Creating VM template in proxmox

Now, we will be creating virtual machine template for our Apache webserver which we created earlier and then deploying a new VM from it.

Some things we don’t want to remain the same which is ssh host keys.

A black background with white text

Description automatically generated

First ensure cloud-init is installed

A screen shot of a computer

Description automatically generated

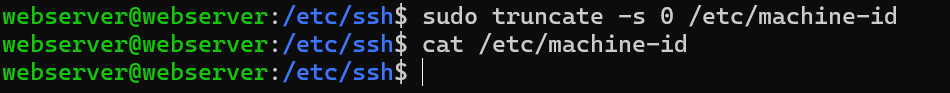
Next cd into /etc/ssh and remove all ss\_host keys



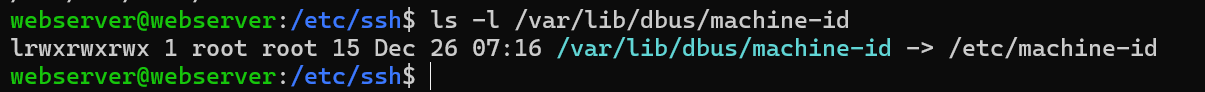
A screen shot of a computer

Description automatically generated

Also empty out the machine id



Ensure there is symbolic link if not have to create one that points to it with sudo ln -s /etc/machine-id /var/lib/dbus/machine-id



Run sudo apt clean to clean the app database

Run sudo apt autoremove as well

A screenshot of a computer program

Description automatically generated

Clean up the log files



/var/log: main system log directory, logs for system boot messages, authentication, system/kernel messages

/var/log/apt: logs specific to package management – package installations/updates/removals etc

/var/lib/apt/lists: contain package repository information (e.g. package metadata/package lists etc)

Power off your webserver VM



Or you can also run qm shutdown 100 (your VM id) from your proxmox host shell

You can run qm template 100 (vm id) to convert it to template

A black screen with white text

Description automatically generated

Or if you right click the VM in the UI you should see Convert to template option

A screenshot of a computer

Description automatically generated

Once converted you can see the icon changed to this

Click on the created template > hardware > edit the CD/DVD drive and select do not use any media

A computer screen shot of a computer

Description automatically generated

Then on the same screen add a cloudinit Drive

A screenshot of a computer

Description automatically generated

Select the vm-storage directory created in previous parts

A screenshot of a computer

Description automatically generated

You can select Cloud-Init then set a new default username/password

A screenshot of a computer

Description automatically generated

Once then click on regenerate image

A screenshot of a phone

Description automatically generated

Now, let’s create a VM out of that template

Using the UI: you can right click on the template and select Clone.

A blue and white box with black text

Description automatically generated

You can choose to clone with these settings from UI

A screenshot of a computer

Description automatically generated

Or you can do the same using your proxmox host terminal

qm clone 100 200 --name hustlercy-webclone --target hustlercy --storage vm-storage --format qcow2 --full 1

Breaking down the parameters:

* 100: Source template ID
* 200: New VM ID
* --name hustlercy-webclone: Name of the new VM
* --target hustlercy: Target node
* --storage vm-storage: Target storage
* --format qcow2: QEMU image format
* --full 1: For full clone (as opposed to linked clone). If linked clone can just exclude –full 1.

A screen shot of a computer program

Description automatically generated

Linked Clone:

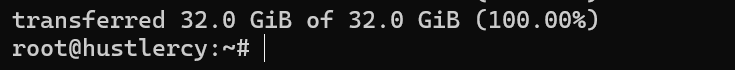
* Creates a thin clone that depends on the original template
* Only stores the differences between the template and clone
* Perfect for testing or temporary VMs
* Cannot exist without the original template

Use cases:

* Full Clone: Production servers, long-term VMs
* Linked Clone: Development environments, testing, temporary VMs, training environments where you need many similar VMs

You can convert a linked clone to a full clone later if needed using: qm template <vm-id> --force

Now that the command has finished running



You can see a new vm with id 200 as specified created



Now we can try to start the new VM



When you see "generating cloud-init ISO", this means:

1. The OS is already installed from your template (VM 100)
2. Cloud-init is just generating a small ISO file that contains initialization data such as:
   * Network configuration
   * SSH keys
   * Hostname
   * User accounts
   * Custom scripts to run on first boot

The cloud-init process:

1. When the VM starts, it detects the cloud-init ISO
2. Reads the configuration from this ISO
3. Applies the settings to your existing OS
4. The cloud-init ISO is temporary and is only used during boot

To delete your cloned vm you can ensure it is shutdown first and use qm destroy 201

A black background with white text

Description automatically generated

To view logs for your cloud-init:

Sudo journalctl -u cloud-init