



KubeCon



CloudNativeCon

Europe 2021

Virtual



Forward Together »



KubeCon



CloudNativeCon

Europe 2021

Kubernetes on Windows: A Journey

Jerry Lozano
RX-M, LLC.

Virtual



Welcome



KubeCon



CloudNativeCon

Europe 2021

Virtual

- Presenter: Jerry Lozano
- Software developer
- Primary focus: Systems development
- Experience with Unix, Linux, and Windows
 - Device driver development thru Application development with frameworks
- Senior Consultant with RX-M LLC
 - Cloud Native Consulting and Training



Kubernetes on Windows?



KubeCon



CloudNativeCon

Europe 2021

Virtual

- Kubernetes is designed to run *Linux* containers in an orchestrated environment
- Built on Linux concepts and kernel specific features
 - Example: cgroups and iptables
- CNCF is part of the Linux Foundation!
- Q:
 - **When** did Windows enter the world of K8s?
 - **Why** does Windows matter in the world K8s?

Does K8s Even Run On Windows?



KubeCon



CloudNativeCon

Europe 2021

Virtual

- As stated clearly in the official documentation for K8s:
 - “The Kubernetes control plane, including the master components, continues to run on Linux. There are no plans to have a Windows-only Kubernetes cluster”
- However, since K8s 1.14, Windows containers on *Windows nodes* within the cluster **are supported**
 - Today, mixed clusters require 1.17 or later
- Mixed clusters may be unsettling to Linux purists
 - But the idea is powerful and makes perfect sense

Why Windows?



KubeCon



CloudNativeCon

Europe 2021

Virtual

- According to W³Techs, Windows is used on more of the top 1000 web sites than Unix/Linux
- statista shows that over 70% of the global server market is Windows-based
- **The point:**

There is a ton of code on Windows that face the same challenges as any application moving forward:

- Scalability
- Availability
- Manageability, etc.

K8s Deployments Are Already Mixed



KubeCon



CloudNativeCon

Europe 2021

Virtual

- Containers running in K8s pods rely upon most suitable OS for the job (read: microservice) at hand
 - Containers are a *virtual OS* after all...
 - Ubuntu and Fedora distros have distinct differences, for example
- If K8s worker nodes running Linux supported Windows containers, this would be a different talk...
- Instead, we need Windows worker nodes to host the pods running Windows containers

Kubernetes + Windows = The Journey



KubeCon



CloudNativeCon

Europe 2021

Virtual

- As with all things, the actual path to hosting a mixed cluster (Linux + Windows) may not be straightforward
- Some problems are specific to the environment (e.g., cloud provider), but...
- Some problems are inherent to the nature of mixing Windows and Linux on the same cluster
- The remainder of this presentation is a chronical of the journey: **Kubernetes with Windows worker nodes**

Our Microcosm



KubeCon



CloudNativeCon

Europe 2021

Virtual

- For the purposes of this talk, we will use a representation of a reasonable implementation:
 - Mixed cluster running on AWS
- We set up:
 - Ubuntu server running the K8s control plane
 - One Ubuntu worker node
 - One Windows Server worker node
- The Linux worker node will host an nginx web page offering “widgets” for sale
- The Windows node will host a .NET C# microservice, authorizing user supplied credit cards

Example Mixed Cluster



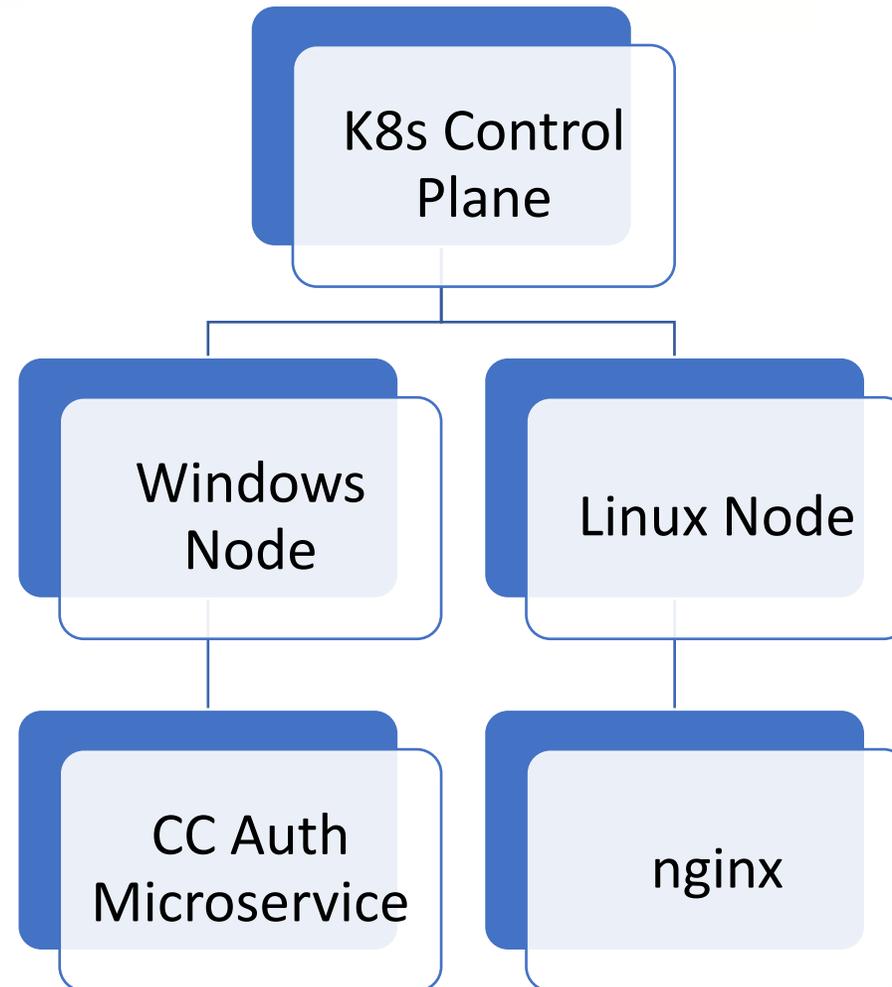
KubeCon



CloudNativeCon

Europe 2021

Virtual



Problem #1: Setting Up Windows Server



KubeCon

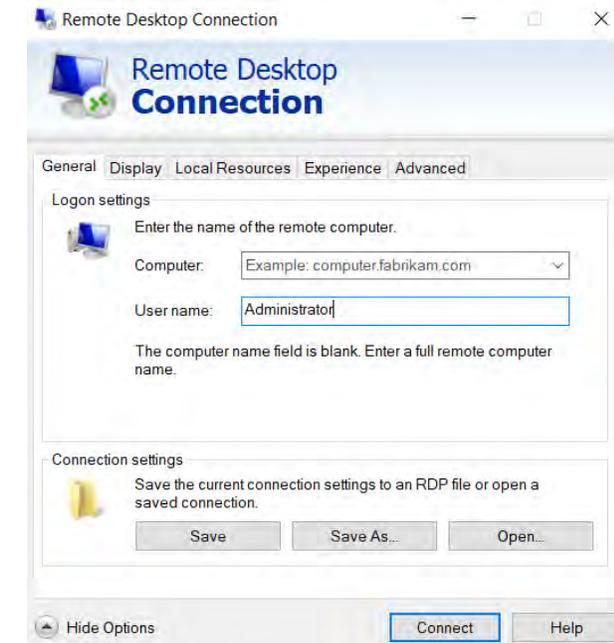


CloudNativeCon

Europe 2021

Virtual

- Obviously, every situation is provider and environment dependent, but...
- K8s requires Windows worker nodes run Windows Server 2019
- Most of us use ssh to connect to our Linux consoles
- Windows Server 2019 starts with a graphical desktop, so...
 - We need to use RDP, not ssh, to log on to each Windows Server, initially



A Windows Server 2019 Node



KubeCon



CloudNativeCon

Europe 2021

Virtual

About

Processor	Intel(R) Xeon(R) CPU E5-2676 v3 @ 2.40GHz 2.40 GHz
Installed RAM	1.00 GB
Device ID	9698954A-AE46-491C-8D6E-9D6A22715088
Product ID	00430-00000-00000-AA462
System type	64-bit operating system, x64-based processor
Pen and touch	Pen and touch support with 10 touch points

Rename this PC

Windows specifications

Edition	Windows Server 2019 Datacenter
Version	1809
Installed on	4/2/2021
OS build	17763.1817

[Change product key or upgrade your edition of Windows](#)

[Read the Microsoft Services Agreement that applies to our services](#)

[Read the Microsoft Software License Terms](#)

Windows specifications

Edition	Windows Server 2019 Datacenter
---------	--------------------------------

Problem #2: Adding a Windows Node to the Cluster



KubeCon



CloudNativeCon

Europe 2021

Virtual

- While some steps are the same as adding Linux as a worker node, Windows nodes are a little different:
 - Networking is “more restricted” for a Windows worker node
- Install Flannel on the control plane
 - VNI must be set to 4096 with Port = 4789 (Windows requirements)
 - Or an L2Bridge/Host-gateway mode can be used (host-gw), possibly a better choice
 - Documentation makes this clear:
<https://kubernetes.io/docs/tasks/administer-cluster/kubeadm/adding-windows-nodes/>

- kube-flannel.yaml:

```
126 net-conf.json: |
127     {
128         "Network": "10.244.0.0/16",
129         "Backend": {
130             "Type": "vxlan",
131             "VNI": 4096,
132             "Port": 4789
133         }
134     }
135 ---
```

Install Container and kubelet Support on Windows Node(s)



KubeCon



CloudNativeCon

Europe 2021

Virtual

- Install-WindowsFeature -Name containers
- Then K8s support for worker nodes:

```
PS C:\Users\Administrator> curl.exe -LO https://github.com/kubernetes-sigs/sig-windows-tools/releases/latest/download/PrepareNode.ps1
% Total    % Received % Xferd  Average Speed   Time    Time     Time    Current
           Dload  Upload   Total   Spent    Left     Speed
100 159    100 159    0    0    159    0  0:00:01 --:--:--  0:00:01  597
100 623    100 623    0    0    623    0  0:00:01 --:--:--  0:00:01  623
100 5971   100 5971    0    0   5971    0  0:00:01 --:--:--  0:00:01 5971
PS C:\Users\Administrator> .\PrepareNode.ps1 -KubernetesVersion v1.20.0
Using Kubernetes version: v1.20.0
```

Directory: C:\

Mode	LastWriteTime	Length	Name
d----	4/3/2021 5:13 PM		k
Downloading https://dl.k8s.io/v1.20.0/bin/windows/amd64/kubelet.exe to C:\k\kubelet.exe			
Downloading https://dl.k8s.io/v1.20.0/bin/windows/amd64/kubeadm.exe to C:\k\kubeadm.exe			
Downloading https://github.com/rancher/wins/releases/download/v0.0.4/wins.exe to C:\k\wins.exe			
Creating Docker host network			
656b3995c87c18c5c51d12b46112142d0591148c1d5daf20fe5a9182167e59d7			
Registering wins service			

Directory: C:\var\log

Mode	LastWriteTime	Length	Name
d----	4/3/2021 5:13 PM		kubelet

```
PS C:\Users\Administrator> kubeadm join 172.31.78.90:6443 --token lutns9.4zpv73ss0dzxgk1 --discover
9cfc2a9e80c302e325eaa733a8a2bc70da6da564ccde0715b49e3dd
[preflight] Running pre-flight checks
[WARNING SystemVerification]: this Docker version is not on the list of validated versions:
9.03
[preflight] Reading configuration from the cluster...
[preflight] FYI: You can look at this config file with 'kubectl -n kube-system get cm kubeadm-config
W0403 17:18:59.504808 3556 utils.go:69] The recommended value for "authentication.x509.clientCAFi
c\kubernetes\pki\ca.crt; the provided value is: /etc/kubernetes/pki/ca.crt
[kubelet-start] Writing kubelet configuration to file "\\var\lib\kubelet\config.yaml"
[kubelet-start] Writing kubelet environment file with flags to file "\\var\lib\kubelet\kubeadm-fl
[kubelet-start] Starting the kubelet
[kubelet-start] Waiting for the kubelet to perform the TLS Bootstrap...
```

This node has joined the cluster:

- * Certificate signing request was sent to apiserver and a response was received.
- * The Kubelet was informed of the new secure connection details.

Run 'kubectl get nodes' on the control-plane to see this node join the cluster.

Both Worker Nodes Up & Running!



KubeCon



CloudNativeCon

Europe 2021

Virtual

- We are ready to deploy

```
ubuntu@ip-172-31-78-90:~$ kubectl get nodes -o wide
NAME                STATUS    ROLES    AGE     VERSION   INTERNAL-IP   EXTERNAL-IP   OS-IMAGE
  KERNEL-VERSION   CONTAINER-RUNTIME
ec2amaz-7bcffrf    Ready    <none>   2m10s  v1.20.0   172.31.53.81  <none>        Windows Server 2019 Datacenter 10.0.17763.1817 docker://19.3.14
ip-172-31-39-65    Ready    <none>   16h    v1.20.5   172.31.39.65  <none>        Ubuntu 20.04.2 LTS 5.4.0-1038-aws  docker://20.10.5
```

- Our journey requires us to containerize the Windows web service that implements the microservice (CC Auth)
- We have options to deploy a Windows-based web services:
 - MSI file to install virtual directory under IIS
 - Or just copy the .svc (or .asmx) file and DLL to the wwwroot directory!

The Windows CCAuth Service



KubeCon



CloudNativeCon

Europe 2021

Virtual

- The interface for the service:

```
namespace CCAuth
{
    // NOTE: You can use the "Rename" command on the "Refactor" menu to change the interface name
    [ServiceContract]
    public interface ICCAuthService
    {
        // TODO: Add your service operations here
        [WebGet(ResponseFormat = WebMessageFormat.Json)]
        [OperationContract]
        string Authorize(UInt64 ccNum, decimal amount);
    }
}
```

- The implementation:

```
namespace CCAuth
{
    // NOTE: You can use the "Rename" command on the "Refactor" menu to change the
    public class CCAuthService : ICCAuthService
    {
        public string Authorize(UInt64 ccNum, decimal amount)
        {
            if (ccNum % 10 != 9 && amount < 1000.00M)
            {
                Random r = new Random();
                int authCode = r.Next(1000, 9999);
                return "Auth Code: " + authCode.ToString("D4");
            }
            return "Declined";
        }
    }
}
```

Problem #3: How To Configure Windows Container Image



KubeCon



CloudNativeCon

Europe 2021

Virtual

- Some steps are familiar:
 - Dockerfile starts with appropriate base image (e.g., Microsoft/iis)
- But base Windows Server images have many features “turned off”
- Examples:
 - ASP.NET Framework
 - Web service hosting (i.e., allowing .SVC files to execute)
 - (Makes it hard to implement a RESTful interface for our microservice!)
- From a Dockerfile, we need to install windows features *and* enable IIS features that would otherwise prevent needed services

```
PS C:\Users\Administrator\Documents\MakeImage> cat .\Dockerfile
FROM microsoft/iis
MAINTAINER Jerry Lozano <jerry.lozano@srx-m.com>
SHELL ["powershell", "-Command", "$ErrorActionPreference = 'Stop';"]
```

```
# Install ASP.NET
RUN Install-WindowsFeature NET-Framework-45-ASPNET; Install-WindowsFeature Web-Asp-Net45; Install-WindowsFeature Web-Mgmt-Compat
RUN Install-WindowsFeature Web-Http-Errors; Install-WindowsFeature Web-Static-Content; Install-WindowsFeature NET-WCF-HTTP-Activation45
```

Building Docker Image Using an MSI File



KubeCon



CloudNativeCon

Europe 2021

Virtual

- Not so familiar steps:
 - COPY msi file to accessible path
 - RUN msixexec
- On Windows, docker image build ... just works
 - (Base Windows images are large, so patience required...)
- Once images are built, normal distribution just works
 - docker container... and docker image... work on Windows

```
PS C:\Users\Administrator\Documents\MakeImage> docker image build -t ccauth .
Sending build context to Docker daemon 1.087MB
Step 1/9 : FROM microsoft/iis
--> a83e2bb65bf3
Step 2/9 : MAINTAINER Jerry Lozano <jerry.lozano@srx-m.com>
--> Using cache
--> 6e84b16b1d94
Step 3/9 : SHELL ["powershell", "-Command", "$ErrorActionPreference = 'Stop';"]
--> Using cache
```

```
Successfully built 8a2fcb9fb809
Successfully tagged ccauth:latest
```

Node Selection Concerns



KubeCon



CloudNativeCon

Europe 2021

Virtual

- The K8s Control Plane must launch Windows containers on Windows worker nodes
 - Select carefully!
- Good news: kubernetes.io/os already contains “Linux” or “Windows” for a node
- nodeSelector can therefore gain affinity from this value
- service yaml extract:

```
ubuntu@ip-172-31-78-90:~$ cat ccauth.yaml
apiVersion: v1
kind: Pod
metadata:
  name: ccauthsvc
  labels:
    purpose: ccauth-svc
spec:
  containers:
  - name: ccauth
    image: ccauth
    ports:
    - containerPort: 80
      hostPort: 80
  restartPolicy: Never
  nodeSelector:
    kubernetes.io/os: windows
```

Kubernetes Brings up the App



KubeCon



CloudNativeCon

Europe 2021

Virtual

- Once the container images are built and available to nodes, K8s brings up the pods

```
ubuntu@ip-172-31-78-90:~$ kubectl run ccauth --image=ccaauth
pod/ccaauth created
```

- And the pods are manageable in the usual way:

```
ubuntu@ip-172-31-78-90:~$ kubectl delete pod ccauth
pod "ccaauth" deleted
```

Important Sidenotes



KubeCon



CloudNativeCon

Europe 2021

Virtual

- Other than the concerns already discussed, working with a mixed cluster in Kubernetes is not *that different*
- But... There are some limitations that are well documented:
 - Host networking mode is not available for Windows pods
 - Accessing service VIPs from nodes will be available with a future release of Windows Server
 - A single service can only support up to 64 backend pods / unique destination IPs
 - Windows containers connected to I2bridge, I2tunnel, or overlay networks do not support communicating over the IPv6 stack.
 - Secrets are written in clear text on the node's volume
 - Many others:
 - <https://kubernetes.io/docs/setup/production-environment/windows/intro-windows-in-kubernetes/>

Summary



KubeCon



CloudNativeCon

Europe 2021

Virtual

- Mixing Windows with Linux on a Kubernetes cluster *is* a viable option – *now*
- There are at least two factors that justify mixed mode cluster:
 - Bringing forward existing Windows code –or–
 - Choosing to use Windows frameworks and technologies (e.g., ASP.NET)
- In many ways, Windows is just another container environment for K8s
 - Albeit requiring a special worker node to host the containers/pods

Thank You!

Certification Training:

CKAD / CKA / CKS – self-study & virtual/in-person instructor-led training by a certified instructor

RX-M Cloud Native Expertise:

Training – Consulting / Advisory – Staffing

- Large breadth of cloud native / open source / digital transformation training curriculum
- Custom consultative training programs aligned to job roles and business goals

Contact us at: rx-m.com / info@rx-m.com



KubeCon



CloudNativeCon

Europe 2021

Virtual



rx-m cloud native
training &
consulting

