Linux VM Template Setup

Introduction

This document details the steps required to configure a Linux Template VHD suitable for hosting Docker containers for development, testing, and production.

**NOTE**: Linux VMs cloned from this template will be optimized for running on SSDs. Development will not be testing Neon related server software on spinning HDs.

VHD Template Creation

Follow these instructions to create an Ubuntu VHD that can be used to quickly instantiate new Ubuntu VMs. Note that these steps will only be **rarely necessary** to regenerate the VHD from scratch. Most of the time, you’ll be able to quickly **clone the pre-built VHD** uploaded to Amazon S3 [here](https://s3.amazonaws.com/lilltek-public/Ubuntu-14-04-Node-Template.vhdx).

1. Download the Ubuntu Server ISO from:   
     
   <https://s3.amazonaws.com/lilltek-public/ubuntu-14.04.4-server-amd64.iso>
2. Open the **Hyper-V Manager** and step through the **New Virtual Machine Wizard**:  
   1. Name the VM **node-template**.
   2. Configure **Generation 1**Note: I tried using generation 2 but I was unable to clone VMs by making copies of the template VHD when I did this. The cloned VMs wouldn’t boot.
   3. Then **4096MB RAM** and **uncheck Dynamic Memory**.
   4. Networking connection = **Intel® Ethernet Connection (2) I218- Virtual Switch**
   5. Create a **127GB VHD** (this seems like a reasonable size for test development but can be customized).
   6. Installation Options: Configure to **boot** from the downloaded **Ubuntu ISO** and then press **Next/Finish**.
   7. Select the new VM in the Hyper-V manager, select settings, click **Processor** in the left panel and set **4 Virtual Processors**.
   8. Press **OK** to close Settings.
3. **Start the VM** and then **connect** to it via the Hyper-V Manager.
4. Press enter to select **English**.
5. Press enter to **Install Ubuntu Server**.
6. Press enter twice to select **English** and **United States**.
7. Press enter three times to select **English (US) keyboard**.
8. A bunch of stuff will be installed.
9. Enter **node-template** as the machine’s **host name**.
10. Create the **admin** **account:**Full Name: **spot**  
    UID: **spot**

PWD: **WagTheDog!**

1. Press **Enter** to **not encrypt** the home directory.
2. Select **Yes** and configure the default **Time Zone** (a later script will change this to UTC).
3. Press **Up-Arrow** and then **Enter** to select **Guided – use entire disk** (**do not setup LVM!**).
4. Press **Enter** to accept the device changes.
5. Press **TAB** and **Enter** to select **Yes** to **Write the changes to disks** andconfigure **standard partitions** (not LVM).
6. Wait for the system to install.
7. Press Enter to **skip proxy** configuration.
8. More software will be installed.
9. Press **Enter** to select **No automatic updates** (I figure we’ll want to control and test this explicitly in specific VM setup scripts).
10. Check **OpenSSL Server** in the **Software selection screen** by pressing **Space** and then press **Enter** to continue.
11. Even more software is installed.
12. Press **Enter** to Install the **GRUB** boot loader.
13. Press **Enter** to **Reboot**.
14. **Login** with the credentials you specified earlier to verify that the VM works.
15. Run this command to start bash with root permissions:  
      
    sudo bash
16. Edit the **/etc/sudoers** file to prevent sudo from requesting passwords (breaking scripts). Use the following command to edit the file:  
      
    visudo

and change the line starting with **%sudo** to:  
  
%sudo ALL=NOPASSWD: ALL

1. Use the command below to discover the VM’s **IP address** for **eth0** and make a note of it for the next step:

ifconfig

1. Use WinSCP to connect to the node (using the IP address from the last step) and copy the **$/Docker/Linux/Node/Ubuntu-14.04/\*.\*** files into a **new folder** called **/setup** on the VM. Then make all the scripts executable via:  
     
   chmod a+x /setup/\*.sh
2. Execute the **/setup/setup-node-template.sh** script to install the Docker related packages other common utilities as well as then upgrading the installed packages to their most current versions.  
     
   /setup/setup-node-template.sh
3. **Logout** and **shut down** the VM**.**
4. Use the AWS Console to **Upload** the new template VHD to: <https://s3.amazonaws.com/lilltek-public/ubuntu-14-04-node-template.vhdx> and grant **public read access**.