VIRTUALIZATION NOW AND IN THE FUTURE

Enterprise virtualization to container-native virtualization

Andrew Sullivan
Principal Technical Marketing Manager

Doron Fediucl
Director, Software Engineering

May 2019
A (BRIEF) HISTORY OF VIRTUALIZATION
FROM THEN, TO NOW

RED HAT ENTERPRISE VIRTUALIZATION BEATS VMWARE on the SPECvirt_sc2010 benchmark on both speed and scale.

2010

2009 QUMRANET ACQUISITION

2012 RED HAT ENTERPRISE VIRTUALIZATION 3.0
More solution partners
RESTful API
Memory overcommit

2013

2012 RED HAT ENTERPRISE VIRTUALIZATION 3.0
Windows guests
NUMA collaboration with HP

2015

2014 RED HAT ENTERPRISE VIRTUALIZATION 3.3, 3.4
OpenStack Neutron integration
Hot Plug CPU Affinity management
IBM Power support

2016

2014 RED HAT ENTERPRISE VIRTUALIZATION 3.6
V2V migration tool

2017

2015 RED HAT ENTERPRISE VIRTUALIZATION 3.1, 3.2

2017 RED HAT ENTERPRISE VIRTUALIZATION 3.6
Ansiible integration
Native SDN

2018

2016 RED HAT ENTERPRISE VIRTUALIZATION 4.0
10th product release

2018 RED HAT ENTERPRISE VIRTUALIZATION 4.3
RHEL 8 Guests
OSP 13/14 SDN
Expanded Ansible roles

2019

2018 RED HAT ENTERPRISE VIRTUALIZATION 4.2
Native DR
New metrics store
New UI
Cisco ACI

#redhat #rhsummit
KERNEL-BASED VIRTUAL MACHINE (KVM)

- KVM is a part of the Red Hat Enterprise Linux kernel
- QEMU uses KVM to execute virtual machines
- libvirt provides a management abstraction layer that homogenizes capabilities and simplifies the creation, consumption, and management of KVM-based virtual machines
- Red Hat Virtualization, Red Hat OpenStack Platform, and Container-native virtualization all leverage KVM, QEMU, and libvirt
RED HAT VIRTUALIZATION
RED HAT VIRTUALIZATION OVERVIEW

Directory service
- Active directory
- IPA
- Red Hat Directory Server
- IBM Tivoli Directory Server

RED HAT ENTERPRISE LINUX

RED HAT VIRTUALIZATION MANAGER

RED HAT VIRTUALIZATION HYPERVISOR

Web browser
- Administrator portal
- User portal

REST API/
Python SDK/Java SDK

Web service

Console access

Storage domain

POSTGRESQL

SPICE or VNC

x86_64

PPC

Internal web service
HTTPS
SSH / SSL
RED HAT VIRTUALIZATION MANAGER

- Primary management interface for RHV
  - Ability to create, manage, and control configuration of physical (hosts, storage), logical (datacenter, cluster, etc.), and virtual machine resources
  - Administrator interface for managing RHV resources
  - Virtual machine portal for non-administrators
- REST API for automation and integration
  - Multiple SDKs available (Python, Java, Ruby)
ADMINISTRATOR DASHBOARD
HYPERVERSORS

- 2 different hypervisor “models”
  - Appliance: Red Hat Virtualization - Host (RHV-H)
  - Traditional OS: Red Hat Enterprise Linux (RHEL) w/RHV packages
- Both result in the same capabilities!
  - RHV-H has a smaller footprint, having only what’s needed to be a hypervisor
- Configuration and management are both handled the same by RHV-M
  - Updates/upgrades, power management, etc. all equivalent
  - Logical entities (e.g., networks and storage) are created and managed the same
- Do you want/need to customize the hypervisor OS layout and/or package set extensively?
  - Yes - RHEL
  - No - RHV-H
HYPERVERSOR ARCHITECTURE
A datacenter has 1 or more clusters.

Clusters are composed of 1 or more hosts.

VMs are hosted by the clusters and can be migrated to any host in the cluster.

All hosts in the cluster must access the same physical networks.

All hosts in the datacenter must have access to the same shared storage.

**Datacenter**

- **Cluster 1**
  - VMs
  - Logical networks
  - Hosts
  - Physical network

- **Cluster 2**
  - VMs
  - Logical networks
  - Hosts
  - Physical network

**Shared storage resources**
DEMO

ADDITIONAL INFORMATION

DOCUMENTATION
- RHV landing page - https://red.ht/2FT3MYO
- RHV documentation - https://red.ht/2uHnf7Z
- RHEL virtualization docs - https://red.ht/2uF4Ulu

OTHER
- RHEL blog - https://red.ht/2JVTcdk
- RHV Partner Connect Zone - https://red.ht/2WGbtqf
USE CASES

- Common Red Hat Virtualization use cases
  - Performance sensitive workloads
  - Dev and test for traditional applications
  - Hybrid and multi-hypervisor, e.g. integration with OpenStack virtual machines
  - Technical workstations, provide vGPU capabilities to virtual machines
  - Consolidate server workloads, e.g. host control plane for OpenStack Platform and OpenShift

- Tier 1 and “mode 1” virtualization
  - Very large VMs with high demanding and sensitive workloads
  - The application can’t or won’t provide native high availability, so it must rely on the infrastructure

- Available in a hyper-converged package with Red Hat Hyper-converged Infrastructure for Virtualization (RHV + Gluster)
RED HAT VIRTUALIZATION 4.3

- Major themes
  - Happy customers - User experience, closed Bugzillas, and completed RFEs
  - Maturity - Red Hat Virtualization is a stable, reliable, and trusted platform for enterprise virtualization
  - Infrastructure migration solution(s) - ready to host workloads migrated to Red Hat’s portfolio
- 74 requests for enhancement (RFEs) completed
- More than 240 customer bugs fixed
- 1,367 Bugzilla bugs fixed
- Several integrations in tech preview
RED HAT VIRTUALIZATION 4.3

- Scale
  - 384 vCPUs, 6TB RAM per VM
  - 4000 VMs, 400 hosts
  - 50 storage domains
- Red Hat Enterprise Linux 8 guest support
- Ansible 2.7 and expanded roles
  - ovirt.infra, ovirt.hosted-engine-setup, ovirt.engine-setup, ovirt.shutdown-env
- Red Hat OpenStack Platform 10, 13, and 14 SDN integration
- IBM POWER9 CPU architecture
- RHV-M - Scale out metrics store
- RHV-H - PVLAN, OpenSCAP, IPv6 host networking
- VMs - Live migration with pinning, Windows Server Failover Cluster
IPv6 in RHV 4.3
WHAT’S NEXT?

- **Network**
  - Cluster support for Open vSwitch

- **Storage**
  - Storage offload and integration via CinderLib
  - 4k native disk support for local datastores
  - Shared and local domains in the same datacenter

- **Backup**
  - Changed block tracking for virtual machines

- **RHV-M**
  - HTML5 console (no VNC)
  - Nested virtualization
  - Import Debian and Ubuntu VMs from VMware and Xen
Managed Block Storage
CONTAINERIZING VIRTUAL MACHINES
NEW APPLICATIONS START WITH CONTAINERS
BUT STILL RELY ON VIRTUAL MACHINES

EXISTING AUTOMATION TOOLSETS

SCM (GIT)

CI/CD

DEVELOPER

OPERATIONS

EXISTING AUTOMATION TOOLSETS

POD

POD

POD

POD

NODE

RHET

PERSISTENT STORAGE

REGISTRY

PHYSICAL

VIRTUAL

PRIVATE

PUBLIC

HYBRID

VIRTUAL/CLOUD MANAGEMENT
CONTAINERS != VIRTUAL MACHINES

VIRTUAL MACHINES

VM isolates the hardware

CONTAINERS

Container isolates the process

VM

App

App

App

App

OS Dependencies

Kernel

Hypervisor

Hardware

CONTAINER Host (Kernel)

Container

App

OS deps

Container

App

OS deps

Container

App

OS deps

Container

App

OS deps
WHAT IF WE COULD COMBINE THEM?
VIRTUAL MACHINES IN CONTAINERS

- A virtual machine is a process
- Containers encapsulate processes
- Both have the same underlying resource needs:
  - Compute
  - Network
  - (sometimes) Storage
- Resources are allocated and assigned to containers just like VMs
KUBEVIRT

Provides a way to transition application components which can’t be directly containerized on to a Kubernetes system

- Integrates directly into existing Kubernetes clusters
- Uses a k8s-native approach whenever possible
- Leverage Container Networking Interface (CNI), Container Storage Interface (CSI), and other k8s-native integrations

Schedule, connect, and consume VM resources just like container-native
CONTAINER-NATIVE VIRTUALIZATION

- Add virtual machines to OpenShift projects from the service catalog!
- Utilize new and existing virtual machine-based applications alongside container-based components
CONTAINERIZED VIRTUAL MACHINES

- Inherit many features and functions from Kubernetes
  - Scheduling, High availability
  - Attach/detach resources
- Containerized virtual machines have the same limitations as non-containerized
  - CPU, RAM, etc. limitations dictated by libvirt and QEMU
  - Linux and Windows guest operating systems
- Storage
  - Use PVCs to request persistent storage for VMs
  - Containerized Data Importer (CDI) simplifies getting VM images into PVCs
- Network
  - Inherit pod network by default
  - Multus enables direct connection to external network
DEMO
USE CASES

- Application components which are not yet containerized
- Consume virtualized applications directly from containerized applications
  - No ingress/egress can connect via pod network
- Decomposition
  - Start with a virtual machine hosted and managed by OpenShift
  - Decompose the VM services as time allows
CONTAINER-NATIVE VIRTUALIZATION

Uses trusted RHEL & RHV (KVM) virtualization technology

Technology Preview available now with OpenShift 3.11!
THE FUTURE OF VIRTUALIZATION
TODAY AND TOMORROW

● Red Hat Virtualization
  ○ Today - enterprise virtualization workloads for “mode 1” applications or any application which needs resiliency from the infrastructure
  ○ Tomorrow - continued reliability, scalability, and incremental improvements in capabilities to meet the needs of enterprise virtual machines
  ○ RHV is the mature, trusted platform for tier 1 application virtualization!

● Container-native Virtualization
  ○ Today - Tech preview (based on KubeVirt), cloud native applications which rely on virtualization components and/or which are transitioning from virtual machines to containers where a single platform is desired
  ○ Tomorrow - Evolution and improvement of features and capabilities to decrease the function gap between traditional and container-native virtualization
  ○ Tech preview, as a part of OpenShift 3.11, enabling you to test and experiment with container semantics for virtual machines
RHV VS CNV?

● Each targets different use cases
  ○ RHV = traditional virtualization for “mode 1” applications, providing robust, resilient infrastructure for traditional applications
  ○ CNV = VMs deployed and managed as containers, simplifying the transition of existing applications from conventional virtualization to containers
● Different maturity stages
● Different availability timelines
● It’s all KVM!
● Both share the same vision

RHV and CNV don’t compete, they complement each other!
WHAT ABOUT OPENSTACK?

- NETWORK FUNCTION VIRTUALIZATION
- WEB APPS/DEVELOPER CLOUD
- EDGE COMPUTING
- AI/ML/HPC
- SERVICE PROVIDER
# OpenShift 4 Provider Roadmap

**Installer Provisioned Infrastructure (IPI)**

<table>
<thead>
<tr>
<th>Version</th>
<th>Installer Provisioned Infrastructure (IPI)</th>
<th>User Provisioned Infrastructure (UPI)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Developer Preview</td>
<td><img src="https://via.placeholder.com/150" alt="Amazon Web Services" /></td>
<td><img src="https://via.placeholder.com/150" alt="Amazon Web Services" /></td>
</tr>
<tr>
<td>4.1</td>
<td><img src="https://via.placeholder.com/150" alt="Amazon Web Services" /></td>
<td><img src="https://via.placeholder.com/150" alt="Amazon Web Services" /></td>
</tr>
<tr>
<td>4.2</td>
<td><img src="https://via.placeholder.com/150" alt="Microsoft Azure" /></td>
<td><img src="https://via.placeholder.com/150" alt="Azure OpenStack Platform" /></td>
</tr>
<tr>
<td>4.3 (tentative)</td>
<td><img src="https://via.placeholder.com/150" alt="Alibaba Cloud" /></td>
<td><img src="https://via.placeholder.com/150" alt="IBM Cloud" /></td>
</tr>
</tbody>
</table>

**Notes:**
- **On qualified hardware stack**
PROGRESSING FORWARD

● Virtualization isn’t going anywhere
  ○ There will always be “mode 1” applications - mainframes are still a multi-billion dollar business!
● Containers are growing
● Applications are changing - both technologies are important
● Using KVM for your hypervisor makes you ready for the future
  ○ RHV today
  ○ CNV tomorrow
  ○ Get there using IMS!
THANK YOU

linkedin.com/company/Red-Hat
youtube.com/user/RedHatVideos
facebook.com/RedHatInc
twitter.com/RedHat