Install Ubuntu VM on Fedora Host OS

Step 1: Install libvirt

```
sudo dnf group install --with-optional virtualization
sudo systemctl start libvirtd
sudo systemctl enable libvirtd
lsmod | grep kvm
```

Step 2: Download Ubuntu ISO image and install OS

```
virt-install --name ubuntu-runner --memory 16384 --vcpus 8 \
             --disk size=80 \
             --os-variant ubuntu22.04 \
             --cdrom /var/lib/libvirt/images/ubuntu-22.04.5-live-server-amd64.iso \
             --network bridge=virbr0 \
             --graphics spice
```

During install, in File System Summary step, unmount / and re-add with full space of 80 GB (it generally tends to only allocation half the space to /)

Useful commands:

```
Start VM
        sudo virsh start ubuntu-runner

Console access
        sudo virsh console ubuntu-runner

SSH Access
        sudo virsh domifaddr ubuntu-runner
        ssh <username>@<ip_address>

GUI Access
        virt-viewer ubuntu-runner

Stop VM
        sudo virsh shutdown ubuntu-runner
Destroy VM
        sudo virsh destroy ubuntu-runner
Reboot VM
        sudo virsh reboot ubuntu-runner
Delete VM
        sudo virsh undefine ubuntu-runner --remove-all-storage
```

Step 3: Facilitate Easy Access

On VM, allow root login

```
sudo nano /etc/ssh/sshd_config

# Edit the file to allow Root Login (line should look like below)
PermitRootLogin yes

# Restart ssh service
sudo systemctl restart ssh
```

On host OS, copy key + hostname

```
# Copy RSA key, so you dont have to type password each time
ssh-copy-id root@192.168.124.77

# Edit .ssh/config file and add following details, so you can simply login with 'ssh myvm'
nano ~/.ssh/config

Host myvm
    HostName 192.168.124.77
    User root
```

Step 4: Networking config

On host

```
sudo sysctl net.ipv4.ip_forward=1
sudo iptables -t nat -A POSTROUTING -o eno1 -j MASQUERADE
sudo firewall-cmd --permanent --direct --passthrough ipv4 -t nat -I POSTROUTING -o eno1 -j MASQUERADE
sudo firewall-cmd --reload

systemctl firewalld stop
```

On VM

```
# 1. update netplan settings
sudo nano /etc/netplan/00-installer-config.yaml

# Add following config (correct the IP addresses as needed)

network:
  version: 2
  ethernets:
    ens3:  # Replace with your actual interface name
      dhcp4: no
      addresses: [192.168.124.77/24]
      gateway4: 192.168.124.1
      nameservers:
        addresses: [8.8.8.8, 8.8.4.4]

# apply settings
sudo netplan apply


# 2. Config system-wide proxy settings

sudo nano /etc/profile.d/proxy.sh

export http_proxy="http://proxy-dmz.intel.com:911/"
export https_proxy="http://proxy-dmz.intel.com:912/"
export ftp_proxy="http://proxy-dmz.intel.com:912/"
export no_proxy="localhost,127.0.0.1"

export HTTP_PROXY="http://proxy-dmz.intel.com:911/"
export HTTPS_PROXY="http://proxy-dmz.intel.com:912/"
export FTP_PROXY="http://proxy-dmz.intel.com:912/"
export NO_PROXY="localhost,127.0.0.1"


sudo chmod +x /etc/profile.d/proxy.sh
source /etc/profile.d/proxy.sh

# 3. Configure APT to use proxy

sudo nano /etc/apt/apt.conf.d/80proxy


Acquire::http::Proxy "http://proxy-dmz.intel.com:911/";
Acquire::https::Proxy "http://proxy-dmz.intel.com:912/";
Acquire::ftp::Proxy "http://proxy-dmz.intel.com:912/";

# 4. Configure wget to use proxy

nano ~/.wgetrc

use_proxy = on
http_proxy = http://proxy-dmz.intel.com:911/
https_proxy = http://proxy-dmz.intel.com:912/
ftp_proxy = http://proxy-dmz.intel.com:912/

# 5. Verify

env | grep -i proxy
sudo apt update
wget https://www.google.com
```