## 存储技术概述

### 存储类型：

DAS Direct Attachment Storage 直接附加存储

磁盘总线/接口类型：

SCSI/SATA/SAS接口

SCSI: 600M/s

SATA: 6G/s

SAS: 6G/s

NAS Network Attachment storage 网络附加存储

NFS服务器，应用服务器作为共享存储

借助于NFS[针对Linux]/CIFS[针对windows]实现文件系统的共享

属于文件系统级别的共享

实现的方式：

NFS服务器

NAS存储设备

SAN Storage Area Network 存储区域网络

属于块级别的共享

iSCSI 协议

实现的方式：

磁盘阵列柜 EMC/IBM/HP/DELL/华为

FC SAN 光纤

IP SAN TCP/IP网络

示例01：基于openfiler配置SAN存储

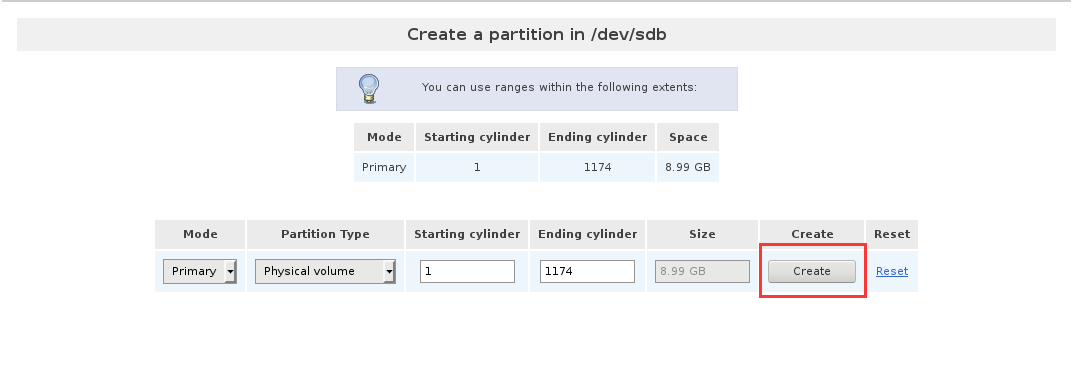
1、安装openfiler

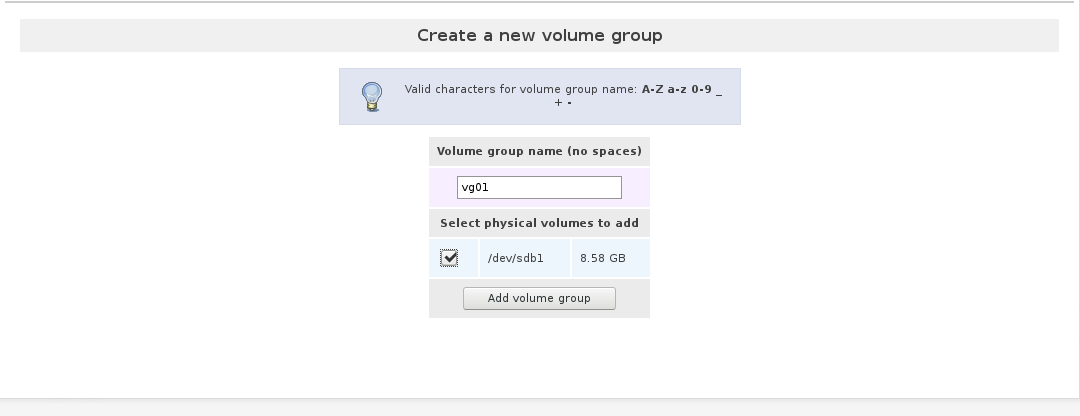
通过如下url地址访问web配置界面 <https://192.168.122.100:446/>

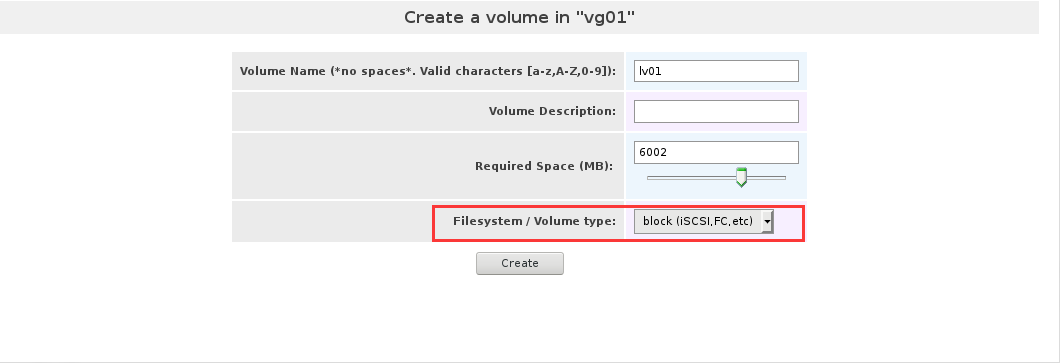
默认用户名：openfiler 密码：password

2、配置iSCSI协议共享逻辑卷

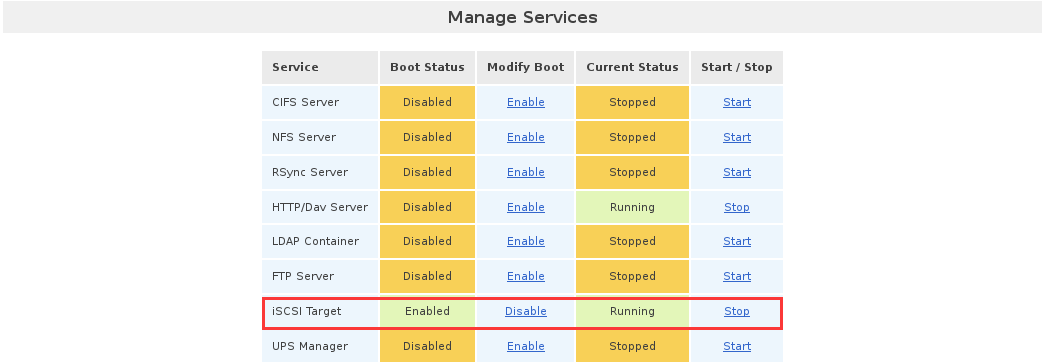
1) 准备逻辑卷/dev/vg01/lv01



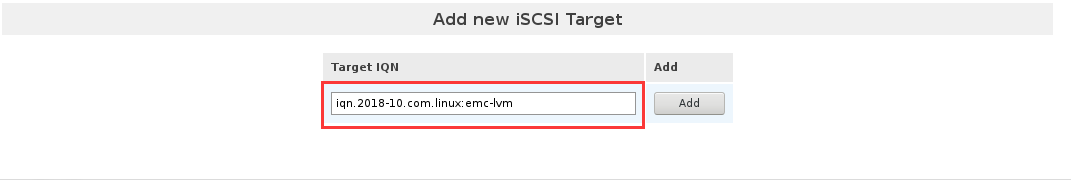


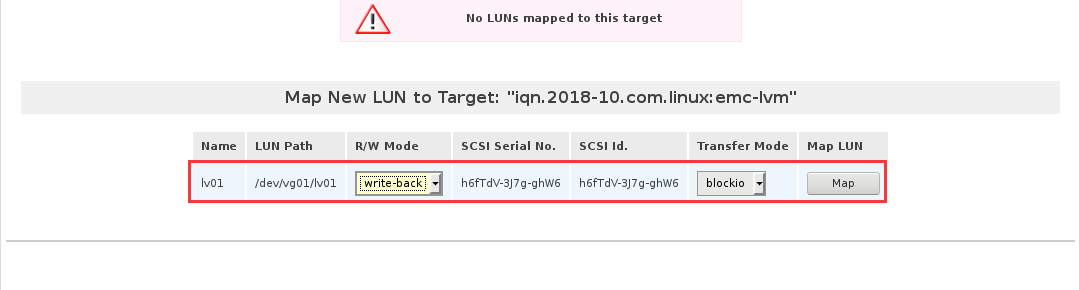


2) 启动iscsi服务

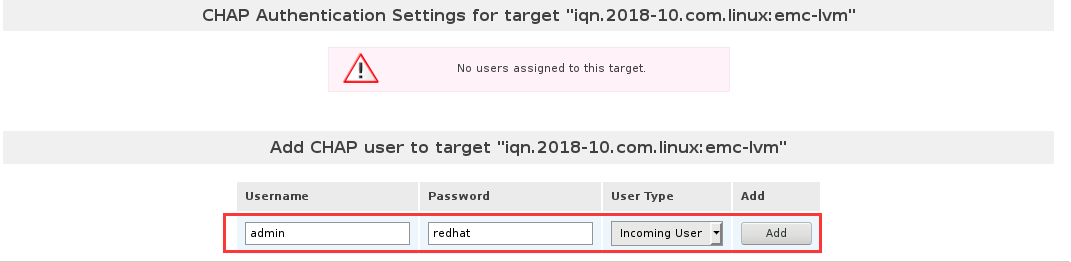


3) 创建iscsi的共享名称，关联逻辑卷

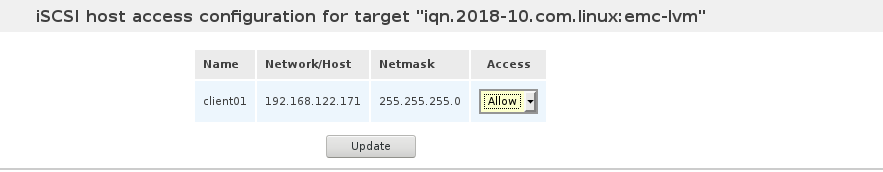




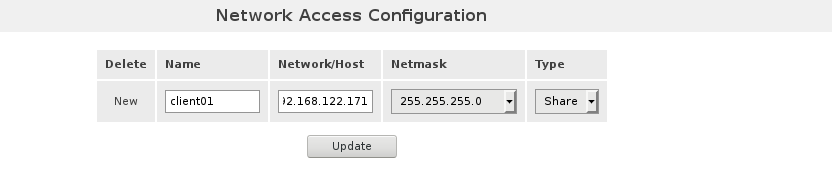
设置用户名密码认证



基于客户端IP地址做访问控制



4) 配置客户端IP地址



2、配置客户端连接存储

[root@client ~]# yum install -y iscsi-initiator-utils

[root@client ~]# iscsiadm -m discovery -t st -p 192.168.122.100

iscsiadm: No portals found

出现以上错误提示，需要注释掉openfiler端/etc/initiators.deny文件中内容

[root@client ~]# iscsiadm -m discovery -t st -p 192.168.122.100

192.168.122.100:3260,1 iqn.2018-10.com.linux:emc-lvm

[root@client ~]#

设置用户名密码认证

[root@client ~]# vim /etc/iscsi/iscsid.conf

57 node.session.auth.authmethod = CHAP

61 node.session.auth.username = admin

62 node.session.auth.password = redhat

[root@client ~]# systemctl restart iscsid

[root@client ~]# iscsiadm -m node -T iqn.2018-10.com.linux:emc-lvm -p 192.168.122.100 -l

建议:

1、使用UUID挂载

[root@client ~]# blkid /dev/sda1

/dev/sda1: UUID="e78effb7-5195-4fd3-b650-37323c19db6d" TYPE="ext4"

2、断开存储

[root@client ~]# iscsiadm -m node -T iqn.2018-10.com.linux:emc-lvm -p 192.168.122.100 -u

### udev机制

在HA集群环境，由于集群中机器的配置不同，可能会导致连接后端存储时，所识别的设备名称不一致；此时，可以通过udev机制实现配置相同的名称

示例：

1、检索后端存储的属性信息

[root@node01 ~]# udevadm info -a -p /block/sdb

2、编辑udev命令规则文件

[root@node01 ~]# cat /etc/udev/rules.d/100-disk.rules

SUBSYSTEM=="block",ATTRS{vendor}=="LIO-ORG ",ATTR{size}=="14680064",SYMLINK+="websan"

[root@node01 ~]# systemctl restart systemd-udev-trigger.service

[root@node01 ~]# ls -l /dev/websan

lrwxrwxrwx 1 root root 3 10月 8 15:00 /dev/websan -> sdb

### mutipath多路径

作用：通过双线连接后端存储实现线路的备份

出现问题：

在FC SAN中，多路径会导致前端的应用服务器在本地映射出多块磁盘

1、安装多路径软件

[root@node01 ~]# yum install -y device-mapper-multipath

2、编辑multipath的配置文件

[root@node01 ~]# vim /etc/multipath.conf

defaults {

user\_friendly\_names yes

}

blacklist {

devnode "vda"

}

3、启动multipathd服务

[root@node01 ~]# systemctl start multipathd.service

[root@node01 ~]# systemctl enable multipathd.service

4、查看多路径设备

[root@node01 ~]# multipath -ll

mpatha (36001405c7f64fc72a264830bce973d27) dm-2 LIO-ORG ,bk01

size=7.0G features='0' hwhandler='0' wp=rw

|-+- policy='service-time 0' prio=1 status=active

| `- 3:0:0:0 sdb 8:16 active ready running

`-+- policy='service-time 0' prio=1 status=enabled

`- 4:0:0:0 sda 8:0 active ready running

[root@node01 ~]# lsblk

NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT

sda 8:0 0 7G 0 disk

└─mpatha 253:2 0 7G 0 mpath

sdb 8:16 0 7G 0 disk

└─mpatha 253:2 0 7G 0 mpath

[root@node01 ~]# ls -l /dev/mapper/mpatha

lrwxrwxrwx 1 root root 7 10月 8 16:35 /dev/mapper/mpatha -> ../dm-2

[root@node01 ~]# ls -l /dev/dm-2

brw-rw---- 1 root disk 253, 2 10月 8 16:35 /dev/dm-2

### GFS 全局文件系统

GFS -------- Global File System

全局文件系统

文件系统类型：

单机文件系统

ext4 xfs swap ext3

网络文件系统

NFS, CIFS

作用：在网络中共享数据

全局文件系统

作用：LB集群所有的real server能够实时识别文件系统变化

dlm机制 分布式锁机制

借助dlm机制，能够通过网络传递文件系统的变化信息

借助HA集群的message layer

示例： cLVM + GFS

环境描述：

node01 192.168.122.101

node02 192.168.122.102

iSCSI 192.168.122.103

1、在node01, node02上使用pacemaker创建HA集群

1) 在所有节点安装pacemaker软件

[root@node01 ~]# yum install -y pacemaker corosync pcs

[root@node01 ~]# systemctl start pcsd

[root@node01 ~]# systemctl enable pcsd

2) 所有节点设置hacluster用户密码

[root@node01 ~]# echo "redhat" | passwd --stdin hacluster

3) 在任意节点认证用户

[root@node01 ~]# pcs cluster auth node01 node02

4) 创建并启动HA集群

[root@node01 ~]# pcs cluster setup --start --enable --name mycluster node01 node02

[root@node01 ~]# pcs cluster status

Cluster Status:

Stack: corosync

Current DC: node02 (version 1.1.16-12.el7-94ff4df) - partition with quorum

Last updated: Tue Mar 12 15:54:44 2019

Last change: Tue Mar 12 15:54:42 2019 by hacluster via crmd on node02

2 nodes configured

0 resources configured

PCSD Status:

node01: Online

node02: Online

5) 禁用stonith

[root@node01 ~]# crm\_verify -L -V

error: unpack\_resources: Resource start-up disabled since no STONITH resources have been defined

error: unpack\_resources: Either configure some or disable STONITH with the stonith-enabled option

error: unpack\_resources: NOTE: Clusters with shared data need STONITH to ensure data integrity

Errors found during check: config not valid

[root@node01 ~]#

[root@node01 ~]# pcs property set stonith-enabled=false

[root@node01 ~]# pcs property set no-quorum-policy=ignore

[root@node01 ~]#

[root@node01 ~]# crm\_verify -L -V

2、配置iscsi存储，节点连接存储 (略)

3、在所有节点安装cLVM及gfs需要的软件

[root@node01 ~]# yum install -y lvm2-cluster gfs2-utils

[root@node01 ~]# lvmconf --enable-cluster

[root@node01 ~]# reboot

4、在任意节点创建cLVM及gfs需要的资源

[root@node01 ~]# pcs resource create dlm ocf:pacemaker:controld \

> allow\_stonith\_disabled=true \

> op monitor interval=30s clone interleave=true ordered=true

[root@node01 ~]#

[root@node01 ~]# pcs resource create clvmd ocf:heartbeat:clvm \

> op monitor interval=30s clone interleave=true ordered=true

[root@node01 ~]# pcs resource show

Clone Set: dlm-clone [dlm]

Started: [ node01 node02 ]

Clone Set: clvmd-clone [clvmd]

Started: [ node01 node02 ]

5、创建资源约束，保证资源正常的启动顺序、多个资源同进同退

[root@node01 ~]# pcs constraint order start dlm-clone then clvmd-clone

Adding dlm-clone clvmd-clone (kind: Mandatory) (Options: first-action=start then-action=start)

[root@node01 ~]# pcs constraint colocation add clvmd-clone with dlm-clone

[root@node01 ~]#

[root@node01 ~]# pcs constraint show

Location Constraints:

Ordering Constraints:

start dlm-clone then start clvmd-clone (kind:Mandatory)

Colocation Constraints:

clvmd-clone with dlm-clone (score:INFINITY)

Ticket Constraints:

6、在任意节点创建逻辑卷

[root@node01 ~]# pvcreate /dev/sda

[root@node01 ~]# vgcreate vg01 /dev/sda

[root@node01 ~]# lvcreate -L 5G -n lv01 vg01

7、在任意节点格式化gfs2文件系统

[root@node01 ~]# mkfs.gfs2 -p lock\_dlm -t mycluster:tb01 -j 4 /dev/vg01/lv01

-p lock\_dml 指定锁协议

-t mycluster:tb01 指定锁表名称

-j n：指定日志区域个数，>=节点数

[root@node01 ~]# blkid /dev/vg01/lv01

/dev/vg01/lv01: LABEL="mycluster:tb01" UUID="b9c9bf8a-685b-4c54-b276-8f88d72aa514" TYPE="gfs2"

8、在任意节点上创建资源挂载集群逻辑卷

[root@node01 ~]# pcs resource create fs\_test Filesystem \

> device=/dev/vg01/lv01 \

> directory=/mnt \

> fstype=gfs2 \

> options="noatime,nodiratime" op monitor interval=10s clone interleave=true

[root@node01 ~]# pcs resource show

Clone Set: dlm-clone [dlm]

Started: [ node01 node02 ]

Clone Set: clvmd-clone [clvmd]

Started: [ node01 node02 ]

Clone Set: fs\_test-clone [fs\_test]

Started: [ node01 node02 ]

[root@node01 ~]# pcs constraint order start clvmd-clone then fs\_test-clone

Adding clvmd-clone fs\_test-clone (kind: Mandatory) (Options: first-action=start then-action=start)

[root@node01 ~]# pcs constraint colocation add fs\_test-clone with clvmd-clone

[root@node01 ~]# pcs constraint show

Location Constraints:

Ordering Constraints:

start dlm-clone then start clvmd-clone (kind:Mandatory)

start clvmd-clone then start fs\_test-clone (kind:Mandatory)

Colocation Constraints:

clvmd-clone with dlm-clone (score:INFINITY)

fs\_test-clone with clvmd-clone (score:INFINITY)

Ticket Constraints:

9、测试

1) 在所有节点查看逻辑卷挂载信息

[root@node02 ~]# df -hT

/dev/mapper/vg01-lv01 gfs2 5.0G 518M 4.5G 11% /mnt

2) 测试写入文件

[root@node02 ~]# ls /mnt/

1.txt

[root@node02 ~]#

[root@node02 ~]# touch /mnt/2.txt

[root@node02 ~]# ls /mnt/

1.txt 2.txt