

Tools & Cattle

One way of doing this

Luc Dekens
Kyle Ruddy



Luc Dekens

PowerCLI Mastermind

Writer @ lucd.info

GitHub @ github.com/lucdekens

Twitter @lucd22

MS MVP Cloud\Datacenter Management





Kyle Ruddy

Senior Architect, Technical Marketing

Writer @ kmruddy.com

GitHub @ github.com/kmruddy

Podcast @ vBrownBag.com

Twitter @kmruddy

MS MVP Cloud\Datacenter Management





Intro

- Where to develop & run your PS scripts?
- PS Core opens new possibilities
- Use and dispose (cattle)
- Use VMware vSphere platform
 - On premises
 - Choice of cloud
 - PowerShell + VMware PowerCLI
- Account of a project
- ... with all the bumps in the road!



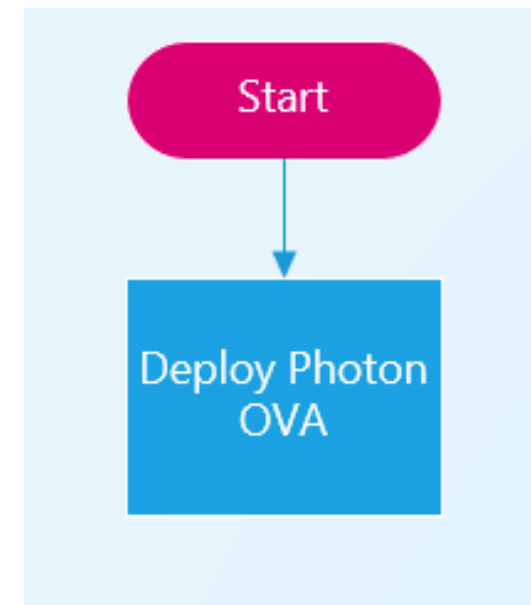
Linux

- Practical advantages
- Cloud standard
- Existing deployment methods
 - Tested
 - Documented
- Handsfree!



Initial Deployment

- Station with PS + VMware PowerCLI
- vSphere vCenter
- OVA format
- Import-VApp





```
$ovaPath = 'D:\Repo\VMware\Photon\photon-hw13_uefi-3.0-26156e2.ova'

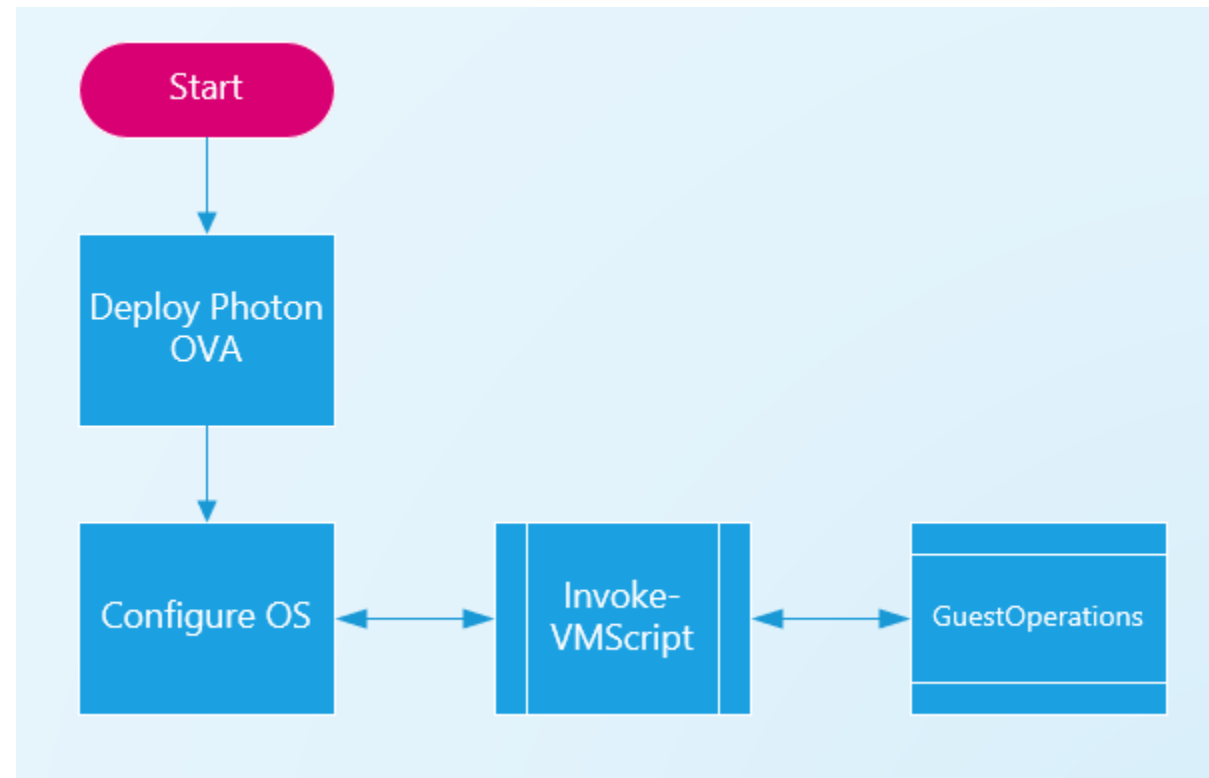
$ovfParm = Get-OvfConfiguration -Ovf $ovaPath
$ovfParm.NetworkMapping.None.Value = $pg

$sVApp = @{
    Name = 'PhotonVM'
    Source = $ovaPath
    OvfConfiguration = $ovfParm
    VMHost = $esx
    Datastore = $ds
    DiskStorageFormat = 'Thin'
    Confirm = $false
}

$vm = Import-VApp @sVApp
```

OS Customization

- Target not (yet) on network
- Invoke-VMScript cmdlet
 - API: GuestOperations
 - Requires VMware Tools
 - Port 902 to ESXi node hosting the VM
 - Account + privileges in guest OS



GuestOperations

- vSphere Web Services API

Managed Object - GuestOperationsManager(vim.vm.guest.GuestOperationsManager)

Property of
[ServiceContent](#)

See also
[GuestAliasManager](#), [GuestAuthManager](#), [GuestFileManager](#), [GuestProcessManager](#), [GuestWindowsRegistryManager](#)

Since
vSphere API 5.0

Managed Object Description

GuestOperationsManager is the managed object that provides APIs to manipulate the guest operating system files and process. Each class of APIs is separated into its own manager. Only one guest operation is allowed at a time per virtual machine.

Properties

| NAME | TYPE | DESCRIPTION |
|--|---|---|
| aliasManager [*] ^P | ManagedObjectReference to a GuestAliasManager | A managed object that provides methods to support single sign-on in the guest operating system. Since vSphere API 6.0 |
| authManager [*] ^P | ManagedObjectReference to a GuestAuthManager | A singleton managed object that provides methods for guest authentication operations. |
| fileManager [*] ^P | ManagedObjectReference to a GuestFileManager | A singleton managed object that provides methods for guest file operations. |
| guestWindowsRegistryManager [*] ^P | ManagedObjectReference to a GuestWindowsRegistryManager | A singleton managed object that provides methods for guest windows registry operations. Since vSphere API 6.0 |
| processManager [*] ^P | ManagedObjectReference to a GuestProcessManager | A singleton managed object that provides methods for guest process operations. |

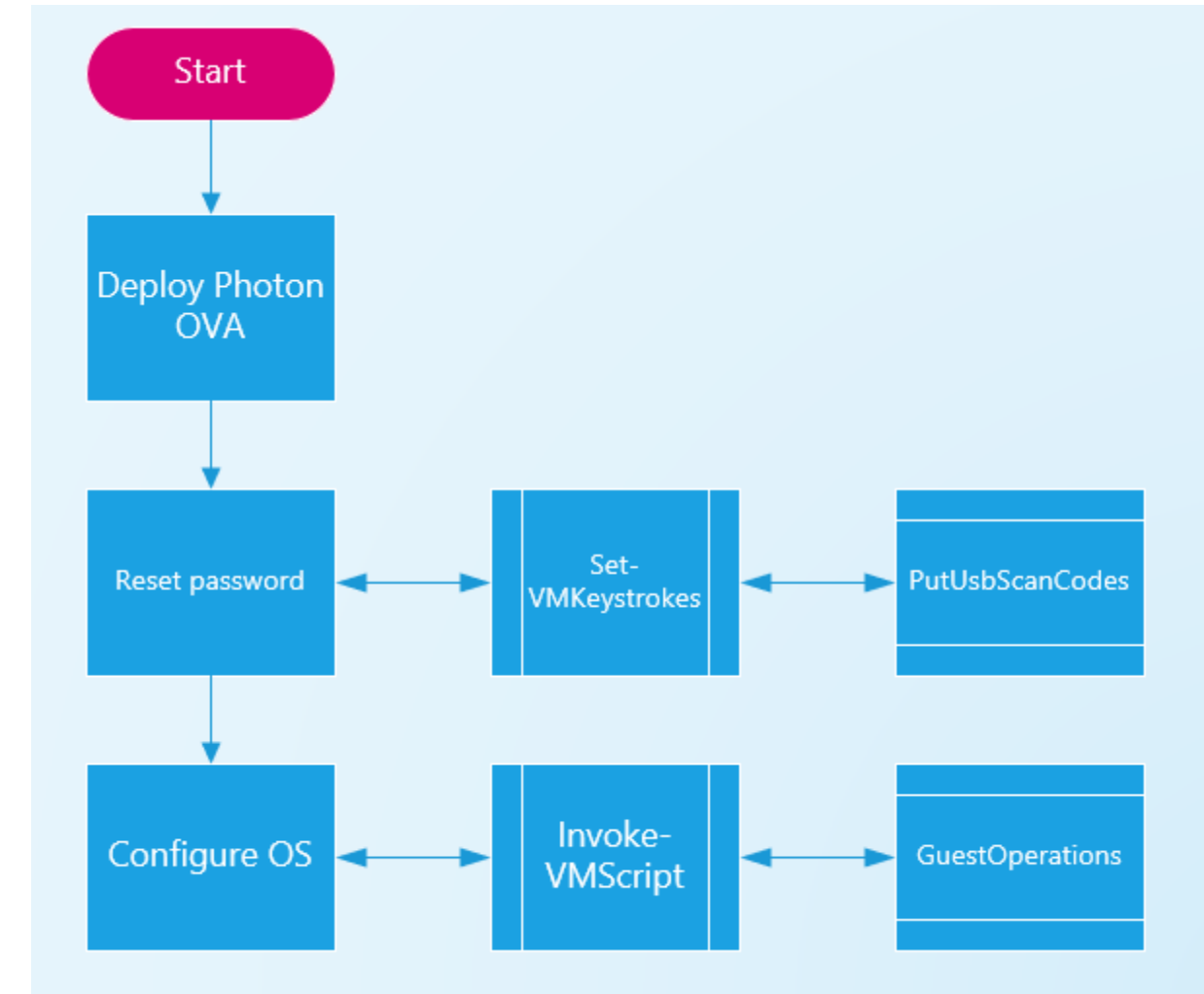


```
$sInvoke = @{  
    VM = $vm  
    ScriptType = 'bash'  
    ScriptText = 'whoami'  
    GuestCredential = $cred  
}
```

```
Invoke-VMScript @$sInvoke
```

The chicken & the egg

- Photon OS requires immediate password change
- Blocks use of Invoke-VMScript
- Solution:
 - Emulate keyboard
 - William Lam's Set-VMKeystrokes function
 - API: PutUsbScanCodes



PutUsbScanCodes(putUsbScanCodes)

Inject a sequence of USB HID scan codes into the keyboard.

Required Privileges

VirtualMachine.Interact.PutUsbScanCodes

Since

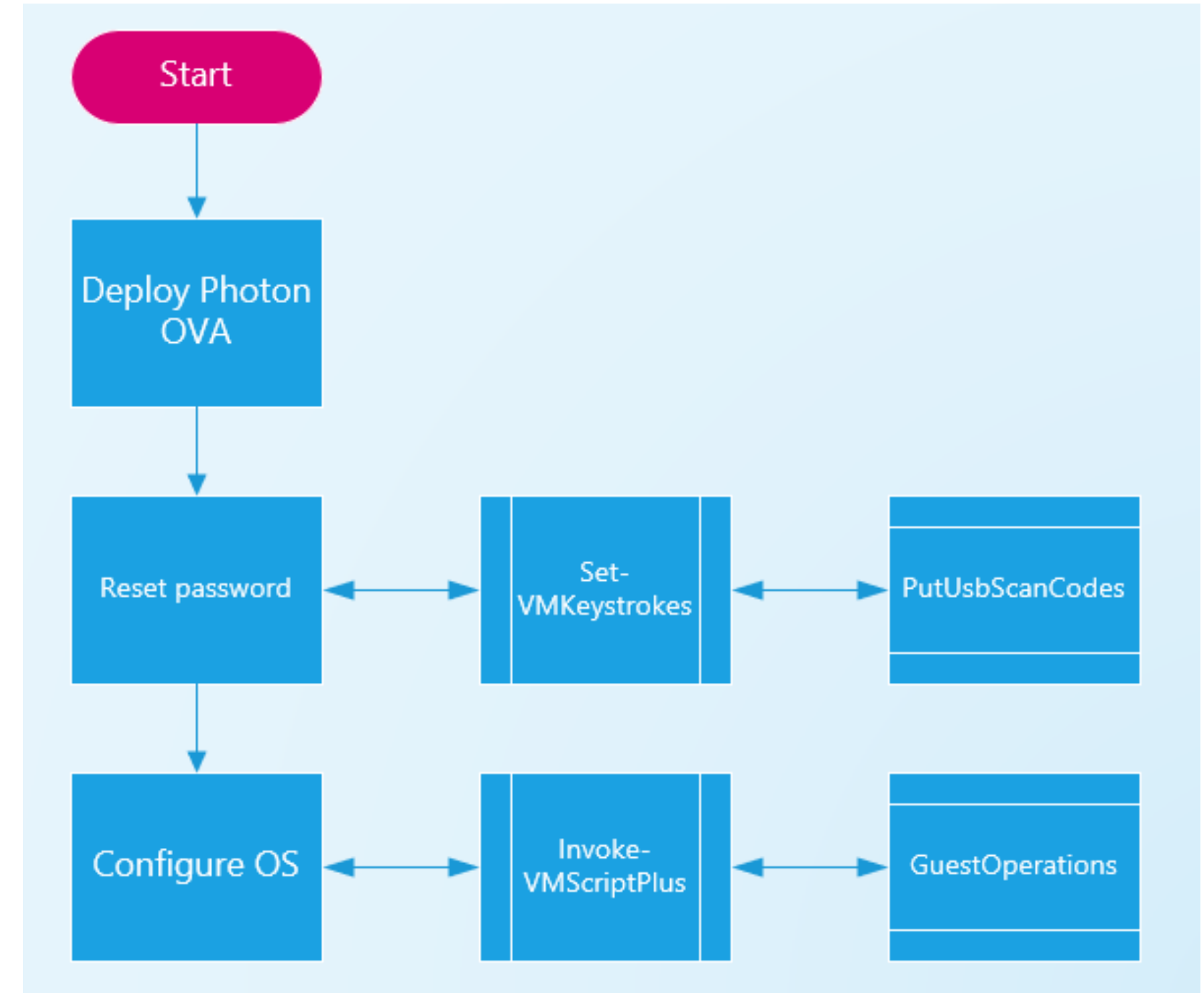
vSphere API 6.5

Parameters

| NAME | TYPE | DESCRIPTION |
|-------|--|---|
| _this | ManagedObjectReference | A reference to the VirtualMachine used to make the method call. |
| spec | UsbScanCodeSpec | |

Linux code & Invoke-VMScript

- Developed pre-PS Core
 - Windows!
- Limited Linux functionality
 - No she-bang support (`#!/usr/bin/env bash`)
 - No here document support
 - Limited size for bash (± 2600 chars)
- Invoke-VMScriptPlus function
 - lucd.info
 - PS Core support
 - Side-by-side support
 - ...

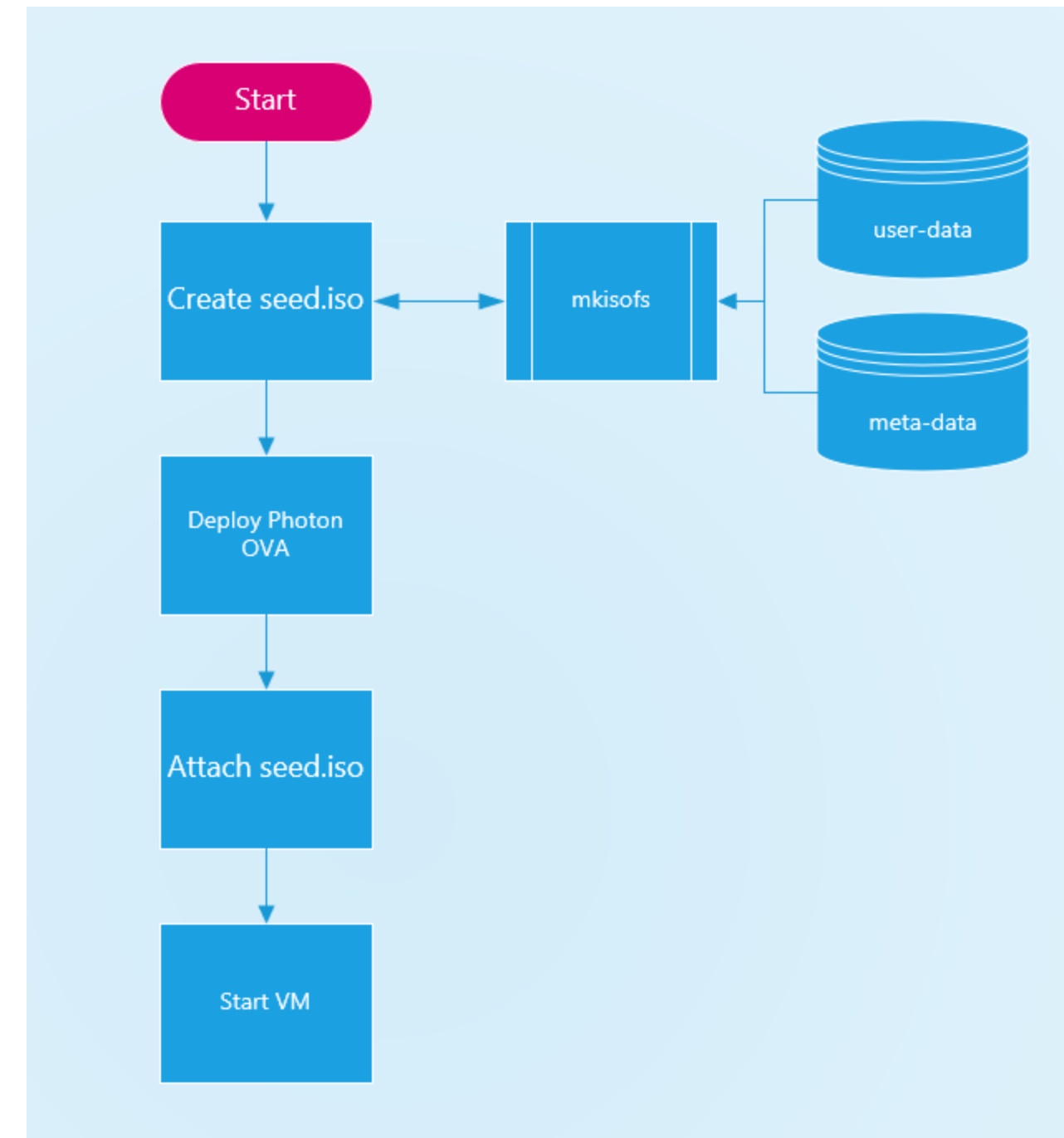




```
$code = '@'  
tdnf update open-vm-tools -y  
/bin/vmware-toolbox-cmd -v  
'@  
  
$sInvoke = @{  
    VM = $vm  
    GuestCredential = $cred  
    ScriptType = 'bash'  
    ScriptText = $code  
    GuestOSType = 'Linux'  
}  
Invoke-VMScriptPlus @sInvoke
```

Cloud friendly?

- Customisation scripts ← Not standardized
- Alternative: cloud-init
- Test with stand-alone datasource
- OVA + seed.iso
 - Seed.iso requires mkisofs
 - Work Photon station





Root password (again)

- The 'normal' way in cloud-init
- Known issue:

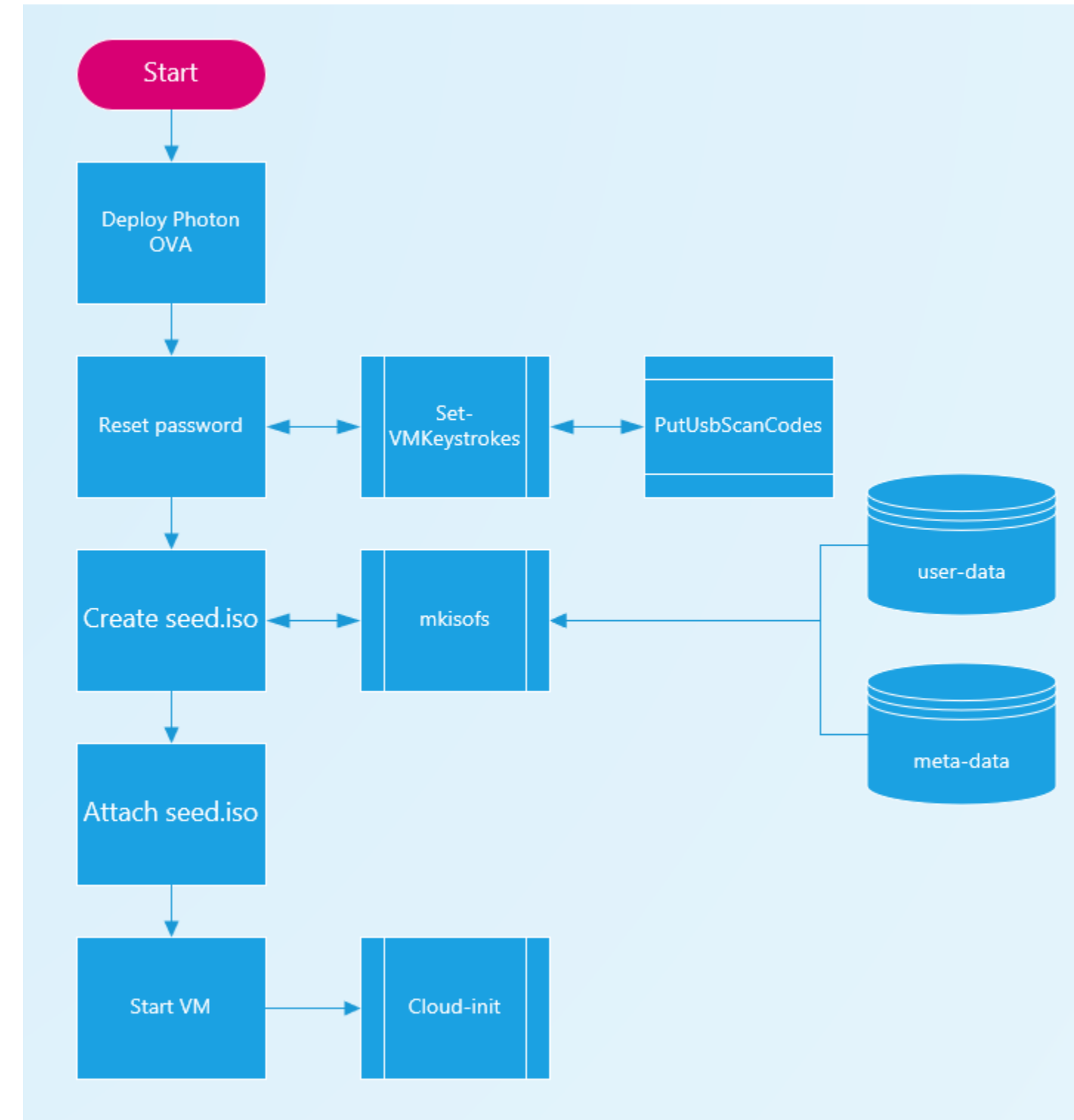
```
#cloud-config  
chpasswd:  
  list: |  
    root:password  
  expire: False
```

```
2019-09-14 13:46:56,169 - helpers.py[DEBUG]: Running config-users-groups using lock (<FileLock using file '/var/lib/cloud/instances/iid-datasource-none/sem/config_users_groups'>)  
2019-09-14 13:46:56,169 - __init__.py[INFO]: User root already exists, skipping.  
2019-09-14 13:46:56,169 - handlers.py[DEBUG]: finish: init-network/config-users-groups: SUCCESS: config-users-groups ran successfully
```

User root already exists, skipping.

Keystrokes (again)

- Deploy OVA
- Use William's Set-VMKeyStrokes
- Attach seed.iso
- Reboot (triggers cloud-init)



The PS Core part 1

- Package in tdnf for 'powershell'
 - Version 6.1.0
 - Limited set of modules
 - No way to install modules
- Alternative: install tar from MSFT repo
 - Not compiled with Photon libraries
 - No PSRepository ← No VMware.PowerCLI
 - Caused by nuget.exe (32 bit)
 - MSFT proposes [mono as solution](#)
 - Alias for nuget.exe

```
PS /root> get-module -listavailable
```

```
Directory: /usr/lib/powershell/Modules
```

| ModuleType | Version | Name | PSEdition |
|------------|---------|---------------------------------|-----------|
| Manifest | 6.1.0.0 | Microsoft.PowerShell.Host | Core |
| Manifest | 6.1.0.0 | Microsoft.PowerShell.Management | Core |
| Manifest | 6.1.0.0 | Microsoft.PowerShell.Security | Core |
| Manifest | 6.1.0.0 | Microsoft.PowerShell.Utility | Core |
| Script | 0.0 | PSDesiredStateConfiguration | Desk |

3. Create an alias by adding the following script to the appropriate file for your OS (typically `~/.bash_aliases` or `~/.bash_profile`):

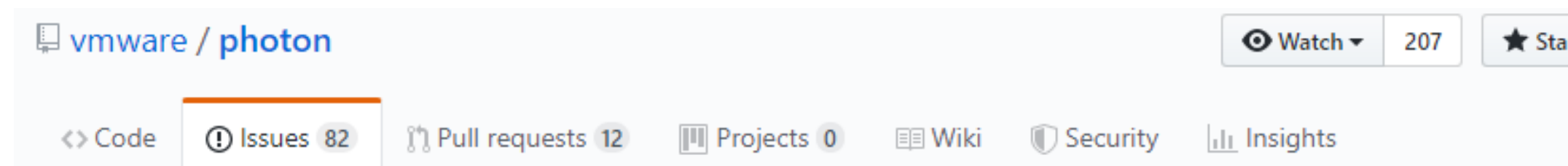
```
bash
```

Copy

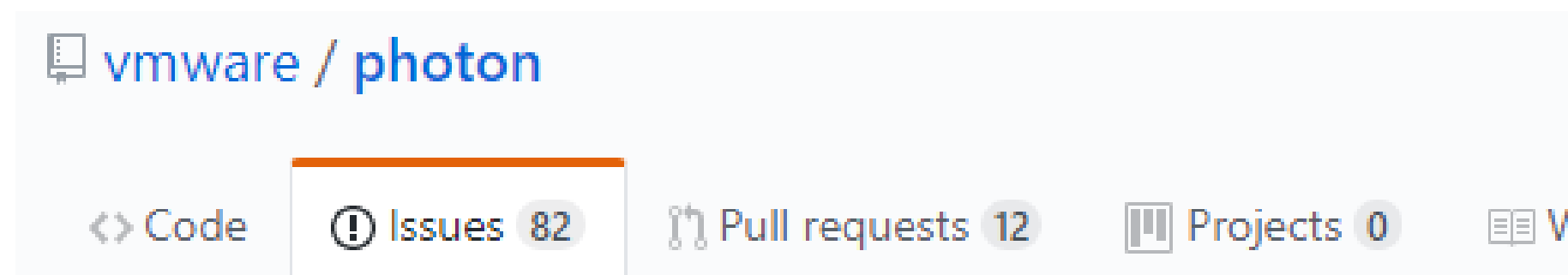
```
# Create as alias for nuget
alias nuget="mono /usr/local/bin/nuget.exe"
```

The PS Core part 2

- Issues raised in Photon
 - Express your support



PowerShell on Photon does not include PowerShellGet or PackageManagement #815

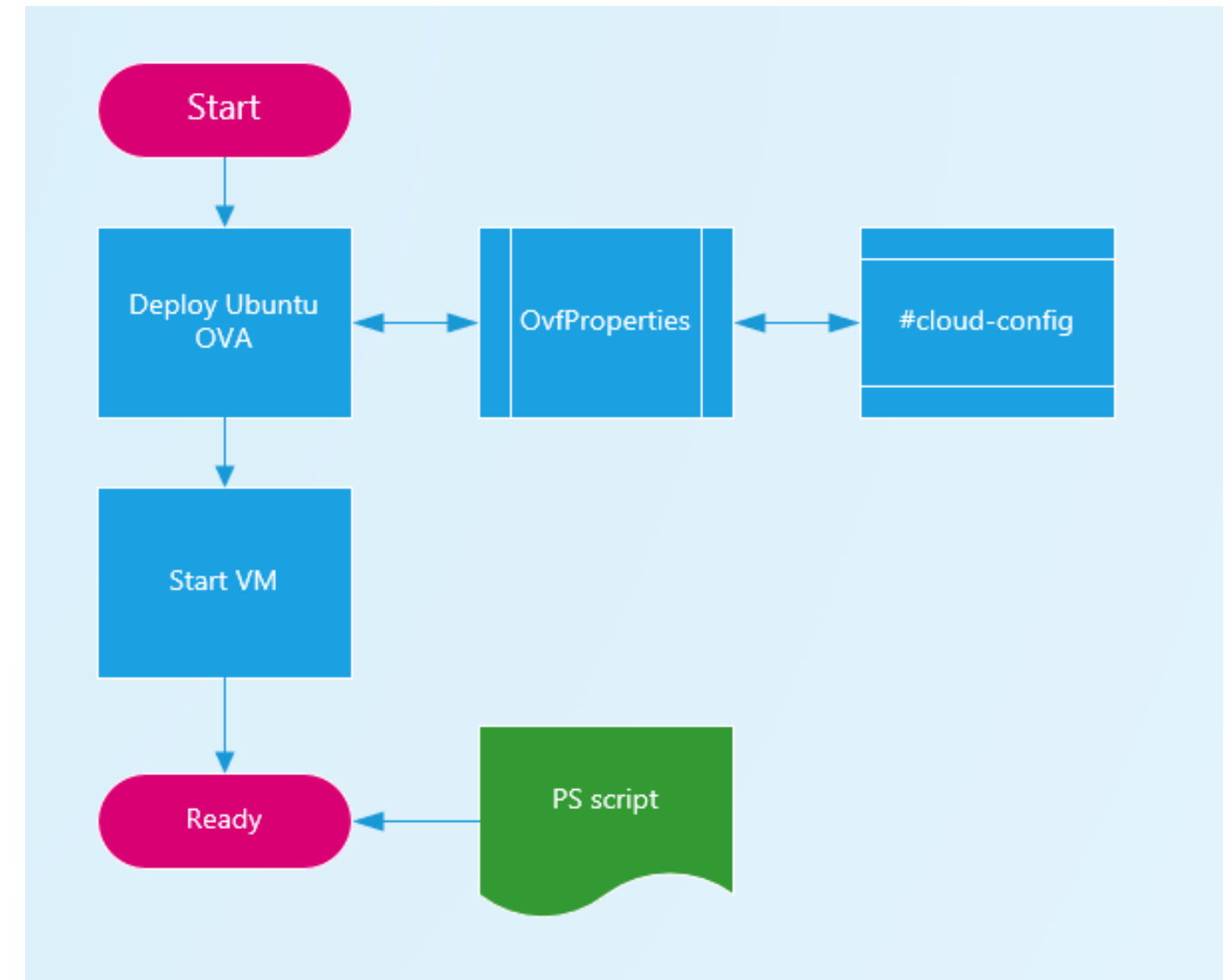


Update powershell package #891

Open lucdekens opened this issue on Apr 11 · 2 comments

Working alternative

- Docker
 - MSFT provides PS Core (on Ubuntu)
- But then why Photon?
- Throw away all our experience?
- Not really
 - Gained cloud-init experience
 - Ubuntu OVA
- OVF property (**Common.user_data.Value**)
 - No need for seed.iso

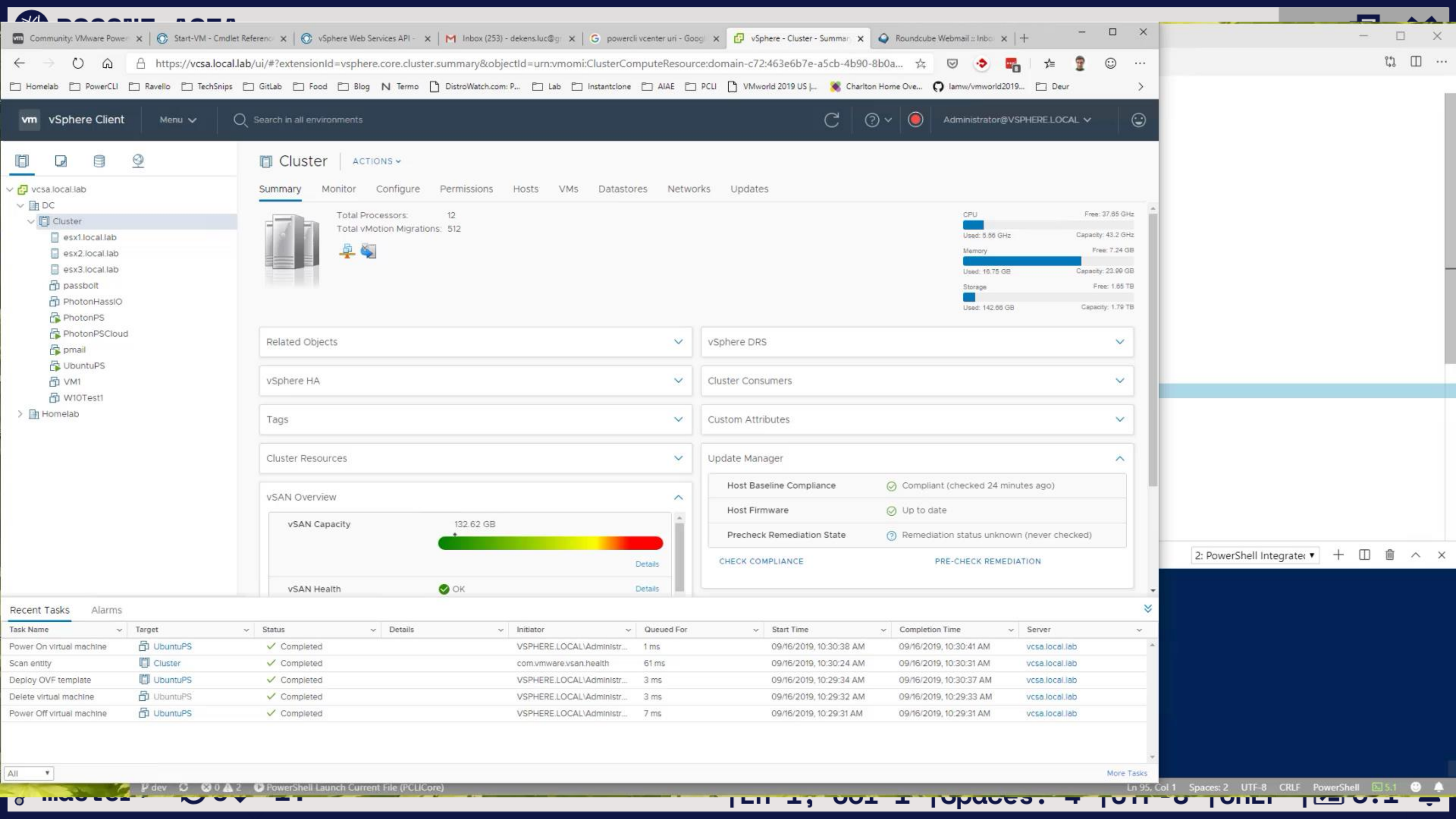




Deploy VM when needed

- Scheduled scripts
- Development platform
 - Add components: VSC, git, ...
- Advantages
 - Resources only used when needed (not another 5% VM)
 - No maintenance/patching
 - Scale (1.. ∞) \leftarrow same script/same platform
 - SDDC
 - No manual intervention
 - Version control

[illegible]





Sample Code

```
$code = @'
if (-not (Get-Module -Name VMware.PowerCLI -ListAvailable))
{
    Install-Module -Name VMware.PowerCLI -Scope CurrentUser -Force -Confirm:`$false
}

Set-PowerCLIConfiguration -InvalidCertificateAction Ignore -Confirm:`$false | Out-Null
Connect-VIServer -Server vcsa.local.lab -Session $($global:DefaultVIServer.SessionSecret) | Out-Null

Get-VMHost |
Select-Object Name, ConnectionState, PowerState, NumCpu, MemoryTotalGB, Model, Version
'@

$sInvoke = @{
    VM = 'UbuntuPS'
    ScriptType = 'powershellv6'
    ScriptText = $ExecutionContext.InvokeCommand.ExpandString($code)
    GuestUser = 'luc'
    GuestPassword = ConvertTo-SecureString -String 'VMware1!' -AsPlainText -Force
    GuestOSType = 'Linux'
}
Invoke-VMScriptPlus @sInvoke
```

Sample Output

```
Name      : homelab.local.lab
ConnectionState : Connected
PowerState  : PoweredOn
NumCpu      : 8
MemoryTotalGB : 63.910800933837890625
Model       : B360 HD3P-LM
Version     : 6.7.0
```

```
Name      : esx3.local.lab
ConnectionState : Connected
PowerState  : PoweredOn
NumCpu      : 4
MemoryTotalGB : 7.995532989501953125
Model       : VMware Virtual Platform
Version     : 6.7.0
```

ESXi node report



From vSphere@local.lab
To luc@local.lab
Date Today 00:49

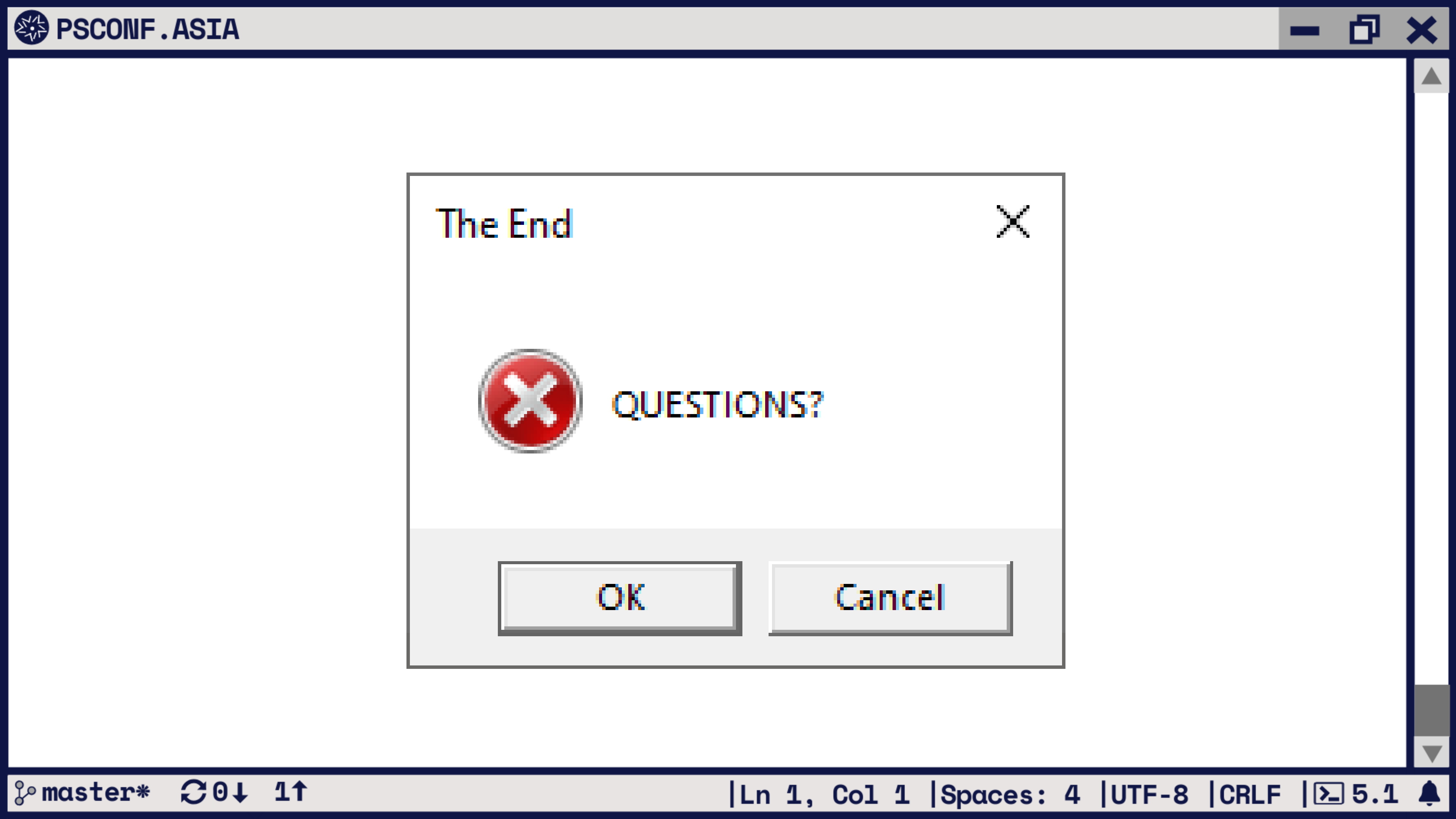
ESXi node report

| Name | ConnectionState | PowerState | NumCpu | MemoryTotalGB | Model | Version |
|-------------------|-----------------|------------|--------|-----------------------|-------------------------|---------|
| homelab.local.lab | Connected | PoweredOn | 8 | 63.910800933837890625 | B360 HD3P-LM | 6.7.0 |
| esx3.local.lab | Connected | PoweredOn | 4 | 7.995532989501953125 | VMware Virtual Platform | 6.7.0 |
| esx2.local.lab | Connected | PoweredOn | 4 | 7.995532989501953125 | VMware Virtual Platform | 6.7.0 |
| esx1.local.lab | Connected | PoweredOn | 4 | 7.9955291748046875 | VMware Virtual Platform | 6.7.0 |

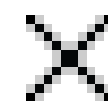


Key takeaways

- Everything is “posh”able!
- Failure is not failure!
- Adopt PS Core
 - Linux flavours
 - Broaden your horizon
- Have fun!!!



The End



QUESTIONS?

OK

Cancel



t to rate the ses

<https://my.eventraft.com/psconfasi>

Opening Ceremony

Fri, 20 Sep, 08:45 AM - 09:00 AM

Beginner

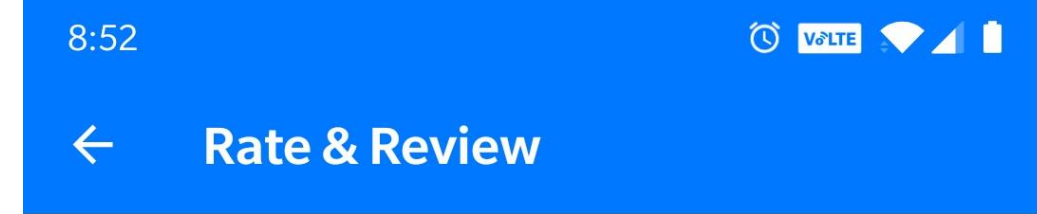
Common

☆☆☆☆☆ 0 Rating

Rate now >

Description

PowerShell Conference Asia 2019 Opening ceremony!



Rate & Review

Tap to Rate



Write a review (optional)

Review Anonymously ☐

SUBMIT

