

Virtualization in Containers

Marek Libra
Senior Software Engineer
Red Hat

Linux Day Milano 2019



About Marek Libra

- Passionate software developer since 2000
- KubeVirt
- oVirt
- <https://github.com/mareklibra>
- <https://www.linkedin.com/in/mareklibra>

Linux Container

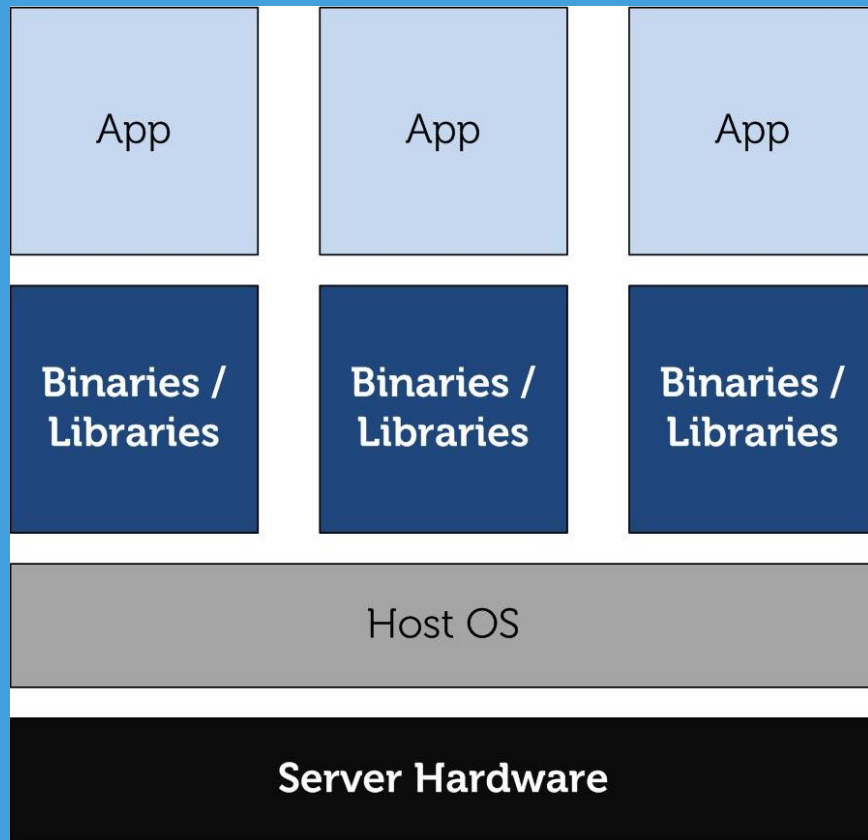
- **Isolated** native process with runtime environment
 - Shared kernel, lightweight
 - CPU, memory effective

Virtual Machine

- A process emulating runtime HW
 - Multiple distinct kernels on a single host
 - Independent process schedulers
 - Wider flexibility

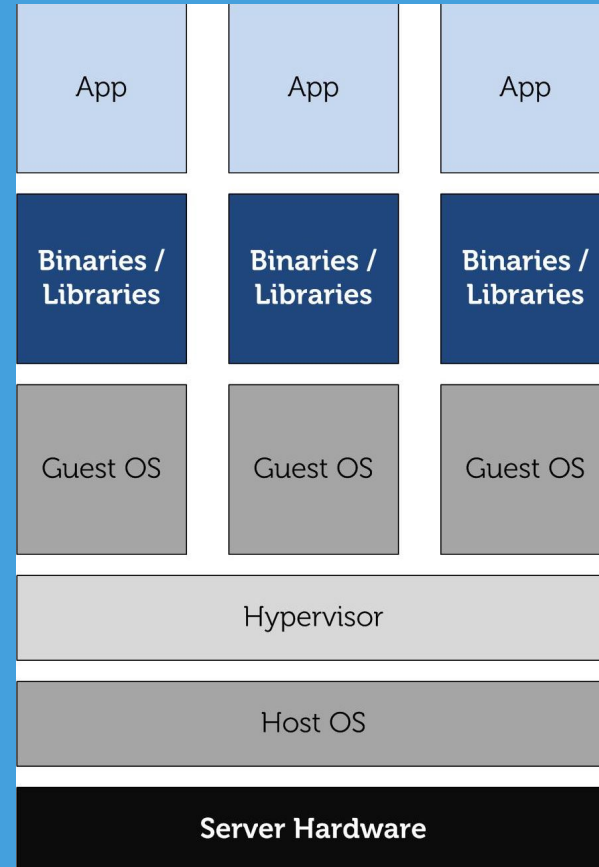
Linux Container - How

- Standard Linux Process
- Cgroups
 - control and limit resource usage
- Namespaces
 - IPC, Net, Mount, PID,
User, UTS, Cgroup
- Image
- CRI-O, Docker, rkt, containerd ?



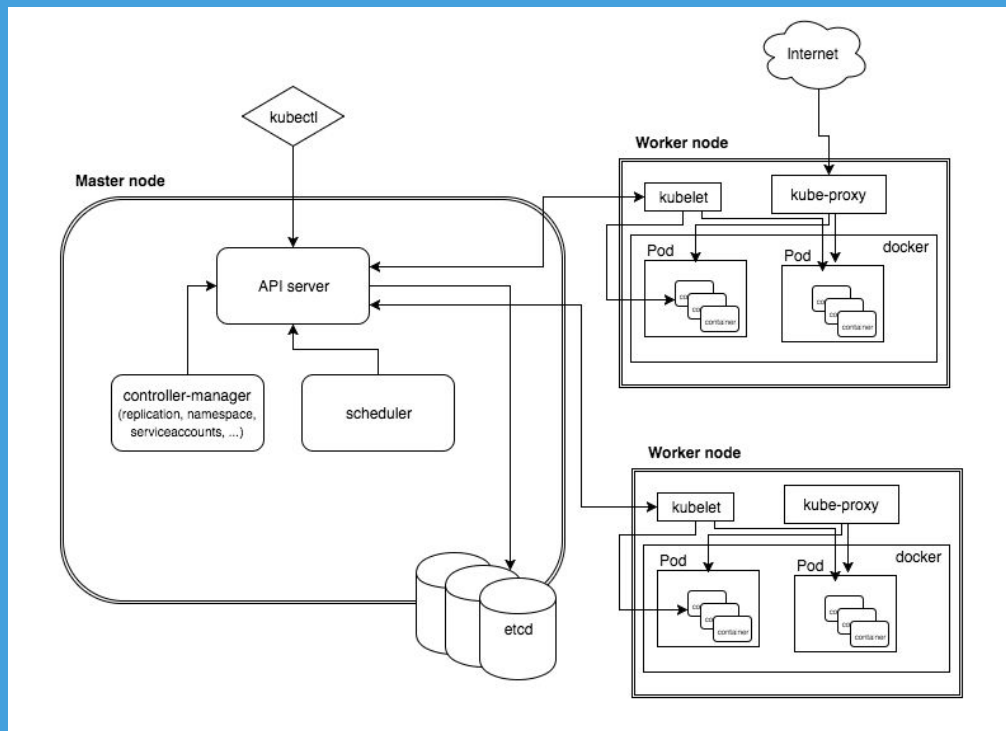
Virtual Machine

- Hypervisor emulates hardware
 - Multiple kernels
 - Memory and CPU ineffective
 - Emulation of devices
 - Guest-OS flexibility
- Multiple implementations
 - KVM/Qemu, Xen, VMware, ...
 - Libvirt



Kubernetes

- Industry standard, huge community
- Pod grouping containers
 - Shared Network namespace
 - Distinct PID, mount, IPC
- Declarative REST API/etcd
- Controllers
 - Scheduler, kubelet, ...
- Storage, Networking



Kubernetes Command Line

- Declarative instead of Imperative
- `kubectl [command] [TYPE] [NAME] [flags]`
- `kubectl get pods`
- `kubectl get pod my-pod-name -o yaml`
- Create, describe, delete, patch, ...

```
$ kubectl get pod myapp-pod -o yaml
```

```
apiVersion: v1
kind: Pod
metadata:
  name: myapp-pod
  labels:
    app: myapp
spec:
  containers:
  - name: myapp-container
    image: busybox:latest
    command: ['sh', '-c', 'echo Hello
              Kubernetes! && sleep 3600']
```

OKD / OpenShift

- PaaS based on Kubernetes
 - Security - tighter SCC, default RBAC, OAuth with infra apps
 - ImageStreams
 - Cmd-line plus Web User Interface
 - Enterprise ready open source - support, installation
 - Web user interface

KubeVirt



- Addon to Kubernetes
- Run virtual machine workloads as / along containers
- Legacy applications in new era of clouds and container native infra
 - Mixed workloads - single pipeline for both
 - Gradual decomposition of VMs into containers
 - Shared infrastructure to decrease operating costs
 - Freedom of operating system choice
 - Increased isolation



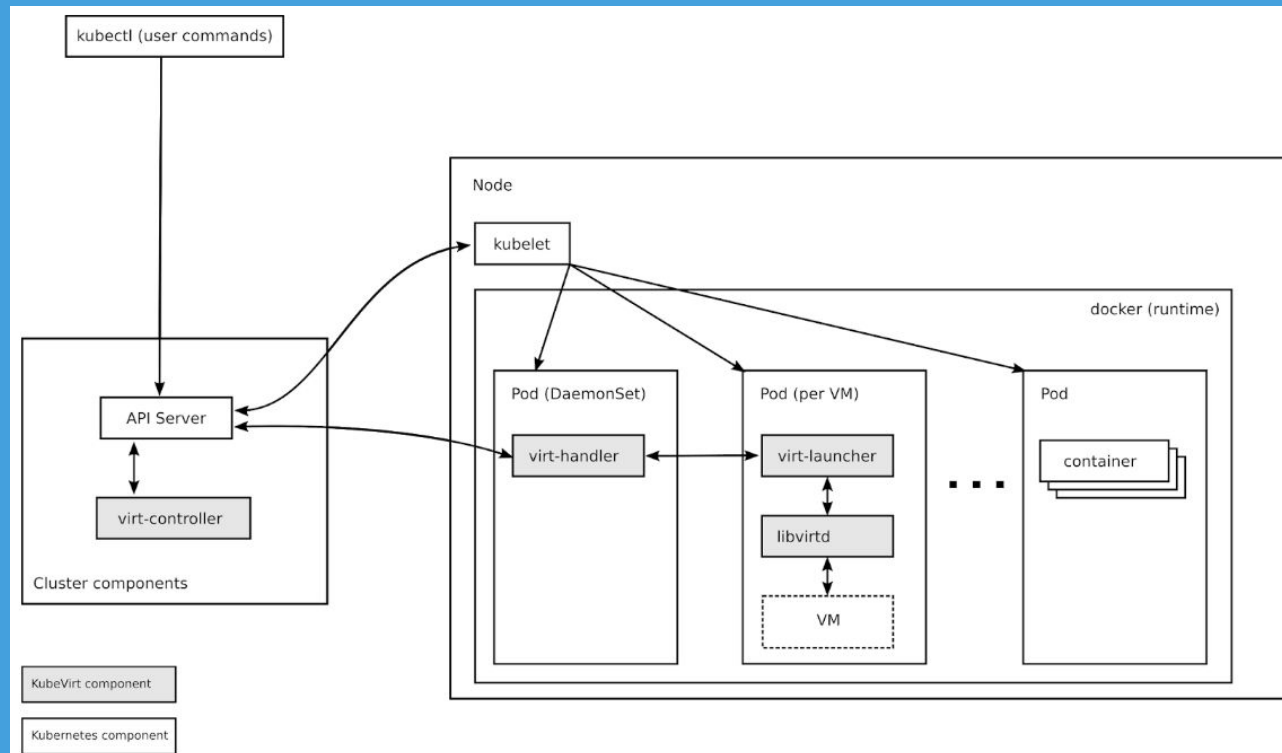
KubeVirt - How

Custom Resource Definition

Virt-controller

Virt-handler

libvirt/qemu in Pod



KubeVirt VirtualMachine

- Declarative
- Defaults
- virt-controller/virt-handler

```
apiVersion: kubevirt.io/v1alpha3
kind: VirtualMachine
metadata:
  labels:
    kubevirt.io/vm: vm-fedora
  name: vm-fedora
spec:
  running: false
  template:
    metadata:
      labels:
        kubevirt.io/os: fedora28
        kubevirt.io/vm: vm-fedora
    spec:
      domain:
        devices:
          disks:
            - disk:
                bus: virtio
                name: containerdisk
                volumeName: registryvolume
            - disk:
                bus: virtio
                name: cloudinitdisk
                volumeName: cloudinitvolume
          cpu:
            cores: 2
          resources:
            requests:
              memory: 256M
        terminationGracePeriodSeconds: 0
      volumes:
        - name: registryvolume
          containerDisk:
            image: 'kubevirt/fedora-cloud-registry-disk-demo:latest'
        - cloudInitNoCloud:
            userData: |-
              #cloud-config
              password: fedora
              chpasswd: { expire: False }
            name: cloudinitvolume
```

KubeVirt VirtualMachine



- Declarative over Imperative
 - Like any other k8s resource

- Fully integrated

```
kubectl [command] [type] [name] [flags]
```

- With shortcuts

```
virtctl [command] [name]
```

```
$ kubectl apply -f my-vm.yaml  
$ kubectl get vm my-vm
```

```
$ virtctl console my-vm
```

The screenshot shows the OKD Web User Interface (WUI) for a virtual machine named 'vm-fedora'. The interface is divided into a sidebar on the left and a main content area. The sidebar contains navigation links for Administrator, Home, Operators, Workloads, Pods, Virtual Machines, Virtual Machine Templates, Deployments, Deployment Configs, Stateful Sets, Secrets, Config Maps, Cron Jobs, Jobs, Daemon Sets, Replica Sets, Replication Controllers, Horizontal Pod Autoscalers, Networking, Storage, Builds, Monitoring, Compute, and Administration. The main content area shows the 'VM Overview' for 'vm-fedora', including its Name, Namespace (default), Labels, Annotations, Description, Operating System (Fedora 28), Template (fedora-highperformance-large-v0.6.2), Created At (9 minutes ago), and Owner (No owner).

Web User Interface

- Fully integrated with the OKD console
- Access
- Provision
- Management
- Import

The screenshot shows the OKD Web User Interface (WUI) for a list of virtual machines. The interface is divided into a sidebar on the left and a main content area. The sidebar contains navigation links for Administrator, Home, Operators, Workloads, Networking, Storage, Builds, Monitoring, Compute, and Administration. The main content area shows the 'Virtual Machines' list, which includes a table with columns for Name, Namespace, and Status. The table lists several virtual machines, including 'windows-02', 'windows-01', 'vm-cirros', 'test-dv-test-tkbud', 'test-dv-test-ezvc', 'container-test-pptpm', and 'container-test-mcws'. A context menu is open for the 'vm-cirros' virtual machine, showing options: Stop Virtual Machine, Restart Virtual Machine, Migrate Virtual Machine, Clone Virtual Machine, Edit Labels, Edit Annotations, and Delete Virtual Machine.

Web User Interface

Create VM wizard



Create Virtual Machine

Basic Settings 1 Networking 2 Storage 3 Result 4

* Name: vm-fedora4

Description: My Fedora

* Namespace: default

Template: --- Select Template ---

* Provision Source: Container

* Container Image: PXE, URL, Container

* Operating System: --- Select Flavor ---

* Workload Profile: --- Select Workload Profile ---

☐ Start virtual machine on creation

☐ Use cloud-init

Cancel < Back Next >

Create Virtual Machine

Basic Settings 1 Networking 2 Storage 3 Result 4

Create Disk Attach Disk

DISK NAME	SIZE (GB)	STORAGE CLASS
rootdisk		
disk0	1	my-storage-class

* Bootable Disk: rootdisk

☒ ☐

Cancel < Back Create Virtual Machine >

The screenshot shows the OKD web console interface. The top navigation bar includes the OKD logo, a hamburger menu, and the user 'kube:admin'. The left sidebar contains a navigation menu with items like Administrator, Home, Operators, Workloads, Networking, Storage, Builds, Monitoring, Compute, and Administration. The main content area shows the 'Virtual Machine Details' for 'vm-cirros'. Below this, there are tabs for Dashboard, Overview, YAML, Consoles, Events, Network Interfaces, and Disks. The 'Consoles' tab is active, displaying a 'VNC Console' window. The console output shows system boot logs for a 'vm-cirros' machine, including kernel messages and a login prompt for the 'cirros' user.

Project: default

Virtual Machines > Virtual Machine Details

VM vm-cirros Actions

Dashboard Overview YAML Consoles Events Network Interfaces Disks

VNC Console ☐ Disconnect before switching Send Key

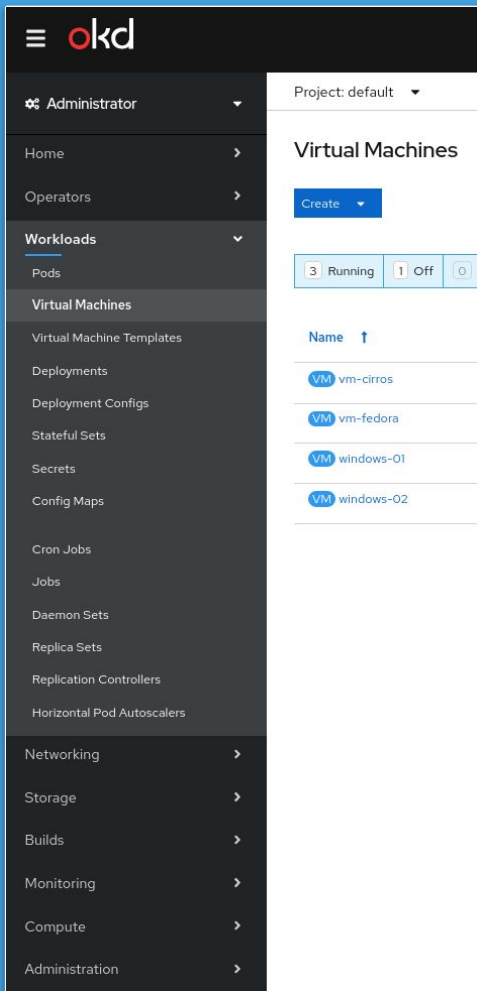
```
[ 4.576225] Key type trusted registered
[ 4.586751] Key type encrypted registered
[ 4.596881] AppArmor: AppArmor sha1 policy hashing enabled
[ 4.609736] ima: No TPM chip found, activating TPM-bypass!
[ 4.622799] evm: HMAC attrs: 0x1
[ 4.641046]   Magic number: 7:747:330
[ 4.651040] tty ttyS1: hash matches
[ 4.662323] rtc_cmos 00:00: setting system clock to 2019-10-01 13:18:50 UTC (
1569935930)
[ 4.681479] BIOS EDD facility v0.16 2004-Jun-25, 0 devices found
[ 4.695771] EDD information not available.
[ 4.709287] Freeing unused kernel memory: 1480K (ffff80001f420000 - ffffff8
20b4000)
[ 4.727239] Write protecting the kernel read-only data: 14336k
[ 4.740679] Freeing unused kernel memory: 1860K (ffff8000182f0000 - ffff88000
1a00000)
[ 4.758781] Freeing unused kernel memory: 168K (ffff80001dd60000 - ffff880001
e00000)

further output written to /dev/ttyS0
[ 6.965069] random: dd urandom read with 18 bits of entropy available

login as 'cirros' user. default password: 'gocubsgo'. use 'sudo' for root.
vm-cirros login:
```

VM Consoles

- VNC
- Serial
- RDP
- In-browser vs. desktop



Related Objects

- Virtualization is first-class citizen along Containers
- Storage
- Networks
- Services
- Monitoring and Management
- Analyze issues - Status info, link through to details

