1. Prepare the VM Template for Bare-Metal Deployment

Before converting the VM template into an image that can be deployed to bare-metal systems:

Remove Hardware-Specific Configurations:

Network Configuration: Ensure the VM’s network configuration is removed or generalized (e.g., remove /etc/sysconfig/network-scripts/ifcfg-eth0 or the equivalent).

Machine ID: Clear the machine ID and any other unique identifiers.

bash

sudo truncate -s 0 /etc/machine-id

sudo rm -f /var/lib/dbus/machine-id

Clean Cloud-Init:

bash

sudo cloud-init clean

sudo rm -rf /var/lib/cloud/\*

Convert the Disk to a Raw Image: Convert the VM’s disk image to a raw format, which is more universally accepted for deployment tools.

bash

sudo qemu-img convert -O raw /sto/KVM/rocky9.4-template.qcow2 /sto/KVM/rocky9.4-template.raw

2. Transfer the Image to Bare-Metal Hardware

You can use various tools to transfer the raw image to bare-metal systems:

Option 1: Using dd (Direct Copy)

If you have physical access to the bare-metal system or can boot into a live environment (like a rescue mode), you can use dd:

bash

sudo dd if=/path/to/rocky9.4-template.raw of=/dev/sdX bs=4M status=progress

Replace /dev/sdX with the target disk on the bare-metal system.

Option 2: PXE Boot with a Custom Image

If your environment supports PXE booting, you can serve the raw image via a network boot. This would involve setting up a PXE server and transferring the image over the network.

Set up a PXE boot environment with the necessary network infrastructure.

Deploy the image using tools like cobbler, pxe, or foreman.

Option 3: Convert Image to ISO for Bootable Media

Convert the image to an ISO that can be used to boot from:

Convert the image to a bootable ISO:

bash

mkisofs -o /path/to/rocky9.4-template.iso /sto/KVM/rocky9.4-template.raw

Burn the ISO to a USB drive or DVD and boot the bare-metal system from it.

3. Automate Configuration with Kickstart

If you want to further automate the process and configure the system post-deployment (like setting up partitions, users, etc.), you can combine this with a Kickstart file.

Prepare a Kickstart file with your desired configurations.

Deploy the image and ensure it points to the Kickstart file (e.g., via a local media or a network location).

4. Finalize the Setup

Once the image is on the bare-metal system:

Boot the system and allow it to initialize.

Post-Deployment Scripts: Use cloud-init or other automation tools to finalize the configuration (hostname, network settings, etc.).

Benefits of This Approach:

Consistency: The same image is used across VMs and bare-metal systems.

Speed: Deployment is faster compared to a traditional installation process.

Customization: You can still customize post-deployment with Kickstart or cloud-init.

This method provides flexibility in how you deploy your systems, allowing for rapid deployment on both virtual and physical infrastructure.