Ubuntu 18.04 Template for XenServer

Introduction

This document details the steps required to configure a template XVA for XenServer.

XVA Template Creation

Follow these instructions to create an Ubuntu-18.04 XVA that can be used to quickly instantiate new deploy Kubernetes to XenServer. Note that these steps will only be **rarely necessary** to regenerate the XVA from scratch. Most of the time, you’ll be able to quickly **clone the pre-built XVA** downloaded from Amazon S3 from (where **#** is the Ubuntu revision):

https://s3.amazonaws.com/neonforge/kube/ubuntu-18.04.#.xva

or the latest version from:  
  
 <https://s3.amazonaws.com/neonforge/kube/ubuntu-18.04.latest.xva>  
  
NOTE: These files are **GZIP** encoded...

**Setup Instructions**

1. Download the **Ubuntu-18.04 Server ISO** from (where **#** is the desired revision) from:   
     
   http://releases.ubuntu.com/18.04/

Edit the downloaded file name to remove the last version field (probably “.0”) and the “live-server-amd64” and then archive the ISO file to (making it public):  
  
<https://s3.amazonaws.com/neonforge/kube/ubuntu-18.04.#.iso>

1. Burn the ISO to a DVD.
2. Power up XenServer and wait for the UX to display on the monitor.
3. Run XenCenter on your workstation and connect to the XenServer.
4. Create the setup VM:  
   1. **VM/New VM…**
   2. Scroll to the end and select **Ubuntu Xenial Xerus 16.04 (this is important)** and click **Next**
   3. Name the VM: **kube-ubuntu-18.04-base**
   4. **Insert the DVD** into the XenServer machine (the new VM will mount this by default).
   5. Click **Next** until you get processors and memory page. Set **4 procs** and **1024MB RAM**.
   6. Click **Next** until the last page and click **Finish** to create the VM.
   7. Click the **Console tab** and wait for the VM to boot.
5. Wait for the installation UX to start.
6. Press enter to select **English**.
7. Press enter to **Install**.
8. Press enter to skip proxy configuration.
9. Press enter to choose the default mirror.
10. Press enter three times to use the entire disk.
11. Press the down arrow and enter to confirm the file system settings.
12. Enter **sysadmin** as your name and username and enter **sysadmin0000** as the password. Set the server name to **ubuntu** and select **done**.
13. Don’t install any server Snaps.
14. Start the installation.
15. **Remove the DVD**, select **<empty>** in the DVD drive dropdown, and **Reboot**.
16. **Login** with the credentials you specified earlier to verify that the VM works.
17. Optional: Use the command below to discover the VM’s **IP address** for the **eth0** interface and connect via PuTTY or another terminal program that allows for easy copy and pasting of commands. Then connect to the server.

ip address

1. Run this command to start bash with root permissions (the password is **sysadmin0000**):  
     
   sudo bash
2. Run the following command to modify **sudo** behavior so it doesn’t request passwords, making remote configuration possible:  
     
   echo "%sudo ALL=NOPASSWD: ALL" > /etc/sudoers.d/nopasswd
3. Run the following commands to install the **ZIP** package (required for neonKUBE setup:  
     
   apt-get update  
   apt-get install -yq zip secure-delete
4. Disable swap by editing /etc/fstab and removing the /swap.img line.
5. Configure the Linux guest integration services:  
   1. nano /etc/initramfs-tools/modules
   2. Append this the file and save:  
        
      hv\_vmbus  
      hv\_storvsc  
      hv\_blkvsc  
      hv\_netvsc
   3. Install and configure the integration services and reboot:  
        
      apt-get install -yq linux-virtual linux-cloud-tools-virtual linux-tools-virtual  
      update-initramfs -u  
      reboot
6. Install the **XenServer/XCP-ng tools**:  
   1. **right-click** on the VM in XenCenter and click **Install XenServer Tools**. You may need to wait a minute or two for the menu item to show up.
   2. **mount** the tools DVD drive:  
        
      mount -o ro,exec /dev/sr0 /mnt  
        
      You may need to modify the /dev/sr0 path. Execute blkid and use the device with the XenServer Tools label. You may need to wait a minute or two for the DVD to mount.
   3. **Install** the tools via and then reboot:  
        
      /mnt/Linux/install.sh
   4. **Eject** the **DVD drive** in XenCenter (at the top-right of the VMs storage tab.
7. Run the following command to clear cached packages and the **cached DHCP** leases and then, zero deleted files (for better compression) and shutdown:  
     
   apt-get cleanrm -rf /var/lib/dhcp/\*  
   sfill -fllz /  
   shutdown -h now
8. Create the **XVA template**:  
   1. Right-click the VM and **Convert to Template**.
   2. Right-click on the template and select **Export to file…** Save the file to your workstation as **kube-ubuntu-18.04.#.xva** where **#** is the revision, set the **format to XVA** and then export to a local file.
9. GZIP compress the file and create a latest version if this is the latest:  
     
   gzip --best kube-ubuntu-18.04.#.xva
10. Use the AWS Console to **Upload** the image to the location below (where **#** is the revision:

Then upload the ZIP file to S3 (**removing the .gz** extension and setting **Content-Encoding=gzip**):  
  
<https://s3.amazonaws.com/neonforge/kube/ubuntu-18.04.#.vhdx>   
  
Be sure to: **Remove the .gz extension.**Be sure to: **Add AWS metadata: Content-Encoding = gzip**Be sure to: **Make these files public on AWS!**

1. If this is the latest Ubuntu image, we’ll also build a fully upgraded template:  
   1. In XenCenter, rename the most recent Ubuntu point release template from **kube-ubuntu-18.04-base** to **x-kube-ubuntu-18.04-base**.
   2. Create a VM named **kube-ubuntu-18.04-base** from the **x-kube-ubuntu-18.04-base** template.
   3. Start the VM, login via the Console tab and **get the IP address**.
   4. SSH into the VM using PuTTY and run these commands to **upgrade Ubuntu**:  
        
      sudo bash  
      apt-get update  
      apt-get dist-upgrade -yq
   5. And then these commands to **clean the disk** and **shutdown**:  
        
      apt-get clean   
      rm -rf /var/lib/dhcp/\*  
      sfill -fllz /  
      shutdown -h now
   6. Export the XVA file like we did above and compress and upload it to S3:  
        
      <https://s3.amazonaws.com/neonforge/kube/ubuntu-18.04.latest.vhdx>   
        
      Be sure to: **Remove the .gz extension.**Be sure to: **Add AWS metadata: Content-Encoding = gzip**Be sure to: **Make these files public on AWS!**

# XenServer Template

**XenServer Template Upload:** XenServer templates are GZIP compressed rather than using ZIP so the XenServer tooling can download them. The steps for creating a XenServer image is much like those for Hyper-V above. Essentially, you’ll use XenCenter to create the VM by mounting the Ubuntu setup ISO, setting up the VM, installing the XenServer tools, and then exporting it as an **\*.xva** file.  
  
I generally go through the full template build from ISO only for Ubuntu point releases. For intermediate package upgrades, I do the following:  
  
\* In XenCenter, rename the most recent Ubuntu point release template from **kube-ubuntu-18.04-base** to something like **x-kube-base**.  
  
\* Create a VM named **kube-ubuntu-18.04-base** from the **x-kube-base** template and run these commands to upgrade it:   
  
apt-get update  
apt-get dist-upgrade -yq  
rm -rf /var/lib/dhcp/\*  
shutdown -h now  
  
\* Convert the VM into a Xen Template and download/upload it to S3 as described further below.  
  
\* Delete the **kube-base** VM and then **rename x-kube-base** back to **kube-ubuntu-18.04-base**.  
  
Follow the steps below compress and upload it to AWS S3:

* 1. Create and initialize an Ubuntu virtual machine on XenServer much like we did above for Hyper-V. Name the VM **kube-ubuntu-18.04-base (this is important)** You’ll be using the console in XenCenter.
  2. After you **finished installing Ubuntu**: **right-click** on the VM in XenCenter and click **Install XenServer Tools**. You may need to wait a minute or two for the menu item to show up.
  3. Wait a minute or two for the tools driver to be mounted.
  4. SSH into the VM and **mount** the tools DVD drive:  
       
     sudo bash  
     mount -o ro,exec /dev/sr0 /mnt  
       
     You may need to modify the /dev/sr0 path. Execute blkid and use the device with the XenServer Tools label. You may need to wait a minute or two for the DVD to mount.
  5. **Install** the tools via and then reboot:  
       
     /mnt/Linux/install.sh  
     reboot
  6. **Login** back into the VM.
  7. **Eject** the **DVD drive** in XenCenter (at the top-right of the VMs storage tab) and shutdown the VM:  
       
     rm -rf /var/lib/dhcp/\*  
     shutdown -h now
  8. Right-click the VM and **Convert to Template**.
  9. Right-click on the template and select **Export to file…** Save the file as **kube-ubuntu-18.04.#.xva** where **#** is the revision, set the **format to XVA** and then export to a local file.
  10. GZIP compress the file and create a latest version if this is the latest. Note that we’re going to rename the file to remove the **.gz** file type after uploading to AWS:  
        
      gzip --best kube-ubuntu-18.04.#.xva
  11. If this is the latest Ubuntu image, restart and log back into the VM and then run:  
        
      sudo bash  
      apt-get update  
      apt-get dist-upgrade -yq  
      rm -rf /var/lib/dhcp/\*  
      shutdown -h now  
        
      After the VM has down, export it to **kube-ubuntu-18.04.latest.xva** and then GZIP it like:  
        
      gzip --best kube-ubuntu-18.04.latest.xva

1. Manually upload the files to S3 setting metadata to:  
     
   **S3** at [https://s3.amazonaws.com/neonforge/kube/\*](https://s3.amazonaws.com/neonforge/kube/*)   
     
   Be sure to: **Remove the .gz extension.**Be sure to: **Add AWS metadata: Content-Encoding = gzip**Be sure to: **Make these files public on AWS!**