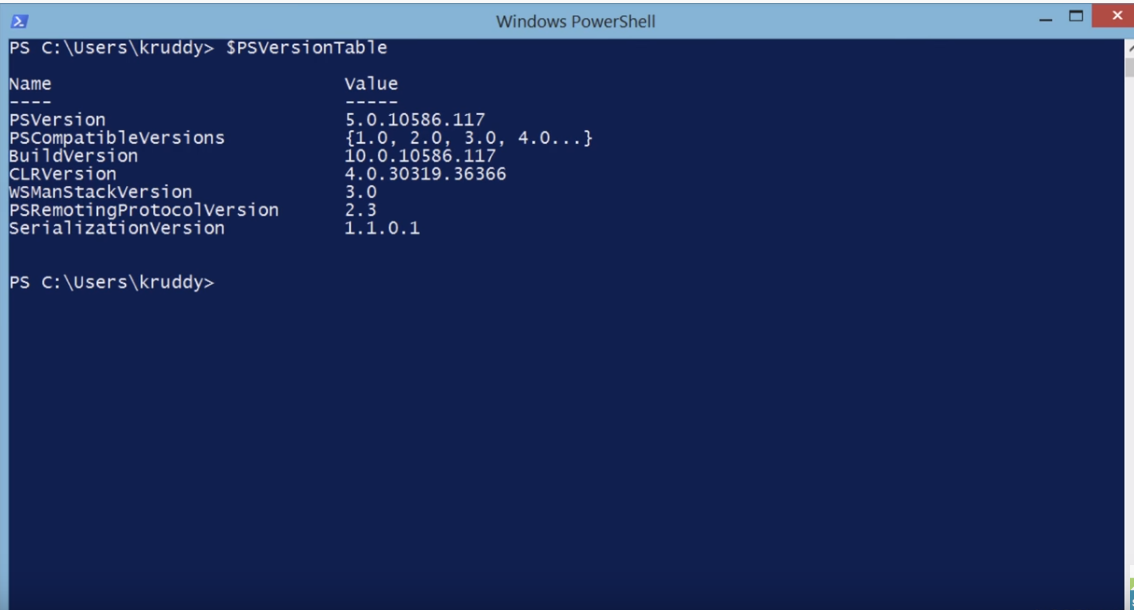
Creation of New VMs in Vcenter By Automation

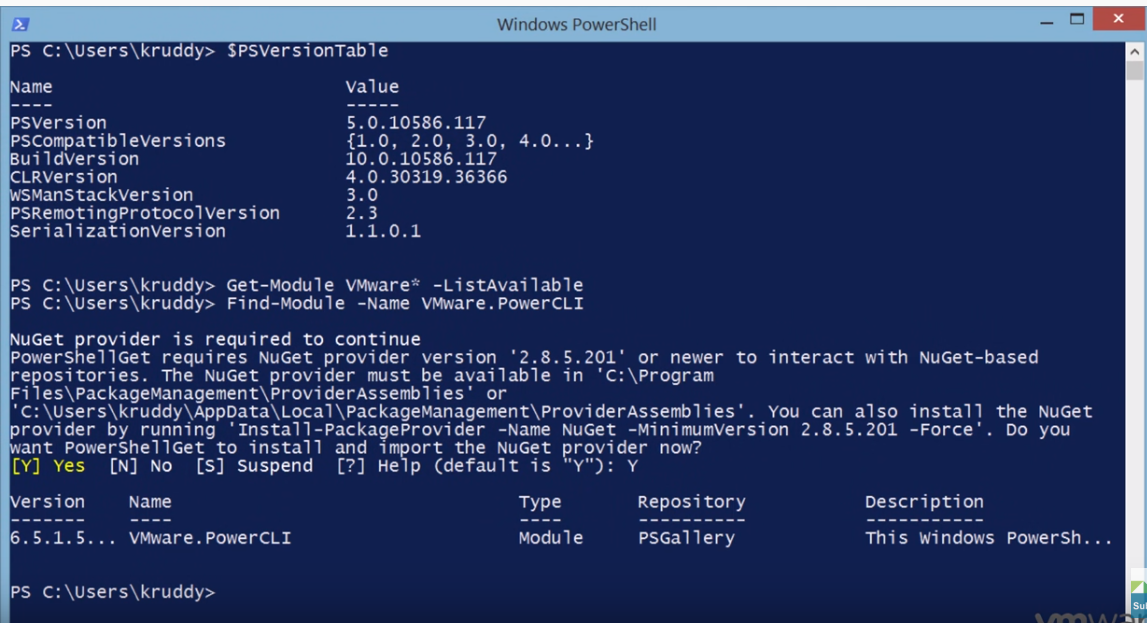
Power Shell CLI Mode

1. Install A Power shell CLI module 6.5 by following Below Steps

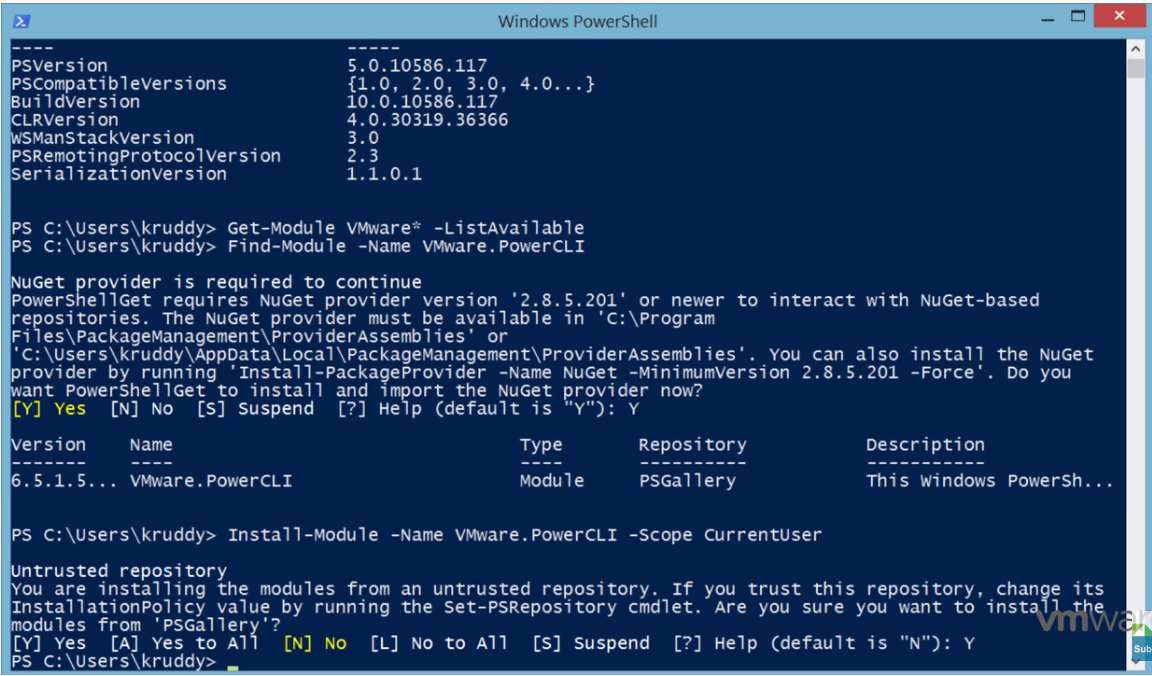
* Open the window Power Shell CLI and check the latest version.



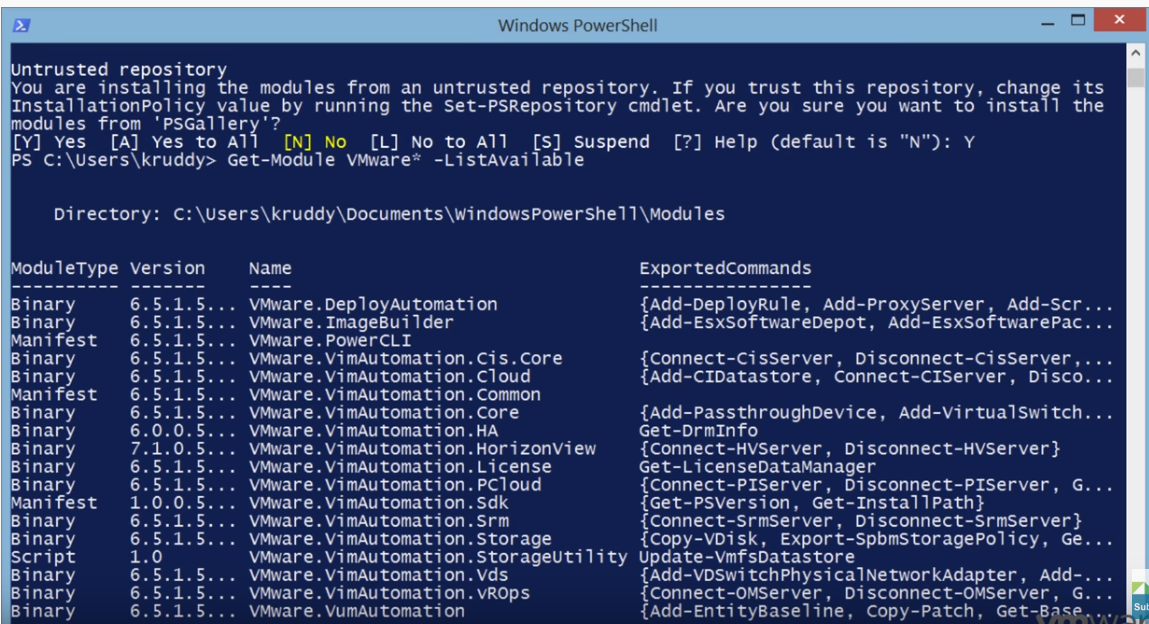
* Find a new Module in power shell Galary using command Find-Module -Name VMware.PowerCLI



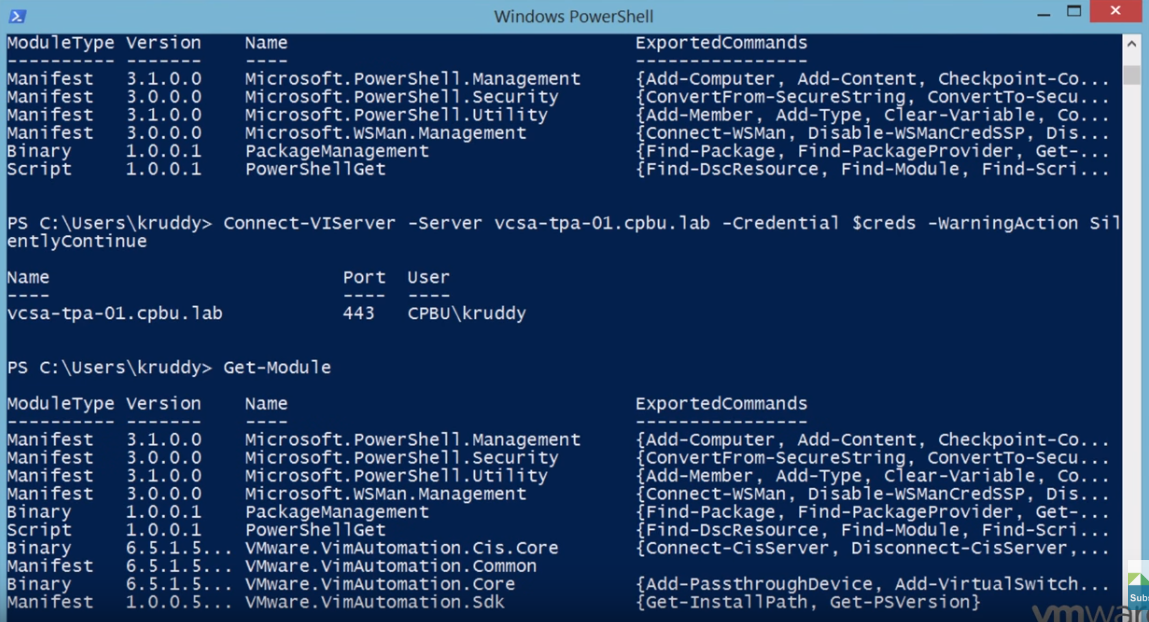
* Now Install a new module using command Install-Module -Name VMware.PowerCLI -Scope CurrentUser



* Now get new module using command Get-Module VMware\* -ListAvailable



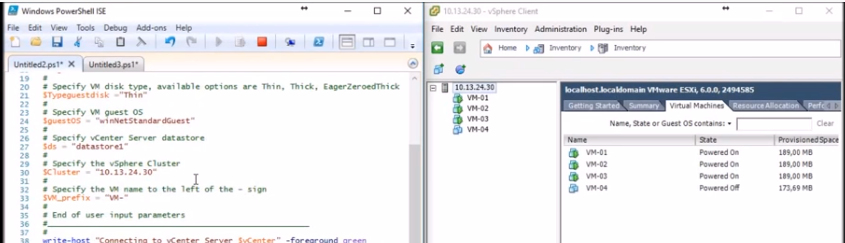
* Then Connect the power shell to Vcenter and also load/Import the new updated Powershell module using this Command connect-VIServer -Server Server name -Credential $creds -WarningAction SilentlyContinue



1. Next step is to write Power shell script for creation of New VMs in Vcenter.



1. Now Run above Script in Power shell and then automatically the New VMs will be installed. On the Other Hand, we can use Window Power shell ISO (Integrated Scripting Environment) which is a graphical host application for windows power shell.



Ansible CM tool Mode

1. Let’s create our inventory list which in this example I will be calling create\_vms\_hosts using below command.

nano create\_vms\_hosts

1. What we are specifying here is the group names (web-vms and db-vms) to specify different variables for each group of servers to build. I will breakdown each variable below to put some meaning to them. ans-web[01:05] and ans-db[01:03] —- These are the actual VM names that we will be creating.

[web-vms]

ans-web[01:05] disk='10' datastore='SSD-Pool\_ELKStack\_dev (NAS01)' network='vSS-OpenStack\_Default' memory='256' cpucount='1' osid='ubuntu64Guest'

[db-vms]

ans-db[01:03] disk='20' datastore='SSD-Pool\_ELKStack\_dev (NAS01)' network='vSS-OpenStack\_Default' memory='1024' cpucount='2' osid='rhel6\_64Guest'

1. Now we need to create the actual ansible playbook to run to create the VMs for us. In this case I will be calling it create\_vms.yml.

nano create\_vms\_hosts

1. Now we will need to do one more thing before running below playbook and that is installing the python module for vSphere.

Sudo pip Install pysphere

1. Now write the playbook as per our requirement

---

- hosts: all

gather\_facts: false

connection: local

user: remote

sudo: true

vars\_prompt:

- name: "vcenter\_hostname"

prompt: "Enter vcenter hostname"

private: no

default: "vcsa"

- name: "vcenter\_user"

prompt: "Enter vcenter username"

private: no

- name: "vcenter\_pass"

prompt: "Enter vcenter password"

private: yes

vars:

datacenter: 'everythingshouldbevirtual'

esxi\_host: 'esxi01.everythingshouldbevirtual.local'

notes: 'Created by Ansible'

tasks:

- vsphere\_guest:

vcenter\_hostname: "{{ vcenter\_hostname }}"

username: "{{ vcenter\_user }}"

password: "{{ vcenter\_pass }}"

guest: "{{ inventory\_hostname }}"

state: present

vm\_extra\_config:

notes: "{{ notes }}"

vm\_disk:

disk1:

size\_gb: "{{ disk }}"

type: thin

datastore: "{{ datastore }}"

vm\_nic:

nic1:

type: vmxnet3

network: "{{ network }}"

network\_type: standard

vm\_hardware:

memory\_mb: "{{ memory }}"

num\_cpus: "{{ cpucount }}"

osid: "{{ osid }}"

scsi: paravirtual

esxi:

datacenter: "{{ datacenter }}"

hostname: "{{ esxi\_host }}"

1. Now Need to run Playbook using below Command

Ansible-playbook -i create\_vms\_hosts create\_vms.yml

Additional

Set-PowerCLIConfiguration -InvalidCertificateAction Ignore

Then use

Set-PowerCLIConfiguration -InvalidCertificateAction Ignore -Confirm:$false

