Divide et Impera
How a fintech company rearchitected their virtualization infrastructure while preserving workloads uptime

Andrea Perotti
Senior Technical Account Manager
Red Hat

Antonio Romito
Senior Technical Account Manager
Red Hat

September 2021
Who we are:

- Andrea Perotti - Senior Technical Account Manager
  
  contact:
  
  name: Andrea Perotti
  email: aperotti@redhat.com
  job: Senior Technical Account Manager
  location:
  city: Milan
  country: Italy
  profiles:
  - network: LinkedIn
    url: https://www.linkedin.com/in/perottiandrea/

- Antonio Romito - Senior Technical Account Manager
  
  contact:
  
  name: Antonio Romito
  email: aromito@redhat.com
  job: Senior Technical Account Manager
  location:
  city: Bergamo
  country: Italy
  profiles:
  - network: LinkedIn
    url: https://www.linkedin.com/in/antonioromito/
Agenda

- Customer scenario
- The idea
- Details of the implementation
Who is the customer?

- In the business of managed services
- Full outsourcer of infrastructures for their customers in the finance field, mainly banks
- Long time Red Hat Virtualization users, rely on the product to deliver and manage many core services like Home Banking applications and so on.
- Multiple Data Centers, spreaded across Europe
Customer Scenario (1/2)

- Customer has various RHV 4.3 environment in stretched architecture with single manager handling two different DCs.

- RHV Manager offer them central view for all their workloads.

- Each DC doesn’t share any resource with the other, they are distinct RHV/oVirt DC.
Initial Scenario

Datacenter DC1

Cluster 1

VMs

Hosts

Physical network

DC1 storage resources

Datacenter DC2

Cluster 2

VMs

Hosts

Physical network

DC2 storage resources
Customer Scenario (2/2)

- But there are some limits:
  - DR procedure requires the redeploy of the Manager, from once DC to the other
  - Costly scenario: buying a node to keep it shut down is not viable
  - VMs cannot be shut down to make room for Manager redeploy: they’re production workloads

- Customer was convinced that the best way to improve the architecture was to have a dedicated Manager in each region managing local DCs
Target Scenario

**Datacenter DC1**

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

- DC1 storage resources

**Datacenter DC2**

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

- DC2 storage resources
The Solution

How to split a RHV environment in two?

- Shutdown workloads in DC2
- Remove DC2 and all its elements from the existing RHV Manager
- Deploy a new manager in DC2
- Attach Storage Domains and import VMs into the DC2

Easy isn’t it?  … not doable without affecting existing workloads!
How to split without affecting the workloads?

Facts

- Workloads continue to exist and run even when the manager is off
- All informations are stored in the DB
- Need a way to remove information about DC2 from the DB without triggering the any effective action on the VMs
How to split without affecting the workloads?

The IDEA

- Isolate Manager running in DC1 from DC2
- Trick it into believing DC2 is gone
- Remove any object related to DC2
- Install a new Manager in DC2 from a backup taken before the split
- Isolate Manager in DC2 from DC1
- Trick it into believing DC1 is gone
- Remove any object related to DC1
- PROFIT!
THE SPLIT
A self-hosted engine (SHE) is a virtualized environment in which the Manager (MNGR) runs on a virtual machine on some of the hosts managed by that Manager.

Self-hosted engine cluster, is totally independent from RHV/oVirt Manager. Its role is just keep running one VM: SHE.

Thanks to SHE Cluster, MNGR VM is highly available, and if the host fails unexpectedly, the virtual machine is re-launched automatically to another SHE cluster host.
Prepare the field

- Identify a hypervisor (HOST-X) in DC2: will be used to install the new Manager
- Evacuate and remove HOST-X from the cluster in DC2
- Set the load balancing policy of the Clusters in DC1 and DC2 to “cluster_maintenance”
- Take a backup of the RHV Manager
Status of the infrastructure in the backup
Removal of DC2 from the existing manager

Smashing RHV/oVirt for fun and profit

- Remove from each host in DC2 the Power management configuration
- Stop communication from/to RHV Manager for hosts in DC2
  - State of hosts in DC2 state will transition from **Connecting** to **NonResponsive**
- For all Hosts in DC2 in NonResponsive state, select “Confirm ‘Host has been Rebooted”“
- Now will be possible to put all the “NonResponsive” HOSTS in "Maintenance" mode and remove them
- Remove templates, Clusters and finally DC2 itself
Removal of DC2 from the existing manager

Datacenter DC1
- Cluster 1
  - VMs
  - Hosts
  - Physical network
  - DC1 storage resources

Datacenter DC2
- Cluster 2
  - VMs
  - Hosts
  - Physical network
  - DC2 storage resources

We're fine ;-) Don’t need a Manager to continue to run
Installation of the second manager on DC2

- Put the hosted-engine cluster in **global maintenance** and shutdown RHV Manager
- Restore network connectivity between DC1 and DC2
- Deploy the new hosted-engine that will manage the DC2 on HOST-X
  - First restore Manager with its original FQDN: Beware to specify DC2 and related cluster as context for the new manager (In 4.4 that data is obtained from backup)
  - Then change its IP and rename it via `ovirt-engine-rename`
Installation of the second manager on DC2
Removal of DC1 from the second manager

Do it once, do it twice...

- Remove from each host in DC1 the Power management configuration
- Stop communication from/to the new Manager for hosts in DC1
  - State of hosts in DC1 will transition from **Connecting** to **NonResponsive**
- For all Hosts in DC2 in NonResponsive state, select “Confirm ‘Host has been Rebooted”“
- Now will be possible to put all the “NonResponsive” HOSTS in "Maintenance" mode and remove them
- Remove templates, Clusters and finally DC1 itself
Removal of DC1 from the second manager
Almost there!

- Restore communication from/to RHV Manager for hosts in DC1
- Exit the hosted-engine cluster in DC1 from **global maintenance**: Manager in DC1 will start
- Rollback the load balancing policy of the Clusters in DC1 from “cluster_maintenance”
- Rollback the load balancing policy of the Clusters in DC2 from “cluster_maintenance”
- Congratulations, you’ve splitted your RHV/oVirt infra with **no workload downtime**!
Destination: reached!
What we’ve learned

- Customers can be source of headaches
  but also of interesting challenges

- With such strict constraints there was no guarantee that we
  would have succeeded

- Test, test, test

- Procedure has been tested with RHV/oVirt 4.3 and 4.4 as well
All this has been possible also thanks to...

- Stephane Vigan - Professional Services, Consultant, Red Hat
- Andrea Mattioli - Professional Services, Consultant, Red Hat
- Sandro Bonazzola - Manager, Software Engineering, Red Hat
- Simone Tiraboschi - Principal Software Engineer, Red Hat
- Yedidyah Bar David - Principal Software Engineer, Red Hat
- Gordon Watson - Senior Software Maintenance Engineer, Red Hat
- The Customer
QUESTIONS?