ESX on ARM (Ampere A12c)

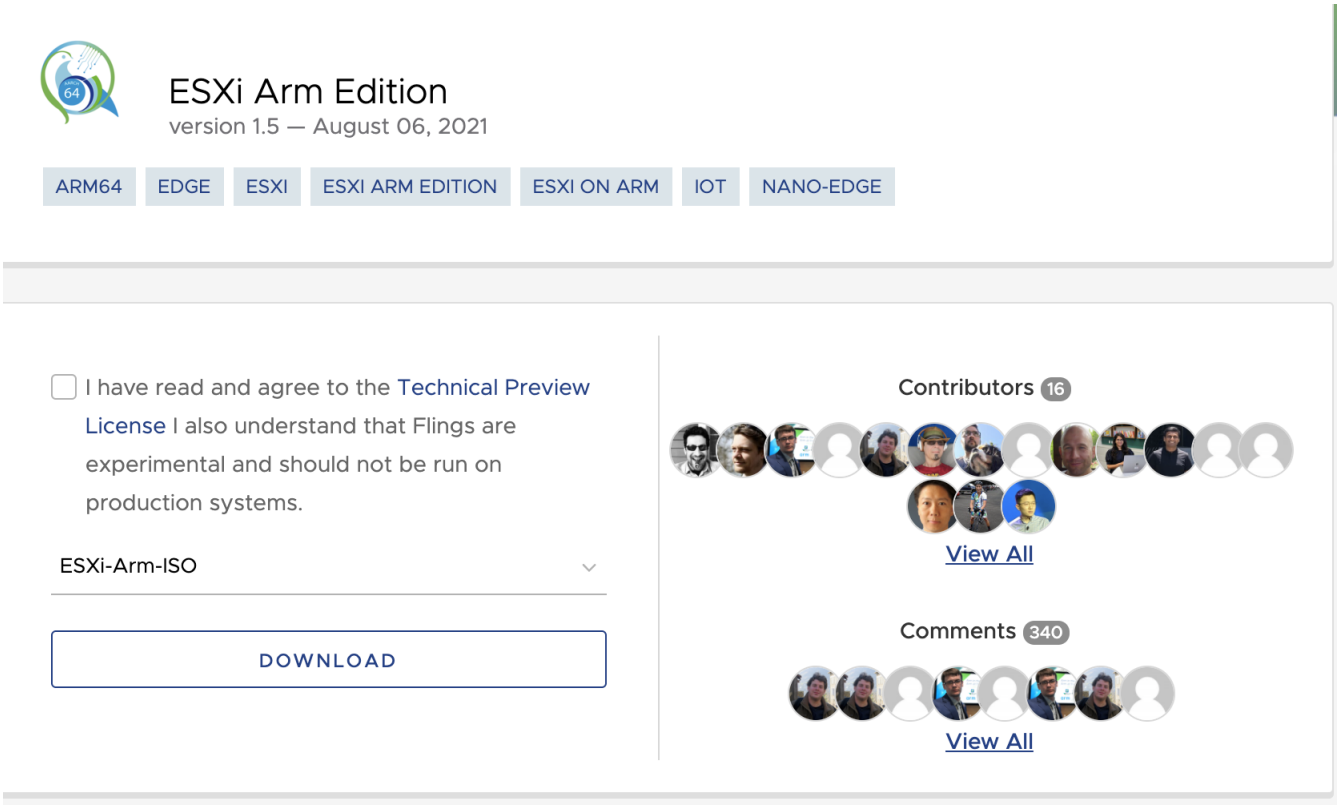
Scope:

This document covers the steps required to boot an ESX image on OCI Ampere A1-2C BareMetal nodes

> ⚠ Before you get started, Please register for OCI ARM accelerator program and request for limits to run a A1-2C BM

Step1:

Download the ESX ARM ISO from the VMware Flings website



Step2:

Create a iPXE server to host the ISO

The iPXE server will hold the ISO needed to perform the install of the ESXi operating system on the donor BM instance. Ensure that port 80 has been allowed on the subnet for the iPXE server.

Follow this procedure:

1. Instantiate a VM instance (try to use a VM for same CPU architecture if possible).
2. Provision a block volume of at least 50G to hold the images. Attach to the instance, format and mount on /var/www (create the directory). Ensure that the /etc/fstab is updated to mount the volume at boot.
3. Install the following packages: httpd rsync lzma gcc libffi-devel python3-devel openssl-devel unzip gcc rsyslog jq python-firewall git
4. Attach a block volume to hold the ISOs, expansion of ISOs, and ipxe information.
5. Run the following commands on the iPXE server

```
      git clone http://git.ipxe.org/ipxe.git
      cd ipxe/src
      LANG=C make bin-arm64-efi/snponly.efi
      mkdir -p /var/www/html/ipxe/70/
      mkdir -p /var/www/html/70/
      cp bin-arm64-efi/snponly.efi /var/www/html/ipxe/70/
      yum -y  install httpd rsync lzma gcc libffi-devel python3-devel openssl-devel unzip gcc rsyslog
jq python-firewall git devtoolset-8 emacs-nox python36-oci-cli
      scl enable devtoolset-8 bash
      mkdir /mnt
      mount -o loop /home/opc/VMware-VMvisor-Installer-7.0.0-18427252.aarch64.iso /mnt
      pushd /mnt; find . | cpio -pmvdu /var/www/html/70/
      cp /mnt/efi/boot/boot.cfg /var/www/html/ipxe/70/
      cp /mnt/efi/boot/bootaa64.efi /var/www/html/ipxe/70/mboot.efi
      service httpd status
      service httpd start
      service httpd status
      systemctl enable httpd
      service httpd status
      chmod -R ug+rwX,o-w,o+rX /var/www/
      firewall-cmd --zone=public --add-service=http --permanent
      firewall-cmd --reload
      mkdir /var/www/html/ipxe/70/esxi_ks
```

6. Update the boot.cfg

```
bootstate=0
title=Loading ESXi installer
timeout=5
# This is the iPXE server IP
prefix=http://10.0.2.175/70
kernel=b.b00
# This is the iPXE server IP
kernelopt=runweasel cdromBoot ks=http://10.0.2.175/70/esxi_ks/ks.cfg
modules=jumpstrt.gz --- useropts.gz --- features.gz --- k.b00 --- procfs.b00 --- vmx.v00 --- vim.v00
--- tpm.v00 --- sb.v00 --- s.v00 --- ena.v00 --- bnxtnet.v00 --- bnxtroce.v00 --- brcmfcoe.v00 ---
brcmnvme.v00 --- elxiscsi.v00 --- elxnet.v00 --- i40en.v00 --- i40iwn.v00 --- iavmd.v00 --- igbn.v00
--- iser.v00 --- ixgben.v00 --- lpfc.v00 --- lpnic.v00 --- lsi_mr3.v00 --- lsi_msgp.v00 --- lsi_msgp.
v01 --- lsi_msgp.v02 --- mtip32xx.v00 --- ne1000.v00 --- nenic.v00 --- nfnic.v00 --- nhpsa.v00 ---
nmlx4_co.v00 --- nmlx4_en.v00 --- nmlx4_rd.v00 --- nmlx5_co.v00 --- nmlx5_rd.v00 --- ntg3.v00 ---
nvme_pci.v00 --- nvmerdma.v00 --- nvmxnet3.v00 --- nvmxnet3.v01 --- pvscsi.v00 --- qcnic.v00 ---
qedentv.v00 --- qedrntv.v00 --- qfle3.v00 --- qfle3f.v00 --- qfle3i.v00 --- qflge.v00 --- rste.v00 ---
sfvmk.v00 --- smartpqi.v00 --- vmkata.v00 --- vmkfcoe.v00 --- vmkusb.v00 --- vmw_ahci.v00 --- elx_esx_.
v00 --- btldr.v00 --- esx_dvfi.v00 --- esx_ui.v00 --- esxupdt.v00 --- tpmesxup.v00 --- weaselin.v00 ---
loadesx.v00 --- lsuv2_hp.v00 --- lsuv2_in.v00 --- lsuv2_ls.v00 --- lsuv2_nv.v00 --- lsuv2_oe.v00 ---
lsuv2_oe.v01 --- lsuv2_oe.v02 --- lsuv2_sm.v00 --- native_m.v00 --- qlnative.v00 --- vmware_e.v00 ---
vsan.v00 --- vsanheal.v00 --- vsanmgmt.v00 --- tools.t00 --- imgdb.tgz --- imgpayld.tgz
build=7.0.0-1.0.18427252
updated=0
```

7. Create a ks.cfg (Kickstart file)

```
#[root@ipxe-arm-server3 opc]# cat /var/www/html/ipxe/70/esxi_ks/ks.cfg
#Accept the VMware End User License Agreement
vmaccepteula

# Set the root password for the DCUI and Tech Support Mode
rootpw myp@ssw0rd

# Install on the first local disk available on machine
install --firstdisk --overwritevmfs

# Set the network to DHCP on the first network adapter
network --bootproto=dhcp --device=vmnic0
#[root@ipxe-arm-server3 opc]#
```

The iPXE server should be ready for use at this point.

Step3 :

Building the image

Building the image is composed of four parts: The iPXE script at launch, launch script, installation of ESXi, and verification.

iPXE Script

The iPXE script is created and placed the same directory from where the BM will be launched using the launch script

```
#!ipxe

set target-iqn iqn.2015-02.oracle.boot:uefi

set root-path iscsi:169.254.0.2::::${target-iqn}

set alt-root iscsi:169.254.0.2:::1:${target-iqn}

set eversion 70

sanhook ${root-path} || sanboot ||

sanhook ${alt-root} || sanboot || set url http://10.0.2.175

kernel ${url}/ipxe/${eversion}/mboot.efi -H -c ${url}/ipxe/${eversion}/boot.cfg

boot
```

Launch Script

This is the launch script that was used to build the image in R1. Modify to your needs:

```
#!/bin/bash

SSH_PUBLIC_KEY=`cat ~/vinay_ssh.pub`

SUBNET_ID="ocid1.subnet.oc1.uk-london-1.aaaaaaaaw5rpswosoowvdbatshxhgtya5a4ncltkdcolyxgyonh6fagimbqq"

COMPARTMENT_ID="ocid1.tenancy.oc1..aaaaaaaaz3vffq654srxtl5zbzqjs7sugln2xthz73inbiig62s3ggpb25fq"

AD="PuSg:UK-LONDON-1-AD-1"

# VERSION is a concatenation of the ESXi version without the "."s

VERSION="7"

SHAPE="BM.Standard.A1.160"

# IMAGE_ID is the AllZeros image (or representation thereof) that can be used as a blank canvas

IMAGE_ID="ocid1.image.oc1.uk-london-1.aaaaaaaaaaay7ngsg5wj3byvhhegi4rqcdiuv7zk6lozosjdvw3o6sqib2lq"

echo ${IMAGE_ID}

echo ${AD}

echo ${SUBNET_ID}

oci compute instance launch \

--compartment-id=${COMPARTMENT_ID} --availability-domain=${AD} \

--image-id=${IMAGE_ID} \

--boot-volume-size-in-gbs=512 \

--ipxe-script-file=ipxeboot-7 \

--shape=${SHAPE} --display-name="arm-esxi1" --subnet-id=${SUBNET_ID} \

--metadata='{"ssh_authorized_keys": """${SSH_PUBLIC_KEY}""" }'
```

Things to note:

- Use the same version string as you have for both the ipxeboot script and the directories on the iPXE instance.
- You need to have an AllZeros image that is devoid of an operating system. This prevents the BM instance to be used to generate the image from booting an OS and ignoring this process.
- OCID for AllZeros image

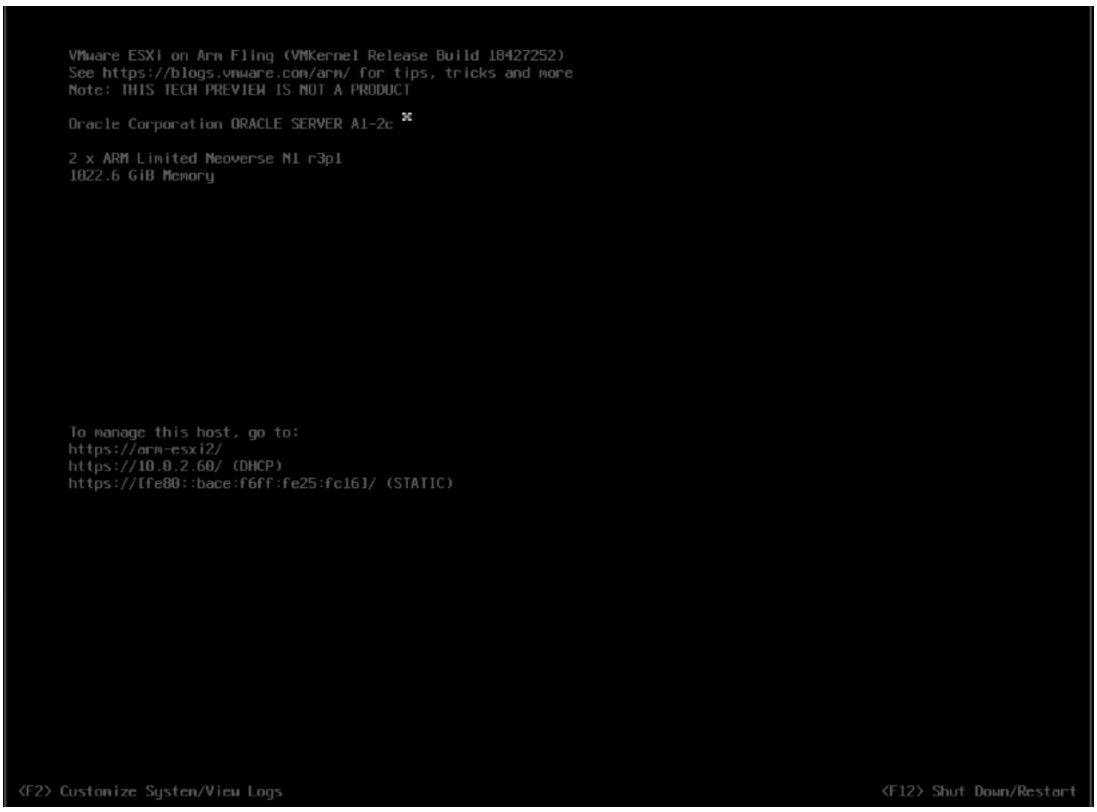
Region	OCID
PHX	ocid1.image.oc1.phx.aaaaaaaay27pdopotkapf2ahjlsn2wxndui5hn5w37hd2wss4ses4ol5xs6a
IAD	ocid1.image.oc1.iad.aaaaaaaaqftkoa5web2r7w4ls3wekgqmqy5f7untloetfiozyqbv2ql6qidq
FRA	ocid1.image.oc1.eu-frankfurt-1.aaaaaaaah4rggbyglst25peqd7vnyjzl6n5lwogiyllb6jaircakom46nswq
LHR	ocid1.image.oc1.uk-london-1.aaaaaaaaaaay7ngsg5wj3byvhhegi4rqcdiuv7zk6lozosjdvw3o6sqib2lq

Step4:

ESXi Installation

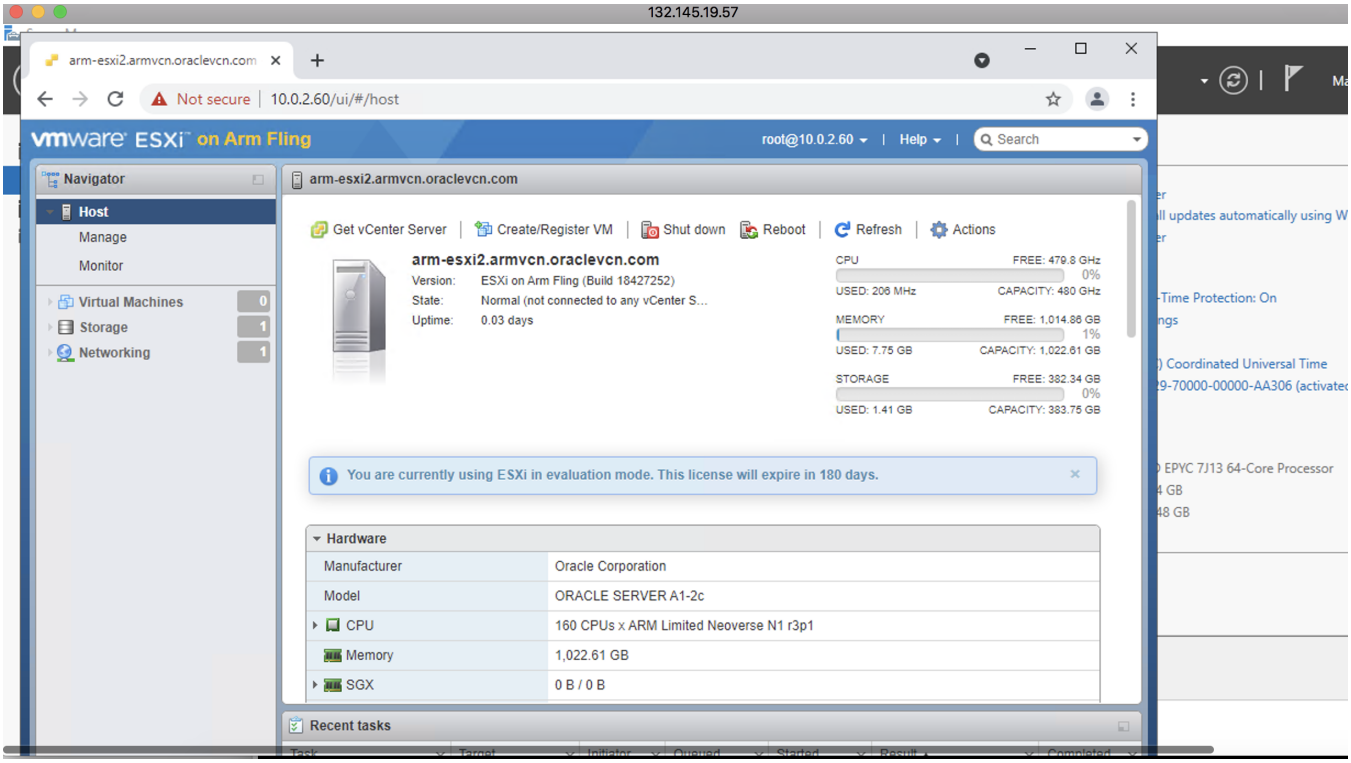
Perform the following procedure to for the initial installation. Execute the launch script above, coupled with the iPXE script to provision the BM instance.

1. Create a console connection to the BM instance when available.
2. Connect to the BM console *USING serial console.*
3. Hit "Enter" to reboot the box
4. Now connect to the BM console using VNC.
5. You will have the following screen.



Verification

Login to the ESX Host from a windows jump host:

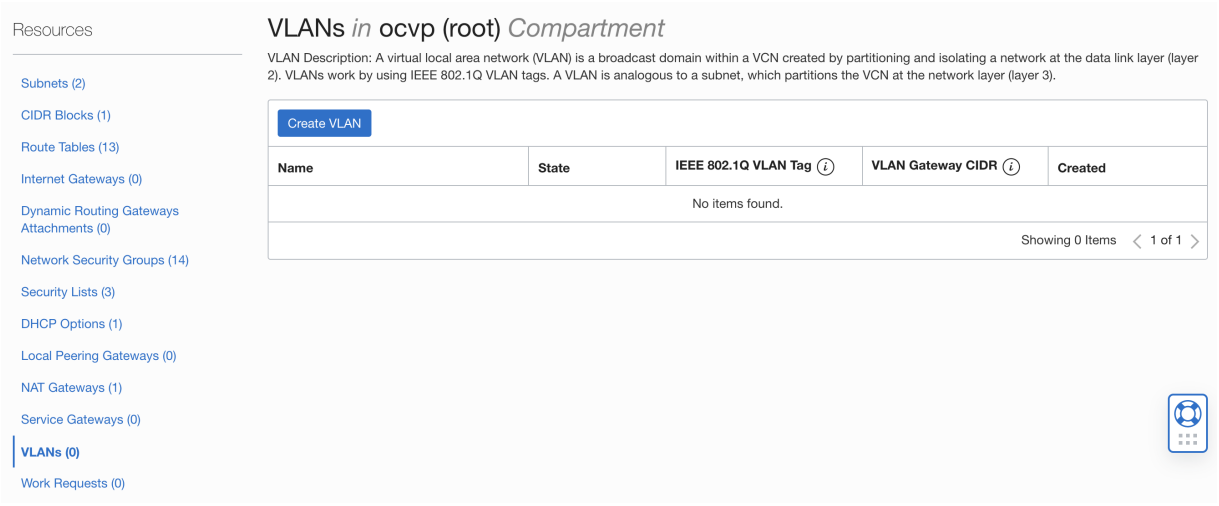


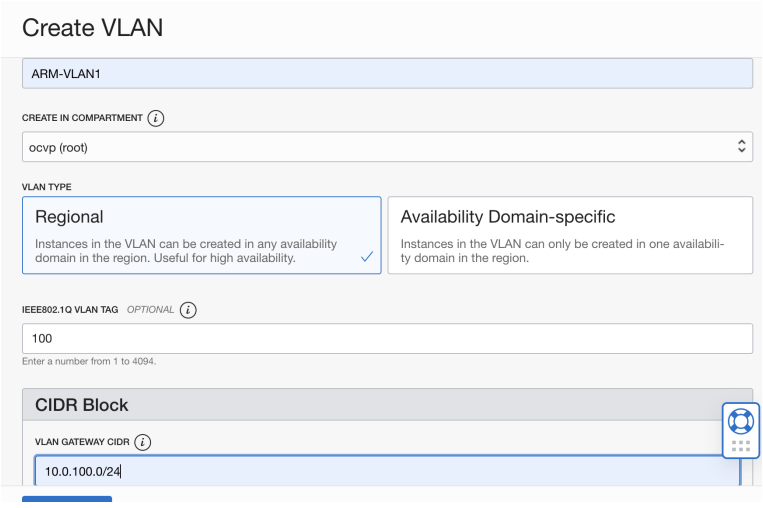
Step5:

Create the L2 VLAN:

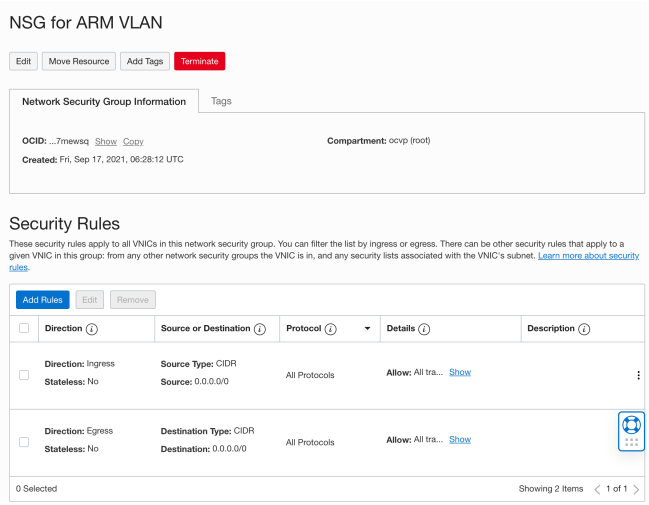
On OCI VCN, Create a VLAN and associate the same with RT and NSG

> ⚠ If you unable to create a VLAN, Please raise a SR with Oracle support and request allow-listing for VCN L2 feature

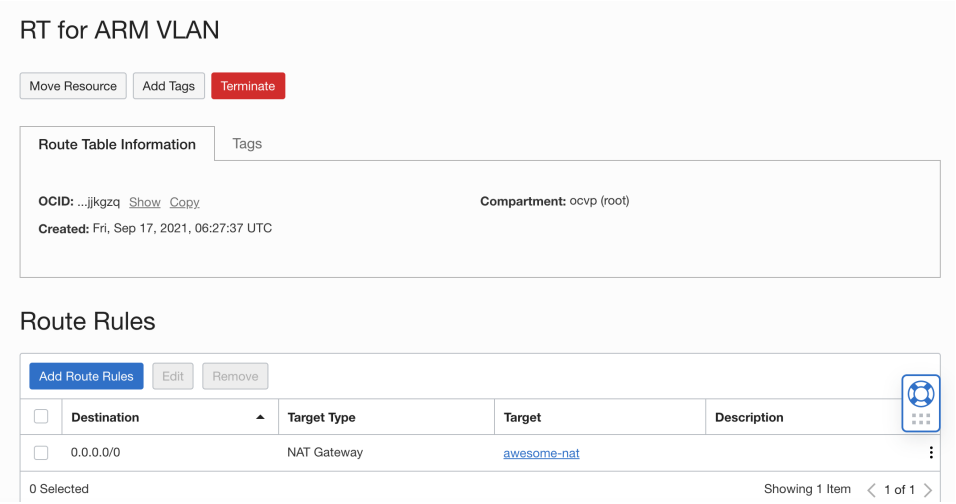




Create a NSG For the VLAN



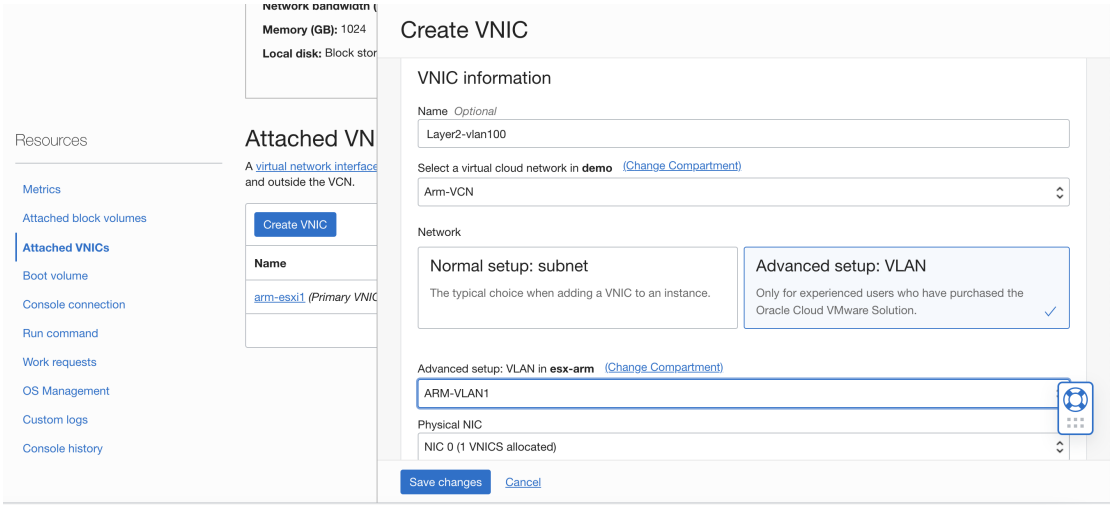
Create a Route table for the VLAN



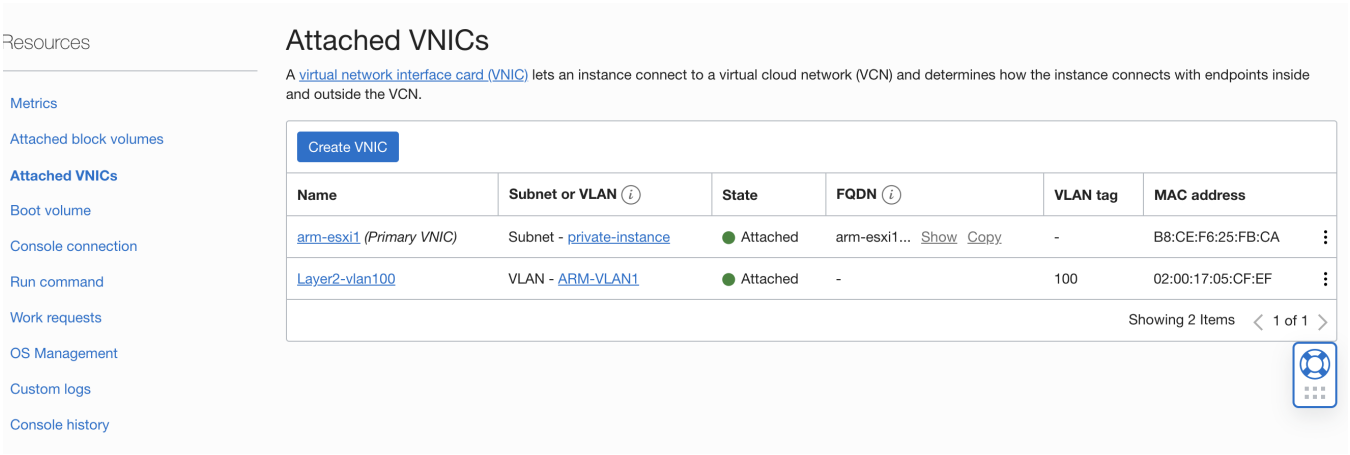
Step6:

Attach VLAN to ESX ARM node

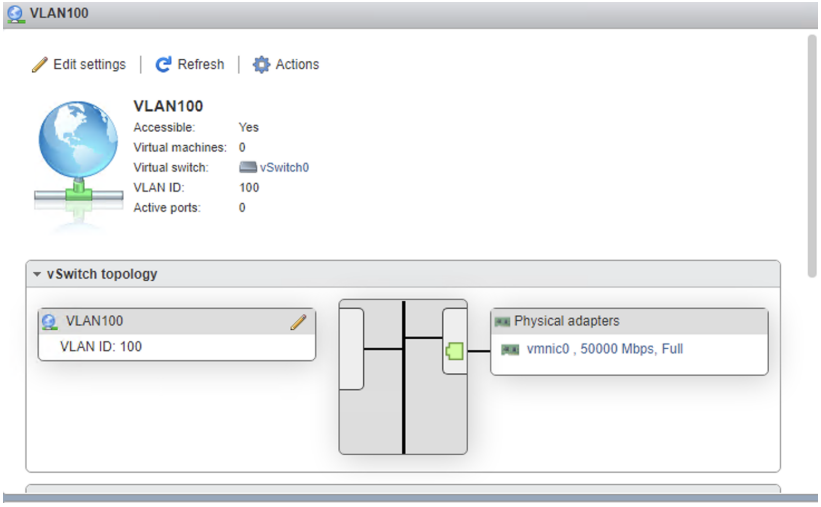
Under compute InstancesClick on the ARM Instance and attach a new Layer2 VNIC



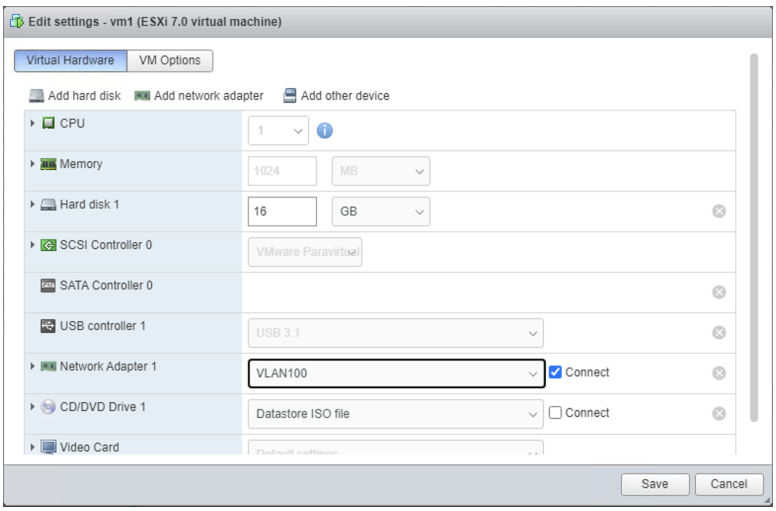
Verify that the VNIC is Attached



Now create a vSphere PortGroup on the ESX host with same VLAN ID



And attach a VM to the portgroup



And now Validate the traffic