**Documentation for Maya Assessment 2**

**Overview:**

I was tasked to complete the Maya Assessment 2, which involves the use of a CentOS 6.x distribution (version 6.4, 6.5, 6.6, or 6.7). The setup includes the installation of necessary packages, user creation, and configuration of directories and cron jobs using Puppet.

**Part 1: Setup**

1. **Installed Oracle VirtualBox** and **Vagrant** to simplify the process of creating and managing virtual machine (VM) environments.

* Oracle VirtualBox: [Download Link](https://www.oracle.com/ph/virtualization/technologies/vm/downloads/virtualbox-downloads.html)
* Vagrant: [Installation Link](https://developer.hashicorp.com/vagrant/install?product_intent=vagrant)

1. **Installed CentOS 6.5 on Vagrant**:

* Download the pre-configured VM image from Vagrant Cloud: [CentOS 6.5 - Vagrant Box](https://portal.cloud.hashicorp.com/vagrant/discover/nrel/CentOS-6.5-x86_64)
* Set up a directory to store the CentOS distro: /Vagrant/centos-minimal-vagrant
* Run the following commands to initialize and start the VM:
* vagrant init nrel/CentOS-6.5-x86\_64 --box-version 1.2.0  
  vagrant up

1. **SSH into the VM**:

* Use the command: vagrant ssh
* Switch to root: sudo -i
* To shutdown: vagrant halt
* To destroy the VM: vagrant destroy

1. **Check CentOS Version**:

* Run: rpm -q centos-release

1. **Optional**: Set up port forwarding in VirtualBox for SSH access using port 2222:

* Name: ssh
* Protocol: tcp
* Host IP: 127.0.0.1
* Host Port: 2222
* Guest Port: 22
* You can copy files using SCP:
* scp -P 2222 file\_dir root@127.0.0.1:/root/sample\_file\_dir

1. **Update CentOS repositories**:

* Update the repository in /etc/yum.repos.d/CentOS-Base.repo
* [Fix Yum after CentOS 6 EOL](https://www.getpagespeed.com/server-setup/how-to-fix-yum-after-centos-6-went-eol)
* Run: yum update

1. **Networking Issues**:

* You can use: ifconfig or ip a to view your network settings.

1. **GitHub Repository**:

* Clone the repository: git clone https://github.com/henzelss/maya-assesment.git
* If you need to push changes, generate a GitHub token and use it like this:
* git clone https://<your-token>@github.com/henzelss/maya-assesment.git

**Part 2: Puppet Setup**

1. **Install Necessary Packages**:  
   I installed the following packages using yum:

* vim
* curl
* git
* wget
* tzdata

1. **Create a User**: I created a user named monitor with the home directory /home/monitor and the shell set to /bin/bash.
2. **Create Directories**: I created two directories:

* /home/monitor/scripts/
* /home/monitor/src/

1. **Download Memory Check Script**: The memory check script was downloaded using wget and saved to /home/monitor/scripts/memory\_check.
2. **Create Soft Link**: A soft link named my\_memory\_check was created in /home/monitor/src/ that points to the memory check script in the /home/monitor/scripts/ directory.
3. **Cron Job**: A cron job was set up to run the my\_memory\_check script every 10 minutes.
4. **Set Time Zone**: The time zone was set to Asia/Manila (PHT) using symlink and updated the /etc/sysconfig/clock file.
5. **Set Hostname**: The hostname was set to bpx.server.local.

**Puppet Class Code (deploy-site.pp)**

|  |
| --- |
| Plain Text class deploy-site {   # Install necessary packages using yum  package { ['vim', 'curl', 'git', 'wget', 'tzdata']:  ensure => installed,  provider => 'yum',  }   # Create user "monitor"  user { 'monitor':  ensure => present,  home => '/home/monitor',  shell => '/bin/bash',  managehome => true,  }   # Create script directory  file { '/home/monitor/scripts':  ensure => directory,  owner => 'monitor',  group => 'monitor',  mode => '0755',  }   # Download memory\_check script  exec { 'download\_memory\_check':  command => '/usr/bin/wget -O /home/monitor/scripts/memory\_check https://raw.githubusercontent.com/henzelss/memory-check-script/main/memorycheck.sh',  creates => '/home/monitor/scripts/memory\_check',  require => File['/home/monitor/scripts'],  }   # Create source directory  file { '/home/monitor/src':  ensure => directory,  owner => 'monitor',  group => 'monitor',  mode => '0755',  }   # Create soft link for memory\_check script  file { '/home/monitor/src/my\_memory\_check':  ensure => link,  target => '/home/monitor/scripts/memory\_check',  owner => 'monitor',  group => 'monitor',  }   # Set up cron job to run memory\_check script every 10 minutes  cron { 'memory\_check':  command => '/home/monitor/src/my\_memory\_check',  user => 'monitor',  minute => '\*/10',  }   # Set timezone to PHT  exec { 'set\_timezone':  command => '/bin/ln -sf /usr/share/zoneinfo/Asia/Manila /etc/localtime',  unless => '/usr/bin/test "$(readlink /etc/localtime)" = "/usr/share/zoneinfo/Asia/Manila"',  }   # Ensure /etc/sysconfig/clock has the correct timezone configuration  file { '/etc/sysconfig/clock':  ensure => file,  content => "ZONE=\"Asia/Manila\"\n",  owner => 'root',  group => 'root',  mode => '0644',  }   # Set hostname  exec { 'set\_hostname':  command => '/bin/hostname bpx.server.local',  unless => '/bin/hostname | /bin/grep bpx.server.local',  }  }  include deploy-site |