存储

ISCSI

存储方式：

DAS：直连式存储，如本地硬盘

NAS：网络附加存储，实际上就是NFS／SAMBA（CIFS）

SAN：存储区域网络，分成基于IP网络的IP SAN 和基于光纤的FC SAN

准备三台虚拟机

Node1.tedu.cn 192.168.4.1 ISCSI服务器

Node2.tedu.cn 192.168.4.2 mysql1

Node3.tedu.cn 192.168.4.3 mysql2

1.添加新虚拟机

]# clone-auto7

]# virsh start rh7\_node1

2.初始化虚拟机

]# virsh console rh7\_node1 连接虚拟机控制台 退出时按ctrl+]

]# LANG=C nmtui 图形界面配置IP, LANG=C 界面改为英语

ifup eth1

]# systemctl restart NetworkManager

1. ISCSI提供块设备

块设备：如硬盘、U盘、光盘

1. ISCSI服务器端为客户端提供块设备，需要起个名，这个名字是IQN

IQN名称规范：iqn.日期.域名的反写.字符串:子字符串

如：iqn.2018-07.cn.tedu.nsd1803:sata

1. 在ISCSI服务器上添加额外的硬盘

可以在虚拟机不关机的情况下，直接添加硬盘，为node1添加一块20G的硬盘

一、安装ISCSI服务

1）装包

]#yum -y install targetcli

1. 为vdb分区

MBR：主引导记录，这种分区格式，最多只能支持四个分区，最大支持2.2TB硬盘（fdisk）

GPT：GUID分区表，可以支持大硬盘，可以支持128个主分区（parted）

]#parted /dev/vdb

(parted) mklabel gpt ( or mktable gpt )

(parted) print

(parted) mkpart primary 1M 50%

(parted) mkpart primary 50% 100%

(parted) quit

]#lsblk

1. 配置ISCSI服务端

]#targetcli

给vdb1起个名字formysql，回到iscsi管理中

/>backstores/block create formysql /dev/vdb1

创建供客户端端访问的名字

/>iscsi/ create iqn.2018-07.cn.tedu:nsd1803

把存储绑定到iqn名字中

/>iscsi/iqn.2018-07.cn.tedu:nsd1803/tpg1/luns create /backstores/block/formysql

配置ACL，限定允许访问的客户端（可以写多个ACL对应客户端）

/> iscsi/iqn.2018-07.cn.tedu:nsd1803/tpg1/acls create iqn.2018-07.cn.tedu:node2

/> iscsi/iqn.2018-07.cn.tedu:nsd1803/tpg1/acls create iqn.2018-07.cn.tedu:node3

保存并退出

/>saveconfig

/>exit

1. 配置node2为ISCSI客户端
2. 装包

]# iscsi-initiator-utils.i686

1. 客户端修改iqn的名字

]# vim /etc/iscsi/initiatorname.iscsi

InitiatorName=iqn.2018-07.cn.tedu:node2

1. 发现存储端，此时可以发现名字，但是不会多出硬盘

]# man iscsiadm

]#iscsiadm --mode discoverydb --type sendtargets --portal 192.168.4.1 --discover

1. 重启iscsi服务，出现新硬盘

]# systemctl restart iscsi

]# lsblk

]# systemctl enable iscsi

]# systemctl enable iscsid

iscsi服务用于自动登录，iscsid是守护进程

1. 安装mariadb-server，将sda分区格式化，挂载到mariadb的工作目录下

登录node2

]# parted /dev/sda

(parted) mklabel gpt

(parted) mkpart primary 1M 100%

]#mkfs.ext4 /dev/sda1

]#df -h

]#yum -y install mariadb-server

]#mount /dev/sda1 /var/lib/mysql

]#chown mysql:mysql /var/lib/mysql

]#systemctl start mariadb

1. 创建mariadb,创建库、表

当node2节点发生故障时，可以使用node3进行替换

注意：千万不要使用两个节点同时挂载相同的文件系统（分区）

如果多个节点同时挂载这个文件系统，很可能会导致文件系统损坏，数据会丢失！因为XFS／EXT3／EXT4这些文件系统都是单节点文件系统。红帽的GFS才能支持多节点同时挂载

udev

udev：动态管理硬件文件的方法。如：把U盘接入到主机，主机会多一个硬盘文件，把U盘移除的时候，硬盘文件消失。

1. udev规则文件存放目录/etc/udev/rules.d/
2. 规则文件命名：数字-名字.rules

数字大小顺序是规则文件执行的顺序

1. 应用udev给iscsi磁盘创建一个软链接叫idisk
2. 查看iscsi（sda）在/sys/目录中的位置

udevadm info --query=path --name=/dev/sda1

1. 通过（1）上一步的路径，查看iscsi磁盘的信息（以和其他设备区别）

udevadm info --query=all --attribute-walk --path=/devices/platform/host2/session1/target2:0:0/2:0:0:0/block/sda/sda1

1. 创建规则文件，以.rules结尾

vim /etc/udev/rules.d/90-iscsi.rules

KERNEL=="sd[a-z]\*",

ACTION=="add",

SUBSYSTEMS=="scsi",

ATTRS{model}=="formysql ", SYMLINK+="idisk%n"

KERNEL==“”表示内核识别出来的设备名

ACTION==“ADD”表示新接入设备

SUBSYSTEMS和ATTRS（model）是第（2）步查到的

SYMLINK表示创建符号链接，＋＝表示额外创建的，％n是分区号,如sda3中的3

1. 只有把磁盘移除再接入才能生效或是重启系统

]# systemctl stop mariadb

]# umount /dev/sda1

]#iscsiadm --mode node --targetname iqn.2018-07.cn.tedu:nsd1803 --portal 192.168.4.1:3260 --logout //targetname是iscsi的名字，不是acl的

]# iscsiadm --mode discoverydb --type sendtargets --portal 192.168.4.1 --discover

]# systemctl restart iscsi

]# ll /dev/idisk\*

测试：mount /dev/idisk1 /var/lib/mysql

NFS网络文件系统 应用较少

实现网络共享的，用于unix-like（类unix）系统间的共享

端口号：2049，基于RPC（远程过程调用，端口号111）服务，NFS只提供了共享功能，底层数据传输交给RPC服务。

1. 只读共享
2. node1作为服务端
3. 装包

nfs-utils rpcbind

2）创建共享目录

]# mkdir -pv /nfsroot/nfsro

3）修改配置文件

]# vim /etc/exports

/nfsroot/nfsro \*(ro) 允许所有地址以只读方式访问

4）启动服务并验证

]# systemctl restart nfs

]# showmount -e 192.168.4.1

1. node3作为客户端

]# mkdir /mnt/nfsshare

]# mount 192.168.4.1:/nfsroot/nfsro /mnt/nfsshare

1. 读写共享

不管是NFS，还是SAMBA，还是FTP，只要是读写共享，必须注意本地权限和配置文件内的授权

1.node1作为服务端

1）创建用于读写的共享目录

]# mkdir -m 777 /nfsroot/nfsrw

2）修改配置文件

]# vim /etc/exports

/nfsroot/nfsrw 192.168.4.\*(rw,sync) //sync同步

]#exportfs -rv //reload,不用重启服务

2.node3作为客户端

]# mkdir /mnt/nfswrite

]# mount 192.168.4.1:/nfsroot/nfsrw /mnt/nfswrite

1. NFS选项

1） no\_root\_squash (出于安全考虑，不建议)表示保留来自客户端的属主属组root,如果客户端是用普通用户写入，则显示的是客户端普通用户的UID，如果本机的UID刚好和这个UID重合，则显示本机的这个UID号的用户，建议创建用户的时候手动指定UID（useradd -u），默认写入到共享目录中的文件属主属组是nfsnobody

2）all\_squash 作用是客户端任何用户写入的文件属主属组都是nfsnobody

Multipath多路径

多路径：当主机到共享存储有多条路径，每条路径都会被识别为一个设备。多路径配置可以将多个设备合并成一个虚拟设备

1. 为node1和node3再配置一个网络192.168.2.0/24
2. node3在192.168.2.0网络上发现共享存储

]# iscsiadm --mode discoverydb --type sendtargets --portal 192.168.2.1 --discover

]# systemctl restart iscsi

]# lsblk //发现了新的设备

1. 主机通过WWID（全球识别符）来判断哪些路径上的设备是同一设备（sda和sdb为同一WWID，视为同一设备）

]# /lib/udev/scsi\_id --whitelisted --device=/dev/sda //查询sda的WWID

4、安装多路径设备软件（客户端）

node3上操作：

装包device-mapper-multipath

1. 合并出来的多路径设备，一般起名为mpath[a-z]
2. 配置多路径
3. 生成配置文件，不使用用户友好名称，默认没有（2）这个文件

]# mpathconf --user\_friendly\_names n //是否使用用户友好名称

1. 修改配置文件

]# vim /etc/multipath.conf

在文件末尾追加以下内容：

multipaths {

multipath {

wwid “360014059b74b9388dac42c7b854e22b2”

alias mpatha //针对wwid取别名

}

}

1. 多路径设备出现在/dev/mapper/目录下

]# systemctl restart multipathd

]# ls /dev/mapper

]# mount /dev/mapper/mapth1 /var/lib/mysql

]# multipath -ll //查看多路径信息

4）测试多路径是否生效

mount /dev/mapper/mpatha1 /var/lib/mysql

node1如果有一个网卡坏掉，会自动切换挂载路径到另外一条上，切换需要一点点时间，iscsi依然可以通过另外一条路径访问