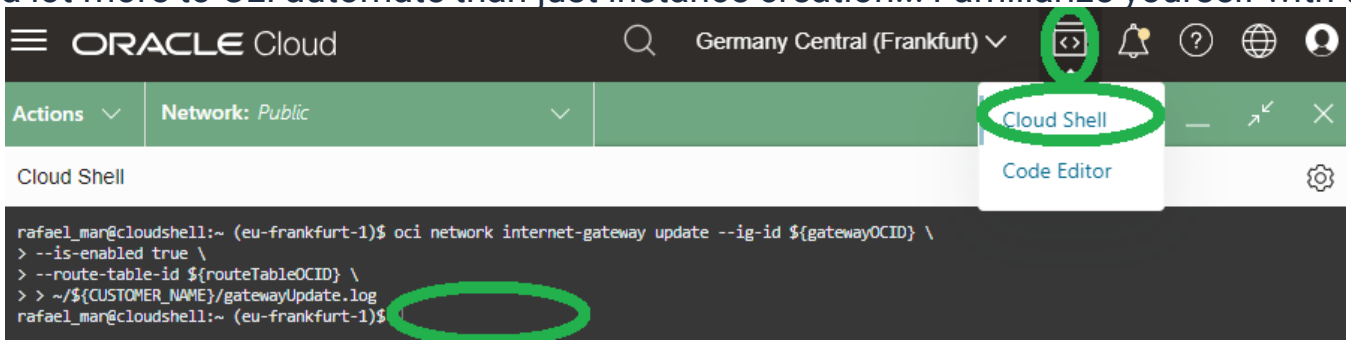
✅ The advantages of running on a non bare-metal machine
  - Scale up and down on CPUs, RAM, and even change the virtual CPU manufacturer (Intel/AMD)
  - ⚠️ Scale up HDD space (*but not down*)

- Documentation
  - Replace CLI version with "latest" for auto redirect 👈
  - https://docs.oracle.com/en-us/iaas/tools/oci-cli/**3.50.2**/oci_cli_docs/cmdref/compute/instance/launch.html

- Inside your instance, you can fetch main OCID hashes (L1 bash):

```
curl -H "Authorization: Bearer Oracle" -L http://169.254.169.254/opc/v2/instance/ > /tmp/metaData
OCI_INSTANCE_NAME=$(cat /tmp/metaData |grep "displayName" |cut -d '"' -f 4); echo "$OCI_INSTANCE_NAME"
OCI_INSTANCE_ID=$(cat /tmp/metaData |grep "\"id\"" |cut -d '"' -f 4); echo "$OCI_INSTANCE_ID"
OCI_COMPARTMENT_ID=$(cat /tmp/metaData |grep "compartmentId" |cut -d '"' -f 4); echo "$OCI_COMPARTMENT_ID"
```

👉 There is a lot more to CLI automate than just instance creation... Familiarize yourself with the OCI CLI



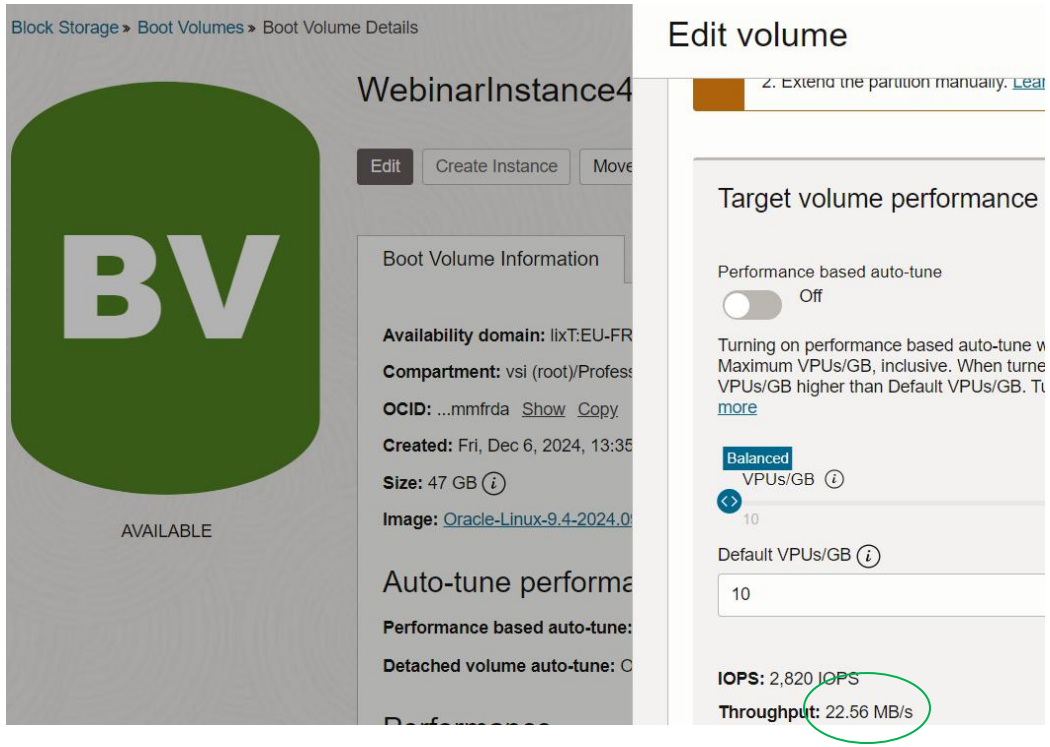



VMS Software

# Creation of an instance – Notes 2

- HDD transfer speeds (and "auto-tune")
  - ⚠️ *Yes, likely the default is not fast enough for your needs.* ***Change it****->*
  - HDD transfer rates as measured by Professional Services team :

| KVM - OCI | | | |
|---|---|---|---|
| | **/dev/random** | **fixed file** | **Settings** |
| OL9 host | 42,8 | 43,8 | cache=default |
| | 62,5 | 59,9 | Block device performance: VPU/GB:20 |
| VMS guest | - | 21,40 | cache=write through<br>virt device=AHCI<br>backingFile format in host: qcow2 |
| | | | in MB/s |






VMS Software

# Oracle Linux 9 (L1 host) - System management

- Very similar to Redhat, also quite intuitive

⚠️ Create a new user (with password, for recovery if network goes down)

```
#TODO: plain password shown. You might want to change this to a manual step instead
sudo useradd recoveryConsole -p "$(echo "MyRecoveryConsolePassword1!" | openssl passwd -6 -stdin)"
sudo usermod -aG wheel recoveryConsole
```

⚠️ Auto updates

```
sudo systemctl enable --now dnf-automatic.timer; sudo systemctl enable --now dnf-automatic-install.timer
```

- Expand host (L1) boot storage to max allocated (can also be done after a subsequent expansion)
  - `/usr/libexec/oci-growfs`
  - `df -h #to confirm the size increased`
  - `lsblk #trust but verify`
- User escalation
  - `sudo su -`
- We were not going to talk about network, but...
  - `sudo oci-network-config -d #delete dangling vNICs`
  - 👉 `sudo oci-network-config -c #add newly created vNICs`
  - `sudo service NetworkManager restart`
  - `sudo nmcli device status`

👉 X forwarding trick when super user

```
sudo cp -f \${HOME}/.Xauthority /root/.Xauthority\n\ #after a regular user ssh connection (if to be escalated)
```

# Oracle Linux 9 (L1 host) - Package Management

- dnf is your friend!
- ⚠️ Enable repos, download updates in bacground
  - ```
    sudo dnf -y config-manager --set-enabled ol9_developer_EPEL ol9_codeready_builder #add repos
    sudo dnf -y update #--downloadonly
    sudo dnf -y upgrade #--downloadonly & #update the system
    ```
- (wait) for package downloads, then enable base-x for later screen forwarding
  - ```
    sudo dnf groupinstall -y base-x
    sudo systemctl set-default graphical
    sudo reboot
    ```
  - **Note: there is no need for full blown kde/gnome/similar desktop environment.** base-x is enough
- Some other interesting packages you may consider:
  - 👉
    ```
    export virtSWPackagesToInstall="virt-manager libvirt qemu-kvm virt-install virt-viewer" #kvm packages
    export OCIPackagesToInstall="oci-utils python39-oci-cli python39-oci-sdk" #OCI packages export
    netPackagesToInstall="bridge-utils wireshark nm-connection-editor" #network packages export
    genericSWPackagesToInstall="xauth dnf-automatic htop cpuid traceroute pv" #other packages export

    sudo dnf install -y ${virtSWPackagesToInstall} ${OCIPackagesToInstall} ${netPackagesToInstall} \
    ${genericSWPackagesToInstall} ${graphicsSWPackagesToInstall} ${temporarySWPackagesToInstall}"
    ```

# KVM – Guest install

```
myVMName="myFirstVMSCloudVM"
myVMDescription="fingers crossed"

sudo virt-install \
 --name="${myVMName}" \
 --description="${myVMDescription}" \
 --ram=6000 --vcpus=3 --cpu host \
 --disk device=cdrom,path=${installISOLocationFullPath},bus=sata,boot_order=1,shareable=on \
 --disk path=${rootedBaseFullPath},size=32,bus=sata,cache=writethrough,discard=unmap \
 --disk path=${sysDiskFullPath},size=8,bus=sata,boot_order=2,cache=writethrough,discard=unmap \
 --disk device=cdrom,path=${updateKitsISOLocationFullPath},bus=sata,cache=none,shareable=on \
 --disk device=cdrom,path=${scratchPadISOLocationFullPath},bus=sata,cache=none,shareable=on \
 --disk device=cdrom,path=${updateKitsISOLocationFullPath},bus=sata,cache=none,shareable=on \
 --network=bridge:virbr0,mac=52:54:00:BE:EF:01,model='e1000' \
 --network=bridge:virbr0,mac=52:54:00:BE:EF:03,model='virtio-net' \
 --graphics none --osinfo detect=on,require=off --os-variant none --machine q35 \
 --boot loader=/usr/share/OVMF/OVMF_CODE.secboot.fd,loader.readonly=yes,loader.type=pflash,loader_secure=yes \
 --features vmport.state=off,smm.state=on \ --serial pty \ --noreboot \ --autostart
```



https://docs.vmssoftware.com/vsi-openvms-x86-64-v923-installation-guide/#section_bnh_chg_vzb

# KVM – Fine Tuning

- GUIs are your friend, (until they aren't)
  👉Not all KVM settings are visible in the python GUI.
- CLIs are your friend, (until they aren't)
  👉Not all KVM settings are available in the command line
- CLI XML editing to the rescue
  ⚠️ Do not edit the xml directly though! (do instead: dump->edit->reimport)
  - Example of a network edit to change from VEPA to macvtap passthrough:

```
sudo virsh dumpxml ${domain} > /tmp/virshOut_${domain}
sed $'s/<source dev=\''${targetIF}$'\' mode=\'vepa\'\/>/<source dev=\''${targetIF}$'\' mode=\'passthrough\'\/>/'
/tmp/virshOut_${domain} > /tmp/virshOut_${domain}_edited
sed $'s/<mac address=.*/<mac address=\''${orgMAC}$'\'\/>/' /tmp/virshOut_${domain}_edited >
/tmp/virshOut_${domain}_edited2 #--mac does not always work on above command
sudo virsh define /tmp/virshOut_${domain}_edited2
```

- Thin Provisioning (in KVM is called *pre-allocate*, or "*Allocate entire volume now*") & Watermarking
  - One or the other: the uselessness of both
  - ext4 use extents. if ext4, read about aio=threads (vs other systems aio=native)
  - Performance:
  - https://www.slideshare.net/slideshow/qemu-disk-io-which-performs-better-native-or-threads/62724391#58

# Host-Guest Interaction – File Systems

**File System Discrepancies**

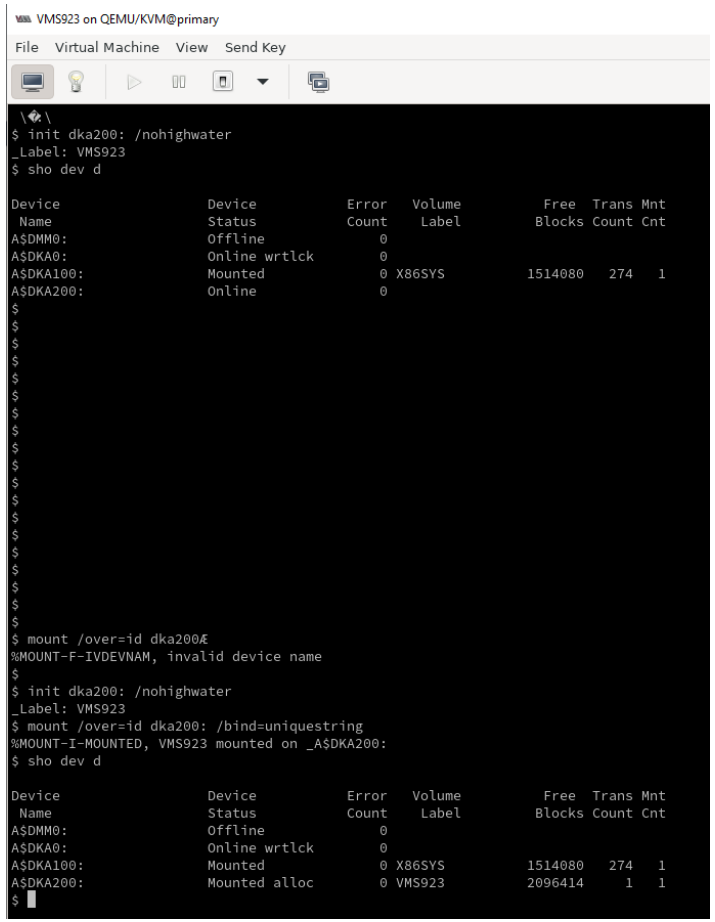
- OpenVMS uses Files-11 file systems.
- ODS-2 and ODS-5 do not currently work with Linux (requires the use of an external plug-in) or Windows natively.
- Prior to the set-up of a Network connection, you'll be unable to transfer files to the Guest. To get around this, we devised a new methodology using Virtual CD Drives.

**Virtual CD-drives**

- Virtual CD-drives are the recommended way to transfer files *into* the guest OS while your network is not set-up. Since they are read-only, transferring files *out of* the guest is not possible.

**Thin Provisioning & High Watermarking**

- Using both is useless. With Thin-provisioning already enabled on the Host, you'll want initialize VMS Devices as ODS-5 with /nohighwater, and /LIMIT:



# Host-Guest Interaction – Transferring Files

## 1. Converting files from Linux to VMS

- XFS > ISO 9660 Level 2 (ODS-3) > ODS-5
- It is important to understand this conversion due to the limitations each file system presents.
- This restricts filenames to:
  - Uppercase letters
  - Numbers
  - Underscores (_)
  - Characters < 31
  - Directory nesting < 8
  - pathnames < 255 chars
  - Single file size < 32b Bytes (i.e. < 4GB)

## 2. To avoid corruption when transferring files from VMS to VMS via CD-ROM, always use a container file! Follow these best practices for .zip:

In Linux, md5sums can be generated with the command:

```
md5sum yourFileName
```

In VMS, md5sums can be generated with the command:

```
$ checksum Filename /algorithm=md5
```

In linux, zip/unzip as follows:

```
zip myZipFile.zip my list of files to be zipped space separated
unzip -o myZipFile.zip -d myDestinationDirectory;
```

In VMS, zip/unzip as follows:

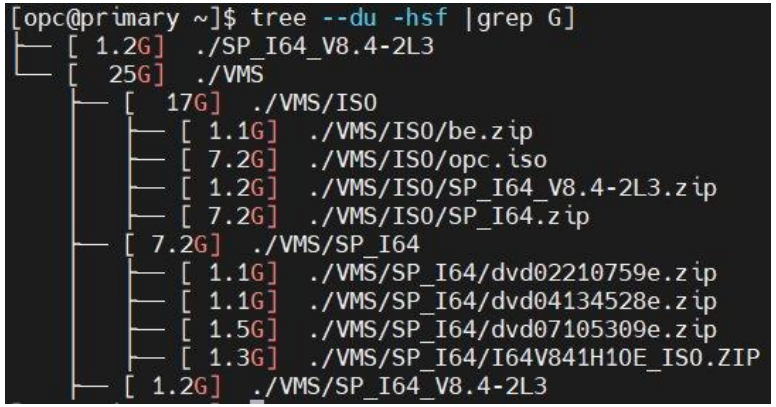
```
$ zip "-V9" file.zip  file.dsk
$ unzip file.zip
```

- Always zip with switch "-V" to preserve VMS file attributes.
- Always unzip with switch "-b" to recover VMS file attributes under all conditions.
- Before you unzip, you should inspect the archive with either one of these two commands:
  - UNZIP "-Z" archive.zip
  - ZIPINFO archive.zip
- Make sure you have the zip and unzip foreign commands defined.

```
zip :== $SYS$COMMON:[SYSHLP.UNSUPPORTED.ZIP]ZIP.EXE
zip -r myZipFile.zip fileToZip.txt
unzip :== $SYS$COMMON:[SYSHLP.UNSUPPORTED.UNZIP]UNZIP.EXE
unzip myZipFile.zip -d [.myDestinationDirectory]
```

# Host-Guest Interaction – Most common issues

1. Sudden loss of guest console access:
   - Check host CPU usage and HDD throughput
   - Check if the VM is "paused"
   - Try to "resume"
     - Rinse & repeat.
   - Check host Hard disk usage. df –h
     - *Bazinga*! Host ran out of disk space
   - Alternative: use disk-based storage (e.g. /dev/sdb1)
     - Side effect: more difficult to snapshot
   - Alternative 2: LVM
     - Looks like disk based, but much easier to snapshot.
2. Console forwarding
   - Avoid the GUI from the virt-manager guest window (**lag**!). Instead:
   - `sudo virsh console \${instance} < /dev/tty" #Press CTRL+5 to exit`
   - Use only 1 at a time! It's a serial tty direct console.
   - That also means you don't lose session state! (compared to ssh).
   - Tabbing out no longer slows things down. (Fixed by increasing SSH keepalive)






VMS Software

# Host-Guest Interaction – Most common issues cont.

3. Two volumes can't be mounted at the same time due to %MOUNT-F-DUPRVN, duplicate volume number already mounted
    - Fixed by adding the /bind=unique_string to each mount, with the unique string being the volume name or whatever you desire.
    - i.e `$ mount /over=id dka500: scratchpad /media=cdrom` **`/bind=scratchpad`**
4. Backup save set corruption
    - Backup save sets are not safe from corruption! Luckily, they can be repaired:
    - $ backup/repair/lis ENGX02$dka100:[000000.backups]USER.BCK/sav
    - Odd behaviors can occur on larger savesets:
5. Backups cannot be restored directly off the CD-Rom.
    - They must first be moved into a physical device to avoid error (%BACKUP-F-MOUNTF11, DKA200:[000000].;* must be mounted Files-11)

```
Directory DKB0:[000000]

BATS.BCK;1                    28224
CREATIONDATE.;1                   1
CREATIONDATE.ZIP;1                1
DATA.BCK;1               4294541637
DOCS.BCK;1                   464940
SYSTEM.BCK;1                1530144
USER.BCK;1               4292929152

Total of 7 files, 8589494099 blocks.
$
```

# Host-Guest Interaction – Console & Disk Usage

- Direct python window opening:
  - `sudo virt-manager --connect qemu:///system --show-domain-console "myFirstVMSCloudVM"`
- See all your guests
  - `sudo virsh list --all`
- Storage overview:
  - `sudo virsh domstats "myFirstVMSCloudVM"`
    `echo -e \"\033[1;32mStorage devices in kvm defined order:\033[0m\"\n\`
    `sudo virsh domstats "myFirstVMSCloudVM" | grep path | cut -d '=' -f2 | xargs -I {} ls -lah {}`
- Managing host space:
  - `tree --du -hsf |grep G]`
- Expanding the guest (L2) storage space
  - `sudovirsh list`
    `sudo virsh shutdown "myFirstVMSCloudVM"`
    `sudo virsh domblklist "myFirstVMSCloudVM"`
    `sudo qemu-img info`
    **`/full/path/to/your/image/backing/storage/file/here/backingFileName.qcow2`**
    `sudo qemu-img`
    `resize` **`/full/path/to/your/image/backing/storage/file/here/backingFileName.qcow2`** `+20G`

# VMS Guest – Networking Pitfalls

**Do <u>NOT</u> use DECnet Phase IV Addressing when configuring networking settings related to OCI**

DECnet Phase IV addressing will change the MAC address to AA:00:04:00:xx:xx, which in the OCI environment causes all network traffic to be blocked, thus making it impossible to establish an SSH connection.

Configure DECnet to avoid this:

```
@SYS$MANAGER:NET$CONFIGURE ADVANCED
```

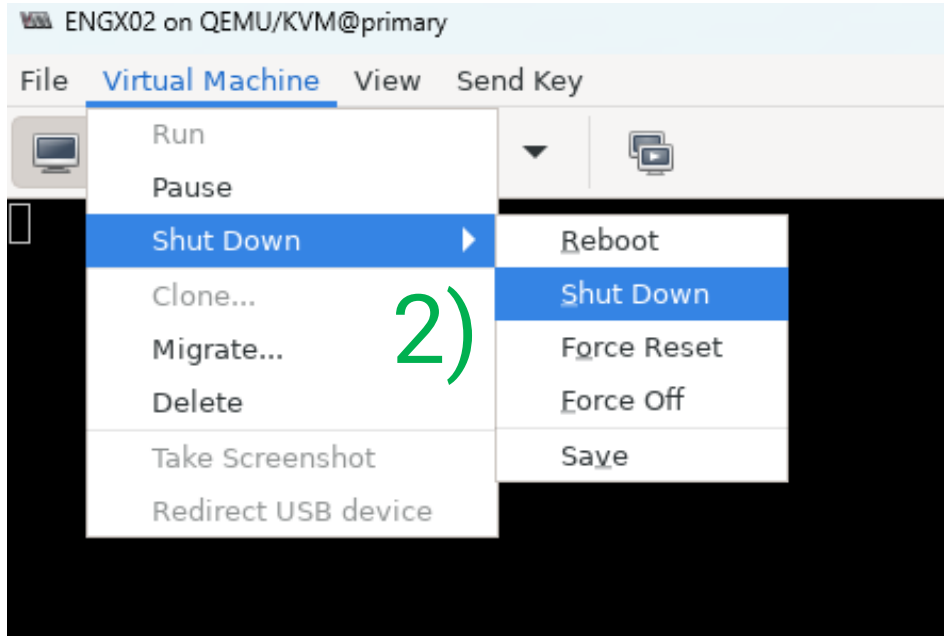
```
        [0]     Exit this procedure

        [1]     Perform an entire configuration
        [2]     Change naming information
        [3]     Configure Devices on this machine
        [4]     Configure Transports
        [5]     Configure Timezone Differential Factor
        [6]     Configure Event Dispatcher
        [7]     Configure Application database
        [8]     Configure MOP Client database
        [9]     Configure Cluster Alias
        [10]    Replace MOP Client configuration
        [11]    Configure satellite nodes
        [12]    Configure cluster script locations

* Which configuration option to perform?                [1] : 3
%NET$CONFIGURE-I-SCANCONFIG, scanning device configuration - please wait
* Data Link name to use for EWA0 (DExxx/TULIP)?   [CSMACD-1] :
* Routing Circuit Name for Data Link 'CSMACD-1'?  [CSMACD-1] :
* Enable Phase-IV Addressing on Routing Circuit 'CSMACD-1'? [NO] : NO
* Data Link name to use for EWB0 (DExxx/TULIP)?   [CSMACD-2] :
```

# VMS Guest – Workarounds

- *“Setting will apply after guest shutdown"*
  - This truly means SHUTDOWN.
    - Not a ***reboot***, not a ***force reboot***, a ***power off*** then a ***power up*** again enforces changes.
  - Some settings are hot applicable, some are not.
    - If in doubt, power off and power on again.
- ACPI commands from OCI OL9 KVM into VMS are not yet working.
  👉 We are looking into why. Therefore, in the meantime:

1)

```
$ @sys$system:shutdown
```



# VMS Guest – Example Scripts

## Restoring Backups of Customer Data

```
$ dism dkb200:
$ mount /for dkb200: System
$ backup/repair/lis ENGX02$dka200:[000000.lds]system.bck/sav
$ backup/image ENGX02$dka200:[000000.lds]system.bck/sav dkb200
```

## *Refreshing Disks

```
$ set noon
$ dism dka400:
$ mount /over=id dka400: updatekits /media=cdrom /bind=kits
$ set def dka400:[000000]
$ unzip creationdate.zip
  $! This is a built-in sanity check for data corruption
$ type creationdate.;1
$ checksum creationdate.zip /algorithm=md5
$ sho symbol checksum$checksum
$ dir /date
$ set def sys$login
$ sho dev d
```

## Ease of use edits to Login.com

```
$ set term/insert
$ set backspace=delete
$! Symbol for Zip/Unzip
$! dir dka300:[000000...]unzip
$ unzip :==$dka300:[VMS$COMMON.SYSHLP.UNSUPPORTED.unzip]unzip.exe
```

## Device Directory Creation

```
create /dir [.work]
create /dir [.quota]
create /dir [.data]
```

## Initializing devices

```
init dka200: system
init dka400: data
init dka500: backups
```

## Mounting Devices

```
$ set noon
$ mount /over=id dka100:
$ mount /over=id dka200:
$ mount /over=id dka300:
$ mount /over=id dka400: updatekits /media=cdrom /bind=updatekits
$ mount /over=id dka500: scratchpad /media=cdrom /bind=scratchpad
$ mount /over=id dkb0: backups /media=cdrom /bind=backups
$ @sys$startup:ld$startup
$ ld connect DKA200:[000000.LDS]user.dsk lda1:
$ mount /over=id lda1: USER
$ ld connect DKA200:[000000.LDS]BATS.dsk lda2:
$ mount /over=id lda2: BATS
$ ld connect DKA200:[000000.LDS]DOCS.dsk lda3:
$ mount /over=id lda3: DOCS
$ sh dev d
$ sh dev ld
```

# A talk about the process - theoretical

- Yes it is fun to play with it, poke around, manually edit things, etc
  - But write it down!
    - Even better as **scripts** (only very few manual prompts)
- Once scripts are ready, run a fully official deployment
  - Rinse and repeat
- Yes, the last 10% of finalizing the scripts will take the other 90% of time
  - But also true that you will reuse those scripts
    - an order of magnitude more than you thought!
- Need more convincing?
  - **Automation** – time saving
  - **Reproduceable**! - Support will love you!
  - **Auditable**! -> if you git, git, git!

```
oci network security-list create \
--display-name "GUEST-SECURITYLIST-PRIVATE-${CUSTOMER_NAME}" \
--compartment-id ${OCI_COMPARTMENT_OCID} \
--vcn-id ${vcn_id} \
 --egress-security-rules \
'[
 {
 "description": null,
 "icmp-options": null,
 "is-stateless": false,
 "protocol": "all",
 "source": "0.0.0.0/0",
 "source-type": "CIDR_BLOCK",
 "destination": "0.0.0.0/0",
 "destination-type": "CIDR_BLOCK",
 "source-port-range": null,
 "tcp-options": null,
 "udp-options": null
 },
 {
 "description": null,
 "icmp-options": null,
 "is-stateless": false,
 "protocol": "all",
 "source": "0.0.0.0/0",
 "source-type": "CIDR_BLOCK",
  "destination": "0.0.0.0/0",
 "destination-type": "CIDR_BLOCK",
 "tcp-options": null,
 "udp-options": null
 },
 {
 "description": null,
 "icmp-options": {
 "code": 4,
 "type": 3
 },
 "is-stateless": false,
 "protocol": "1",
 "source": "0.0.0.0/0",
 "source-type": "CIDR_BLOCK",
  "destination": "0.0.0.0/0",
 "destination-type": "CIDR_BLOCK",
 "tcp-options": null,
 "udp-options": null
 },
 {
 "description": null,
 "icmp-options": {
 "code": null,
 "type": 3
 },
 "is-stateless": false,
 "protocol": "1",
 "source": "10.0.0.0/16",
 "source-type": "CIDR_BLOCK",
  "destination": "0.0.0.0/0",
 "destination-type": "CIDR_BLOCK",
 "tcp-options": null,
 "udp-options": null
 }
]' \
```

# Topics for next cloud/KVM session (please vote)

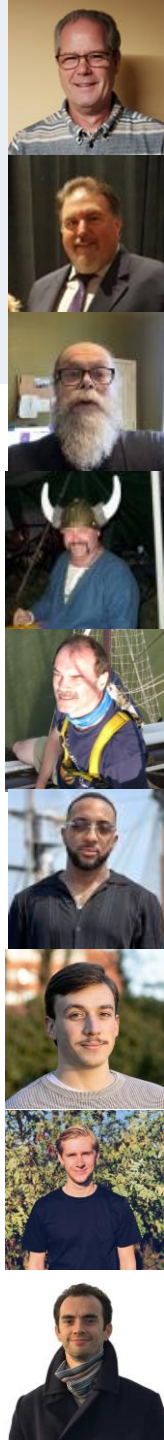
- Networking setup
- Host (L1) Backup
- Performance benchmarking
- Fine tuning
- Costing
- Host scripting and automation
- More wrong corners and gotchas! (*show us your mistakes*)
- Others?

- Please vote, comment, subscribe!

# Get in touch with Professional Services!

- We are a mixed team, with lots of years of VMS experience and well versed in many contemporary IT tools

- We are here to help, get in touch!
- Save your time! Tools, scripting, automation
- Learn from our mistakes! Processes, best practices
- Leverage our attention to details: Performance Testing, Fine Tuning
- Delegate your responsibilities: Full system migration, updates and upgrades, plus 24/7 managed services
- And much more! Contact your account manager for details.



# Any questions?

- OCI
- KVM
- VMS