Integration Community of Practice: Installing Fuse Fabric over 4 nodes

Posted by Michael Thirion Apr 21, 2016

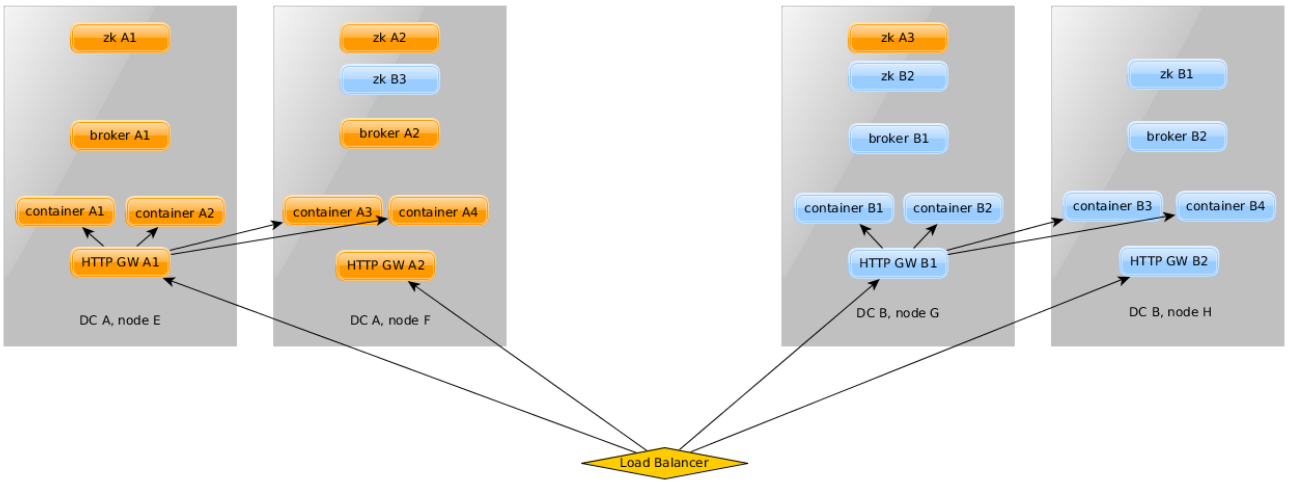
Fabric uses zookeeper to store the information related to its own topology.
Zookeeper is a distributed data store specially designed to avoid split-brain situations. It uses a quorum calculation to guarantee the resiliency of its data and assumes an odd number of members in its cluster (Ensemble) to do so.
To improve the general availability and robustness of the solution (number of nines), it's best to have a zookeeper cluster and to have each cluster member on a separate VM.
This means a good Fabric HA setup (a production-like setup) would requires a minimum of 3 VMs with regards to the zookeeper cluster.
The other components (applications containers, http gateways...) can share those hosts or have their own ones.
Personally, I usually target (but not always reach) a production topology like below, but there are other options:
- 3 VMs for the management (zookeeper, insight...)
- 3 VMs for the technical components, when present (HTTP/MQ gateways, brokers...)
- x VMs for the applications

Here I'll focus on the zookeeper-related issues only.

Most of the companies use several data centers for disaster recovery reason.
If you have an odd number of data centers (like 3), then it's quite easy to create a highly available Fabrics.
However, most of our them still have only 2 data centers.
Though it's supported by Red Hat, it's recommended to avoid both having cross data center Fabrics and having several Fabrics sharing the same hosts.
If you can have 6 VMs over 2 data center (3 VMs per data center), then the best solution is to create 2 Fabrics, one for each data center; each one of them spreading over 3 nodes and being highly available.
Of course, with such a setup, the deployment of the applications will have to be done twice.
However, this is not a big deal because this is supposed to be automated and therefore be transparent for the operations team.
Possibly this could also sometimes naturally fit with a Ble/Green deployment model.

The situation becomes more complex when you are let with only 2 VMs per data center.
In such a case, you'll still create 2 Fabrics, however it will become impossible to avoid having Fabrics sharing the same hosts if each of the 2 Fabric has to be HA.
Here's a diagram showing a setup over 4 VMs. Such a setup was put in place at STC (Saudi Arabia) for the Sapphire project.
Each zookeeper cluster has 2 members in one data center, and an extra one in the opposite DC for high availability.

The "web"-based applications are stateless and therefore the external L3 load balancer can freely spread the load over the 2 data centers whose endpoints are exposed via a (couple of) HTTP gateway.



When you have to have several Fabrics sharing the same host, be extremely careful about avoiding port conflict.
The Zookepper members communicate with each other on the port 2888 & 3888, and the containers finds the registry server on the port 2181 by default.
If the ports are let as default, the 2 zookeeper clusters might start synchronizing data with each other and the containers might register themselves on the wrong Fabric, leading to the corruption of both Fabric.
Thankfully, the --min-port option is there to help avoiding port conflict (in the below example 4000 is chosen randomly, as well as the zookeeper port 4181).

To create Fabric A
$FUSE_HOME/bin/fuse
node E zkA1 > fabric:create --wait-for-provisioning --zookeeper-password admin
node E zkA1 > fabric:container-create-ssh --user <user> --host <node F> --path <path> zkA2
node E zkA1 > fabric:container-create-ssh --user <user> --host <node G> --path <path> zkA3
node E zkA1 > fabric:ensemble-add zkA2 zkA3
node E zkA1 > fabric:container-create-ssh --user <user> --host <node E> --path <path> containerA1
...
==> check the cluster configuration and port number using the fabric's config:list command

To create Fabric B
add **+4000** to the "rmiRegistryPort" and "rmiServerPort" in etc/org.apache.karaf.management.cfg
$FUSE_HOME/bin/fuse
node H zkB1 > fabric:create --wait-for-provisioning --zookeeper-password admin ***--zookeeper-server-port 4181***

node H zkB1 > fabric:container-create-ssh --user <user> --host <node G> --path <path> ***--min-port 4000*** *zkB2*
node H zkB1 > fabric:container-create-ssh --user <user> --host <nodeF> --path <path> ***--min-port 4000*** *zkB3*
node H zkB1 > fabric:ensemble-add zkB2 zkB3
node H zkB1 > fabric:container-create-ssh --user <user> --host <node H> --path <path> ***--min-port 4000*** *containerB1*
...
==> check the cluster configuration and port number using the fabric's config:list command
42 Views

There are no comments on this post