VM: Hey VM, can I share a host with you?

Affinity rules in a virtual cluster

4th of Feb 2017

Martin Sivák
Senior Software Engineer
Red Hat Czech
About me and oVirt

Martin Sivák
Working for Red Hat since 2007
oVirt SLA team

- oVirt | Red Hat Virtualization (since 2008)
- Virtual datacenter management solution
- Our virtual machines are pets, not cattle.
Agenda

- What is affinity
- Why affinity - use cases
- Overview of affinity types
- Affinity conflicts
- Affinity management in oVirt
- Future ideas
What is affinity

- Attraction factor between virtual machines
- Attraction factor between virtual machines and hosts
- **Automatically enforced** logical rules for virtual machine placement
  - Less strict than pinning
  - Can *dynamically adapt* to the current cluster situation and load
Why affinity?

- Licensing requirements
- Security
- Better performance
- High availability
- Planning
- Customer locality
Why affinity? - licensing reasons

- Node-based licensing
  - Software that can run only on certain amount of nodes / CPUs according to some kind of license agreement
Why affinity? - high availability

- VMs that provide failover to each other should not run on the same host to avoid compromising high availability
Why affinity? - security reasons

- Data protection enforcement
  - There are ways to access other VMs' data (e.g.: cache-based timing attacks)
  - Sensitive VMs have to be placed so they only have good neighbours
Why affinity? - performance aspects - VM

- Storage locality
- Network overhead, microservices, databases
Why affinity? - performance aspects - hosts

- Client locality
  - continent locality (with failover) - details later

- Host hardware considerations
  - not all hosts need to be equal
  - better outside connectivity
  - prefer faster CPUs, better NICs, or storage cards

- Dynamic subclusters
  - accounting between teams
  - zero downtime during resize, spare capacity utilization
Why affinity – hardware pass-through

- VMs can use some devices directly
  - But need to handle cases where they are not available
  - Migration of VMs with pass-through devices is needed
Why affinity – operations and planning

- Gradually migrate VMs off a host to lower the number of migrations when going into maintenance
  - Balancing will gradually move VMs away
  - No new VMs will be started on the host unless necessary

- Keep certain services close together (in the same rack or location)
  - But handle maintenance or failures
Customer or management locality

- The goal is to keep services together and close to users to improve performance
Customer or management locality - failure

- Failure of the preferred location does not prevent access to the service, but may hinder performance.
Once the preferred location is restored, service VMs are gradually migrated back to improve performance
  ○ Live migration = no downtime
Affinity types summary

- **Hard (strong)**
  - Does not allow violation

- **Soft (weak)**
  - Best effort, might still allow violations
  - Depends on scheduling policy factors

- **Positive**
- **Negative**

- **VM to VM**
- **VM to Host**
(Solving) affinity conflicts and violations

- During affinity rule creation
  - A ★ B ♡ C ★ A
  - H1 ★ A ★ B ★ H2

- During runtime
  - New rules can create violations
  - Might conflict with balancing
  - Affinity rule enforcement manager
  - Tries to resolve both hard and soft rule violations
Affinity management in oVirt

- **Affinity labels**
  - simple
  - VM to host only
  - strong positive affinity

- **Affinity groups**
  - advanced
  - all types

- **Conflicts**
Affinity labels

Only hosts with all labels can run a VM.

- Finance
- DB

- Strong Finance
- Weak Finance
- DB

✗ DB missing
✓ OK
Affinity groups

- Define a group of VMs and their relationship
  - “No relationship” is important for pure VM to host groups

- Define a group of hosts and their relationship to VMs
Current oVirt support for affinity

- **Affinity Groups**
  - VM – VM, all types – since 3.5 (1)(2)
  - VM – Host, all types – since 4.1 (2)

- **Affinity Labels**
  - VM – Host, strong positive – since 4.0 (2)
  - [https://www.ovirt.org/blog/2016/07/affinity-labels/](https://www.ovirt.org/blog/2016/07/affinity-labels/)

1. Configurable via the user interface
2. Configurable via the API and SDKs
Future ideas

● Allow using Affinity labels in Affinity Groups
  ○ To specify VMs and hosts
  ○ Simplified management
  ○ No need to edit all groups separately
    ■ the updated label will immediately propagate to all rules and relationships

● Inversion of a rule
  ○ Match all VMs except A, B, and C
  ○ Useful for the maintenance planning case
Summary

- Affinity allows you to define (complex) relationship rules between VMs and/or between VMs and hosts.

- Affinity is enforced dynamically and takes the current cluster situation into account.
THANK YOU!

http://wiki.ovirt.org/wiki/Category:SLA
users@ovirt.org
devel@ovirt.org

#ovirt irc.oftc.net