#### 节点down移除volume-peer方案

##### 背景

当节点down掉后，其brick处于不可访问状态，移除volume需要同时移除每个节点上brick信息，所以该节点会导致操作不可执行。

##### 条件

1. 如果节点可以恢复，优先考虑重新上线节点；
2. 当确定需要强制删除volume时，可认为：
3. 该节点不再使用；
4. 该节点数据不需要
5. 该卷vol的数据不需要
6. 可以通过移除其余节点配置信息来完成，但需明确其余节点上的有关故障节点的所有volume信息与故障节点的peer信息都要删除。

##### 场景

* 1. node1,node2,node3组成了一个集群，其中有卷vol1（node1,node2,node3）和vol2(node1,node2,node3)；如果node1故障,此时用户想移除vol1，那么需要同时移除node2,node3上vol1、vol2、peer-1的配置，即删除后整个集群只剩2个节点，没有卷vol。

# umount {/vol1,/vol2}

# gluster volume stop {vol1,vol2}

========node2,node3上均执行=========

# rm -rf /var/lib/glusterd/vols/{vol1,vol2}

# rm -rf /var/run/gluster/vols/{vol1,vol2}

# rm -f /var/lib/glusterd/peers/f20f467a-76d1-4645-9e49-8643a84865b8(uuid)

========之后重启node2,node3,node4,node5,node6glusterd服务=========

# /etc/init.d/glusterd restart

* 1. node1,node2,node3,node4,node5,node6组成了一个集群，其中有卷vol1（node1,node2,node3）和vol2(node4,node5,node6)；如果node1故障,此时用户想移除vol1，那么需要移除移除node2,node3,node4,node5,node6上vol1和peer-1的配置，即删除后整个集群只剩5个节点和卷vol2。

# umount /vol1

# gluster volume stop vol1

========node2,node3,node4,node5,node6上均执行=========

# rm -rf /var/lib/glusterd/vols/{vol1}

# rm -rf /var/run/gluster/vols/{vol1}

# rm -f /var/lib/glusterd/peers/f20f467a-76d1-4645-9e49-8643a84865b8(uuid)

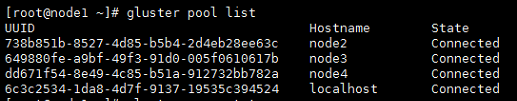
========之后重启node2,node3,node4,node5,node6glusterd服务=========

# /etc/init.d/glusterd restart

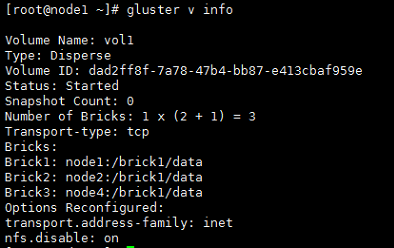
* 1. node1,node2,node3,node4,组成了一个集群，其中有卷vol1（node1,node2,node3）；如果node4故障，则直接gluster peer detach node4 即可。

##### 验证

###### 集群信息

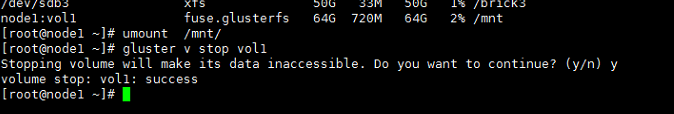


###### 卷信息



###### node4故障

###### 卸载客户端，停止vol1卷



###### 删除node1,node2,.node3节点上的配置文件

# rm -rf /var/lib/glusterd/vols/vol1/

# rm -f /var/lib/glusterd/peers/dd671f54-8e49-4c85-b51a-912732bb782a

# rm -rf /var/run/gluster/vols/vol1/

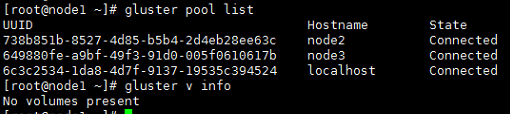






###### 重启node1,node2,node3glusterd服务，查看peer/vol信息

# /etc/init.d/glusterd restart



如果要清理集群，此时将peer detach掉即可。

======此项(7)一般不考虑========

###### 收尾工作

如果故障节点重新上电（非重装），即配置信息均存在，则需清理该节点上所有的数据，它就是一个干净的节点



