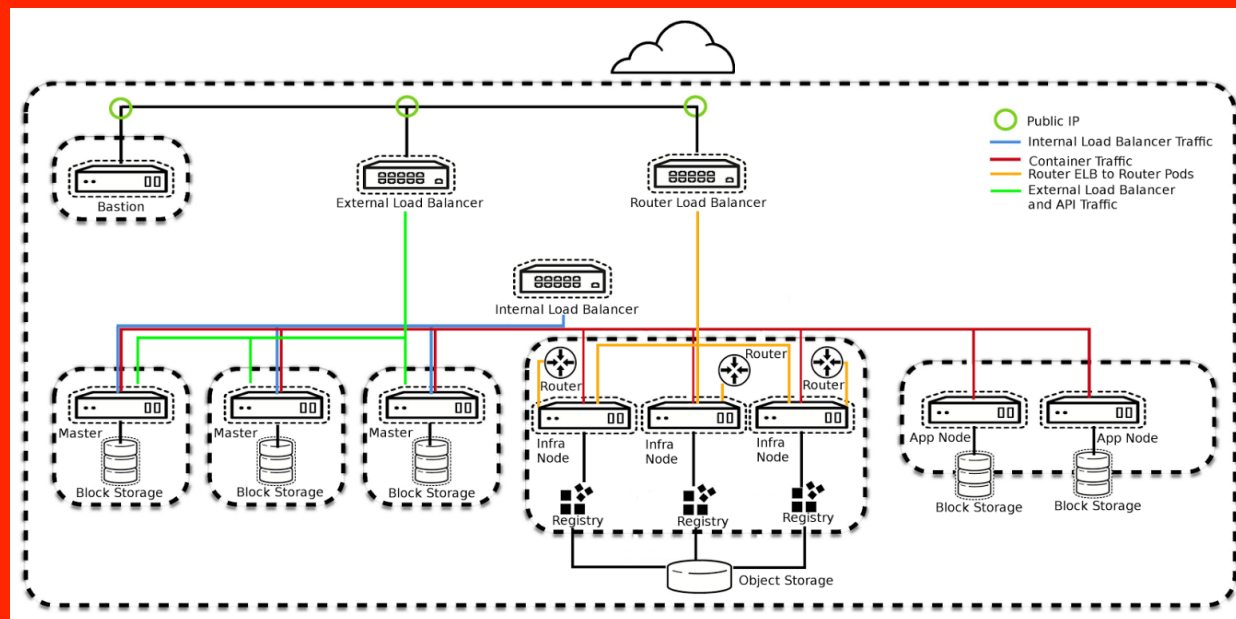


OpenShift Hybrid Windows and Linux



Glenn West, Principle Engineer

Version 1 – April 23 2018











Challenges/Questions

- How to deploy Windows nodes in a OpenShift Cluster?
- What is needed to do this?
- How to make it Simple.
- Is the Microsoft Doc good enough?
- What are the issues?

Infrastructure

- Cluster Deployed Directly On Azure
- 3 Masters – 3 Infrastructure VM's – 1 Rhel Node – 2 Windows Node
- Standard Windows 2016 Datacenter Server Edition (No Patches)
- RHEL 7.4 with latest patches (Auto Applied)

Azure Portal of OCP Hybrid Cluster

<input type="checkbox"/>	NAME <small>↑↓</small>	TYPE <small>↑↓</small>	LOCATION <small>↑↓</small>
<input type="checkbox"/>	 bastion	Virtual machine	South Central US
<input type="checkbox"/>	 infranode1	Virtual machine	South Central US
<input type="checkbox"/>	 infranode2	Virtual machine	South Central US
<input type="checkbox"/>	 infranode3	Virtual machine	South Central US
<input type="checkbox"/>	 master1	Virtual machine	South Central US
<input type="checkbox"/>	 master2	Virtual machine	South Central US
<input type="checkbox"/>	 master3	Virtual machine	South Central US
<input type="checkbox"/>	 node01	Virtual machine	South Central US
<input checked="" type="checkbox"/>	 winnode01	Virtual machine	South Central US
<input type="checkbox"/>	 winnode02	Virtual machine	South Central US

Software Components

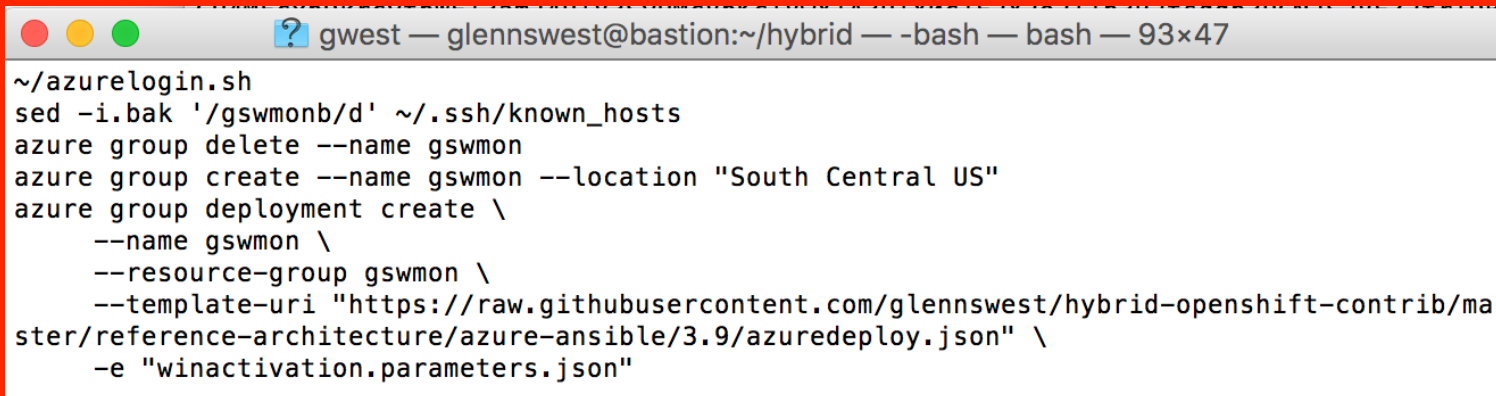
- A Bash script to deploy a cluster.
- A Container based repo for 3.9 puddles of ocp.
- A ARM based(Azure Resoure Manager) script to deploy all virtual infrastructure on Azure including Linux and Windows nodes
- A Bash/Ansible Script that fully installs ocp 3.9 HA cluster with AV Sets, Load Balancers, 1-32 linux compute,
- A Separate Ansible script that installs 1-32 windows nodes
- Simple script to tunnel to Windows nodes via bastion for RDP
- Various Powershell scripts

Interesting Parts

- No Code Changes to OpenShift
- No Code Changes in Openshift Ansible Installer
- No Special Version of OCP 3.9 – Standard Puddle – Expect GA Bits to be fine
- Few additional RPM's to support Ansible to Windows nodes
- Just add rdp client to work on windows nodes via ssh to bastion host
- Existing security and firewall all match existing ref arch best practice
- Begin to end of deploy estimate 45 minutes for full HA multi-node cluster

How hard is it to repeat

- Need a proper Azure account with good resources available
 - Mark Heslin controls this access
- Run a simple script (Assuming the Azure CLI Is installed)
- Forked from 3.6/3.7 Ref arch – Formal doc and theory is same

A terminal window with a title bar showing three colored circles (red, yellow, green) and a text label "gwest — glennswest@bastion:~/hybrid — -bash — bash — 93x47". The terminal content shows a script being executed, starting with "~/azurelogin.sh" and followed by several Azure CLI commands to delete and create a resource group, and then create a deployment.

```
~/azurelogin.sh
sed -i.bak '/gswmonb/d' ~/.ssh/known_hosts
azure group delete --name gswmon
azure group create --name gswmon --location "South Central US"
azure group deployment create \
  --name gswmon \
  --resource-group gswmon \
  --template-uri "https://raw.githubusercontent.com/glennswest/hybrid-openshift-contrib/master/reference-architecture/azure-ansible/3.9/azuredeploy.json" \
  -e "winactivation.parameters.json"
```

Walk thru of changes - Baseline

- Forked Azure ref arch from existing 3.7 work (Official)
 - Removed all unrelated files/sub projects
- Did a container based repo of puddles using Nginx for 3.9 OCP
- Upgraded the existing ref arch to handle 3.9 differences.
- Added new variables to ref arch to support Windows Node Count
- Added new sub-arm script for windows node deployment
- Debugged 3.9 setup and make sure cluster comes up

Windows Specific using Ansible – Part 1

- Added a new ansible group for Windows
- Added ansible config file (Group-vars) to allow ansible to control windows nodes
- Developed a new Ansible Script to Setup the Windows Nodes
- Original Microsoft uses a bootstrap token, we use a pre-generation of all certs/config files instead. The bootstrap token is not recommended even upstream till 3.10 at soonest, tbd on ocp
- Implemented the pre-generation and deployment of configs and nodes

Windows Specific using Ansible – Part 2

- Install Kubernetes on windows - No Package management ☹️
- No ETC – So create directory structure (c:\k) and setup environment variables (Powershell in ansible)
- Download and install cni – No Package Management ☹️
- Download and install windows package manager
- Download and install git
- Generate OCP Config files on Master
- Pull configs to bastion
- Push config to each Node
- Start Kubelet on Windows (Currently done manually)

Objectives Accomplished

- Allow fully automated – repeatable install of Linux and Windows VM ready for OCP
- Installs OCP 3.9 puddles using a local container based repo on the bastion (No breaks)
- Windows nodes part of OCP (Kubernetes is installed enough to function and report to Cluster and stays up

```
glennswest@bastion:~/hybrid — s
[glennswest@bastion hybrid]$ oc get nodes
NAME          STATUS          ROLES    AGE    VERSION
infranode1    Ready           <none>   10d    v1.9.1+a0ce1bc657
infranode2    Ready           <none>   10d    v1.9.1+a0ce1bc657
infranode3    Ready           <none>   10d    v1.9.1+a0ce1bc657
master1       Ready,SchedulingDisabled <none>   10d    v1.9.1+a0ce1bc657
master2       Ready,SchedulingDisabled <none>   10d    v1.9.1+a0ce1bc657
master3       Ready,SchedulingDisabled <none>   10d    v1.9.1+a0ce1bc657
node01        Ready           <none>   10d    v1.9.1+a0ce1bc657
winnode01     Ready           <none>   20m    v1.9.3
[glennswest@bastion hybrid]$
```

To Be done:

- 1 Day – Make sure everything has been captured and is repeatable
- 2 Days +/- Add Networking Setup and Validation
- 1 Day +/- Application Validation
- Demo Day (This slide deck will expand)

OCP Hybrid Repo (ARM/Bash/Ansible)

The screenshot shows the GitHub interface for the repository 'glennswest / hybrid-openshift-contrib'. The browser address bar shows the URL 'https://github.com/glennswest/hybrid-openshift-contrib'. The repository page includes a header with navigation links (Pull requests, Issues, Marketplace, Explore) and repository statistics (1 pull request, 0 issues, 0 projects, 0 wiki, 0 insights, 0 settings). The repository description is 'No description, website, or topics provided.' Below this, a bar shows 38 commits, 1 branch, 0 releases, and 1 contributor. A section for recent commits lists three entries: 'gwestredhat Add windows scripts for startup' (2 days ago), 'reference-architecture Add windows scripts for startup' (2 days ago), and '.gitignore Add windows node' (16 days ago). At the bottom, there is a prompt to 'Add a README'.

Secure | <https://github.com/glennswest/hybrid-openshift-contrib>

This repository Search Pull requests Issues Marketplace Explore

glennswest / hybrid-openshift-contrib Unwatch 1 Star 0 Fork 0

<> Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

No description, website, or topics provided. Edit

Add topics

38 commits 1 branch 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find file Clone or download

gwestredhat Add windows scripts for startup Latest commit 0fd3b76 2 days ago

reference-architecture Add windows scripts for startup 2 days ago

.gitignore Add windows node 16 days ago

Help people interested in this repository understand your project by adding a README. Add a README

Hybrid Script – Windows Node Setup

The screenshot shows a web browser displaying the GitHub repository page for 'glennswest / hybrid'. The browser's address bar shows the URL 'https://github.com/glennswest/hybrid'. The repository page includes a header with the GitHub logo, a search bar, and navigation links for 'Pull requests', 'Issues', 'Marketplace', and 'Explore'. Below the header, the repository name 'glennswest / hybrid' is displayed, along with statistics: 1 Unwatch, 0 Stars, and 0 Forks. A tab bar shows 'Code' as the active tab, with other tabs for 'Issues', 'Pull requests', 'Projects', 'Wiki', 'Insights', and 'Settings'. The main content area shows the repository name 'glennswest / hybrid' and a description 'Openshift Windows Node Setup Script'. Below this, there are statistics: 4 commits, 1 branch, 0 releases, and 1 contributor. A row of buttons includes 'Branch: master', 'New pull request', 'Create new file', 'Upload files', 'Find file', and 'Clone or download'. A list of files and folders is shown, including 'rsc7 Remove windows.retry', 'group_vars', '.gitignore', and 'windows.yml'. At the bottom, there is a prompt to 'Add a README'.

Secure | <https://github.com/glennswest/hybrid>

This repository Search Pull requests Issues Marketplace Explore

glennswest / hybrid Unwatch 1 Star 0 Fork 0

<> Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

Openshift Windows Node Setup Script Edit

Add topics

4 commits 1 branch 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find file Clone or download

rsc7 Remove windows.retry Latest commit 2750fb7 a day ago

group_vars	Add groupvars example	a day ago
.gitignore	Remove windows.retry	a day ago
windows.yml	Try pushing the full system:node:master1 dir	a day ago

Help people interested in this repository understand your project by adding a README. Add a README

Findings

- Microsoft Doc is not usable by typical RedHatter or Customer (Even advanced)
 - The bootstrap token / certs alone invalidates the Document
- The lack of package management is really lacking in polish
 - Consider RedHat Packaging into Chocolatey to get consistent Windows nodes working with OCP
- Recommend we take the “Hybrid” script functionality and build directly into Ansible Installer (Prepped for that)
- Consider a “Ref Arch” for OCP Hybrid (Basically this work upgraded to GA/Tech Preview Bits and a fork of the Ref Arch Doc)
- Technology Side does appear to work – lots more testing is needed

Going Forward

- Complete the networking and basic app test (It's already in my trello card) – 5 days +/-
- Checkout how Persistent Volumes work
- Trial and Feed back this as a POC
- Do the DOC and productise the bits/package to Tech Preview Status
- Test it and Push it to Supportable
- Consider supporting multiple network types (Quite a few coming)
- Consider multiple cloud providers (Beyond Azure)