Deep Dive of Windows Containers on Kubernetes

Pengfei Ni
Microsoft Azure
Kubernetes Maintainer
1. Windows Containers
What is a container

Containers = operating system virtualization

Traditional virtual machines = hardware virtualization
Why containers

- Fast iteration
- Agile delivery
- Immutability
- Cost savings
- Efficient deployment
- Elastic bursting

For developers
For IT
What inside a container

#Our base image
FROM microsoft/windowsservercore:1803
#Enable IIS
RUN powershell -Command Add-WindowsFeature Web-Server
#Declare a default action that will run each time a container starts.
CMD [ "ping", "localhost", ":t" ]
Windows Base Images

<table>
<thead>
<tr>
<th></th>
<th>nanoserver</th>
<th>windowsservercore</th>
<th>Windows</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Born in the cloud applications</td>
<td></td>
<td></td>
<td>Automation workloads</td>
<td></td>
</tr>
<tr>
<td>.NET core support</td>
<td></td>
<td></td>
<td>Most Windows OS components</td>
<td></td>
</tr>
<tr>
<td>93MB</td>
<td></td>
<td></td>
<td>3.4GB</td>
<td></td>
</tr>
<tr>
<td></td>
<td>App Compat</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Full .NET framework support</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.4GB</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

microsoft/windowsservercore:1803 ➔ mcr.microsoft.com/windows/nanoserver:1803
### Container Layers

**Directory:** C:\ProgramData\docker\windowsfilter\646c94c76da839f088e2e4a5209100496d7f4ae5b01802fa6a9209f530661293

<table>
<thead>
<tr>
<th>Mode</th>
<th>LastWriteTime</th>
<th>Length</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>d-----</td>
<td>10/9/2018 8:41 PM</td>
<td></td>
<td>Files</td>
</tr>
<tr>
<td>d-----</td>
<td>10/9/2018 8:41 PM</td>
<td></td>
<td>Hives</td>
</tr>
<tr>
<td>-a----</td>
<td>10/16/2018 12:18 PM</td>
<td>429</td>
<td>layerchain.json</td>
</tr>
</tbody>
</table>

**Directory:** C:\ProgramData\docker\windowsfilter\04f084cfcd981103e0efc7da2b8c341913d63f36ebcaa537e9cae2899912bdad

<table>
<thead>
<tr>
<th>Mode</th>
<th>LastWriteTime</th>
<th>Length</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>d-----</td>
<td>4/12/2018 10:29 AM</td>
<td></td>
<td>Files</td>
</tr>
<tr>
<td>d-----</td>
<td>9/13/2018 4:58 PM</td>
<td></td>
<td>Hives</td>
</tr>
<tr>
<td>d-----</td>
<td>9/13/2018 4:58 PM</td>
<td></td>
<td>UtilityVM</td>
</tr>
<tr>
<td>-a----</td>
<td>9/13/2018 4:58 PM</td>
<td>16384</td>
<td>bcd.bak</td>
</tr>
<tr>
<td>-a----</td>
<td>9/13/2018 4:58 PM</td>
<td>16384</td>
<td>bcd.log.bak</td>
</tr>
<tr>
<td>-a----</td>
<td>9/13/2018 4:58 PM</td>
<td></td>
<td>bcd.log1.bak</td>
</tr>
<tr>
<td>-a----</td>
<td>9/13/2018 4:58 PM</td>
<td></td>
<td>bcd.log2.bak</td>
</tr>
<tr>
<td>-a----</td>
<td>9/13/2018 4:58 PM</td>
<td>41943040</td>
<td>blank-base.vhdx</td>
</tr>
<tr>
<td>-a----</td>
<td>9/13/2018 4:58 PM</td>
<td>4194304</td>
<td>blank.vhdx</td>
</tr>
<tr>
<td>-a----</td>
<td>9/13/2018 4:58 PM</td>
<td>4</td>
<td>layerchain.json</td>
</tr>
<tr>
<td>-a----</td>
<td>9/13/2018 4:58 PM</td>
<td>15</td>
<td>layout</td>
</tr>
</tbody>
</table>
How containers run

Architecture In Linux

- Docker Client
- Docker Compose
- Docker Registry
- Docker Swarm

REST Interface

Docker Engine

- libcontainerd
- libnetwork
- graph
- plugins

- containerd + runc

Operating System

- Control Groups
cgroups
- Namespaces
Pid, net, IPC, mnt, uts
- Layer Capabilities
Union Filesystems AUFS, btrfs, vmlinux, devicemapper
- Other OS Functionality

Architecture In Windows

- Docker Client
- Docker Compose
- Docker Registry
- Docker Swarm

REST Interface

Docker Engine

- libcontainerd
- libnetwork
- graph
- plugins

Host Compute Service

- Control Groups
Job objects
- Namespaces
Object Namespace, Process Table, Networking
- Layer Capabilities
Registry, Union Filesystem extensions
- Other OS Functionality

Operating System
Isolation Model

Process Isolation: shared kernel

Hyper-V isolation: isolated kernel
## Window Server versions

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Initial launch of containers</strong></td>
<td><strong>Optimized container images for Nano Server and Server Core</strong></td>
<td><strong>Optimized Server Core image</strong></td>
<td><strong>Optimized Server Core image</strong></td>
</tr>
<tr>
<td><strong>Process and Hyper-V isolation</strong></td>
<td><strong>Platform level support for Linux containers</strong></td>
<td><strong>App compat improvements</strong></td>
<td><strong>App compat improvements</strong></td>
</tr>
<tr>
<td><strong>Docker EE Basic Included at no additional cost</strong></td>
<td><strong>Windows Subsystem for Linux</strong></td>
<td><strong>Native command line tools – curl.exe, tar.exe and SSH</strong></td>
<td><strong>Enhanced Group Managed Service Account support</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Networking enhancements for overlays and SDN</strong></td>
<td><strong>Enhancements to the Windows Subsystem for Linux</strong></td>
<td><strong>Platform functionality for Kubernetes and Microsoft Service Fabric</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Networking enhancements for greater density and quicker endpoint creation</strong></td>
<td><strong>Performance and density improvements</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Improved network security with Calico Open source storage plugins for Kubernetes</strong></td>
<td><strong>Platform and open source work on CNI networking plugins such as Calico and Flannel</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Platform functionality required for Kubernetes conformance</strong></td>
<td><strong>Enhancements to the Windows Subsystem for Linux</strong></td>
</tr>
</tbody>
</table>
## Version matrix

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows Server 2016</td>
<td>Both Hyper-V and Process Isolation</td>
<td>Hyper-V Isolation only</td>
<td>Hyper-V Isolation only</td>
<td>Hyper-V Isolation only</td>
</tr>
<tr>
<td>Windows Server 1709</td>
<td>Not supported</td>
<td>Both Hyper-V and Process Isolation</td>
<td>Hyper-V Isolation only</td>
<td>Hyper-V Isolation only</td>
</tr>
<tr>
<td>Windows Server 1803</td>
<td>Not supported</td>
<td>Not supported</td>
<td>Both Hyper-V and Process Isolation</td>
<td>Hyper-V Isolation only</td>
</tr>
<tr>
<td>Windows Server 1809</td>
<td>Not supported</td>
<td>Not supported</td>
<td>Not supported</td>
<td>Both Hyper-V and Process Isolation</td>
</tr>
</tbody>
</table>
2. Kubernetes on Windows Server
Challenges of Containers

- Scheduling
- Service Discovery
- LoadBalancing
- Monitoring
- Failover
Why Kubernetes

The Production-Grade Container Orchestration

Portable
Public, private, hybrid, multi-cloud

Extensible
Modular, pluggable, hookable, composable

Self-healing
Auto-placement, auto-restart, auto-replication, auto-scaling
Kubernetes 101

Master node
- etcd
- Controller-manager
  - replication, namespace, serviceaccounts, etc.
- API Server
- scheduler

Worker node
- Kubelet
- Kube-proxy
- Docker
- Pod
  - Containers
- Internet

Containers
Kubernetes on Windows Server
Core Features

- Alpha since v1.5
- Beta since v1.9
- GA ETA v1.14

- Shared network namespace (compartment) with multiple Windows Server containers
- Load balancing via Virtual Filtering Platform (VFP) Hyper-v Switch Extension

- Container Runtime Interface (CRI) pod and node level statistics
- Support for kubeadm to add Windows Server nodes
- Both process and hyperv isolation
Clustering

Master nodes (Linux)
- **etcd**
- **API Server**
  - replication, namespace, service accounts, etc.
- **controller-manager**
- **scheduler**

Linux Nodes
- **Kubelet**
- **Kube-proxy**
- **Pod Containers**
- **Docker**

Windows Nodes
- **Kubelet**
- **Kube-proxy**
- **Pod Containers**
- **Docker**
Network Plugins

- Azure CNI
- ovn-kubernetes
- winncni
- flannel
- Calico (Policy Only)
The diagram illustrates a network setup with two nodes labeled Node1 and Node2. Each node has a Hyper-V Virtual Switch with connected Pods and Containers. The Pods are labeled 10.244.1.3 and 10.244.1.5 on Node1, and 10.244.2.10 and 10.244.2.2 on Node2. The Pods are connected to the Hyper-V Virtual Switch through NICs labeled 10.240.0.10 and 10.240.0.20.

A table below shows the prefixes and corresponding next hops:

<table>
<thead>
<tr>
<th>Prefix</th>
<th>Next Hop</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.244.1.0/24</td>
<td>10.240.0.10</td>
</tr>
<tr>
<td>10.244.2.0/24</td>
<td>10.240.0.20</td>
</tr>
</tbody>
</table>
Storage

**In-tree volume**
- AzureDisk, AzureFile
- HostPath, ...

**Flex volume**
- KeyVault, blobfuse, SMB

**CSI**
- Working In Process
Examples

```yaml
apiVersion: v1
classKind: Pod
metadata:
  - name: iis
  labels:
    name: iis
spec:
  containers:
    - name: iis
      image: microsoft/iis:windowsservercore-1709
      ports:
        - containerPort: 80
  nodeSelector:
    "beta.kubernetes.io/os": windows
```

```yaml
apiVersion: apps/v1
classKind: DaemonSet
metadata:
  name: my-daemonset
  labels:
    app: foo
spec:
  selector:
    matchLabels:
      app: foo
  template:
    metadata:
      labels:
        app: foo
    spec:
      containers:
        - name: foo
          image: microsoft/windowsservercore:1709
          nodeSelector:
            beta.kubernetes.io/os: windows
```
Known issues

• Multiple containers per pod is only supported on Windows Server 1709 or later
• Windows container OS must match Host OS
• Pod’s DNS is set via network plugins
• Only limited volumes are supported
• Weave Net and full Calico are not supported
• Local traffic policy is not yet supported
3. What’s Next?
GA in v1.14

• Windows Server 2019
• More flex volumes: SMB, iSCSI
• Alpha containerd integration
• Fully conformance tests and node e2e tests
Future work

• Linux containers on Windows
• Containerd as default CRI
• More volumes: flexvolume, CSI
• Multiple containers per pod for Hyper-V isolation
• Daemon Sets with host privileges
References

• https://docs.microsoft.com/en-us/virtualization/windowscontainers/kubernetes/getting-started-kubernetes-windows
• https://kubernetes.io/docs/getting-started-guides/windows/
• https://www.youtube.com/watch?v=j2B7cLdTXMw
• https://github.com/Microsoft/SDN/tree/master/Kubernetes
• https://trello.com/b/rjTqrwjL/windows-k8s-roadmap
Thanks & QA