# Linux 网络存储服务器

前言：

目前计算机领域的存储解决方案主要有直连存储（DAS），存储区域网络（SAN）以及网络附加存储（NAS）三种。

DAS（Direct-Attached Storage）指的是主机总线直接通过SCSI接口与存储设备相连，这种连接方式主要应用在家庭个人电脑环境中。

SAN（Storage Area Network）指的是一整套存储网络的解决方案，SAN采用的是光纤通信技术，SAN就是通过光纤交换机将服务器与存储设置连接在一起的网络整体环境，SAN的优势在于其先进的光纤通信技术，为当今爆炸式的数据增长环境提供了快速高效的存储方案。

NAS（network-Attached Storage）可以使用普通的网络环境，通过以太网交换机等设置连接服务器与存储设备，NAS的优势在于无需对网络环境进行改造，NAS可以使用现有的网络环境，而且不用厂家的设备只要采用TCP/IP协议，就可以满足设备之间对兼容性的要求，NFS与CIFS就属于网络附加存储的解决方案，它们都提供了对文件系统的共享。

不管是哦NAS还是SAN，它们都可以实现企业环境中服务器与存储设备相分离的要求。这样服务器就可以专注于服务器业务，而存储设置则专注于数据存储的速度于安全，并且由于服务器于存储设置使分离的，这样多台服务器就可以加载同一存储设备中的数据没实现数据的集中、统一管理。由于传统的SAN需要价格昂贵的光纤设备作为基础，所以现在又有了基于IP技术的SAN，可以通过IP网络进行数据存储，服务器在不进行任何改造的情况下使用现有的以太网卡就可以访问IP SAN（ISCSI就是这样的技术）。

NSA与SAN的主要区别在于，NAS共享的使文件系统，而SAN共享的使块设备。

## iSCSI网络存储

简介：

iSCSI是典型的IP-SAN技术，是基于因特网的ASN存储技术，该技术是的我们可以在基于IP协议的网络上传输SCSI命令，在iSCSI环境中，客户端发送SCSI命令给远程的SCSI存储设备，实现数据的存储与备份功能。iSCSI使用TCP的860与3260端口进行通信。iSCSI与SCSI最大的区别在于iSCSI摆脱了存储设备的距离限制，使得任何主机都可以通过局域网或广域网访问我们的存储设备，而对于数据中心而言，这是只管重要的。

### 1，部署iSCSI服务

在CentOS7平台上部署iSCSI服务需要安装 targetcli 软件，安装完成后，可以通过命令的方式部署iSCSI服务。如果需要使用命令配置服务，可以参考 targetcli 命令的 man手册，其中 QUICKSTART 部分有比较详细的示例。

在配置存储单元前，需要搞清楚两个概念：

LUN、IQN。

LUN（Logical Unit Number）是设备的逻辑单元号，一般为一个数字，我们使用LUN来表示存储设备。

IQN（iSCSI Qualified Name）为iSCSI合格名称，一般格式为iqn.yyyy-mm.<reversed doamin name>:identifier，其中,yyyy表示年，mm表示月，reversed domain name 是域名和的反写，identifier 为表示名称。

如：iSCSI服务器域名为www.example.com，则IQN全称可以写成：

iqn.2012-10.com.example.www:disk1。

这种IQN的书写格式不是创建iSCSI服务必须遵守的，但推荐使用这样的标准格式。

部署ISCSI服务器时，首先需要安装targetcli软件。另外，为了给其他服务器提供存储，ISCSI服务器上应该有足够的存储设备，可以使用fdisk查看计算中的磁盘设备列表。

[root@centos ~]# yum -y install targetcli

[root@centos ~]# fdisk -l | grep /dev/sd

Disk /dev/sda: 21.5GB，21474836480 bytes

/dev/sda1 \* 2048 1026047 512000 83 Linux

/dev/sda2 1026048 41943039 20458496 8e Linux LVM

Disk /dev/sdb: 10GB，1099511627776 bytes

[root@centos ~]# targetcli #下方红色字体需要自己输入

Warning： Could not load perferences file /root/.targetcli/prefs.bin.

targetcli shell version 2.1.fb41

Copyright 2011-2013 by Datera, Inc and others.

For help on commands, type 'help'.

/> ls # 查看配置

.....

/> backstores/block create iscsi\_store /dev/sdb

# 定义后端存储，这里个/dev/sdb做共享存储

/> /iscsi create iqn.2016-6.com.example:server

# 定义共享名称为（IQN）为：iqn.2016-06.com.example:server

/> /iscsi/iqn.2016-06.com.example:server/tpg1/acls create \

> iqn.2016-06.com.example:desktop

# 设置访问控制权限，客户端配置文件需要配置iqn.2016-06.com.example:desktop，才可以访问iqn.2016-06.com.example:server这个共享磁盘

/> iscsi/iqn.2016-06.com.example:server/tgp1/luns create \

> /backstores/block/iscsi\_store

# 将IQN共享名称和前面定义的后端存储绑定在一次，客户端访问IQN将获取该存储空间

/> iscsi/iqn.2016-06.com.example:server/tgp1/portals create 0.0.0.0

# 配置服务器监听的IP地址和端口，0.0.0.0为本机所有IP，默认端口为3260

/> exit # 退出配置程序，该程序会自动保存所有操作

[root@centos ~]# systemctl start target #启动服务

[root@centos ~]# ststemctl enable target #设置开机启动

[root@centos ~]# firewall-cmd --permanent --add-port=3260/tcp

[root@centos ~]# firewall-cmd --reload

#设置防火墙规则允许3260端口，并重新加载配置

### 2，客户端访问

Linux客户端访问iSCSI服务器需要安装iscsi-initiator-utils软件包。

第一步，配置客户端IQN配置我呢见必须与服务器配置中的一致（iqn.2016-06.com.example:desktop）

第二步，通过iscsiadm命令可以防线服务器端iSCSI的IQN，在通过login选项加载服务器的IQN。

客户端主机完成以上操作后，在通过fdisk -l 命令可以防线计算机中多出了块磁盘。最后对磁盘进行分区格式化后即可使用该ISCSI磁盘存储设备。

注意：有时在客户端使用discoverydb无法发现服务器IQN信息，此时可以检查防火墙是否设置正确，并检查有没有设置账户及IP地址的ACL限制，最后在系统换将中重启服务器才可以被客户端识别新的IQN。

假设服务器的IP地址为192.168.0.106，客户端通过如下命令挂载该服务器的共享磁盘：

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

[root@centos ~]# vim /etc/iscsi/initiatorname.iscsi

**InitiatorName=iqn.2016-06.com.example:desktop**

[root@centos ~]# systemctl start iscsi

[root@centos ~]# systemctl enable iscsi

[root@centos ~]# iscsiadm -m discovery -t st -p 192.168.0.106

# 查看192.168.0.106服务器的ISCSI IQN信息

[root@centos ~]# iscsiadm -m node -T iqn.2016-06.com.example:server -l

# login挂载共享磁盘

[root@centos ~]# fdisk -l

..部分内容省略...

Disk /dev/sdb: 10GB, 1099511627776 bytes

## Rsync文件同步

简介：

Rsync（remote sync）时UNIX及类UNIX平台下一筐神奇的数据镜像备份软件，它不像FTP或其他文件传输服务那样需要进行全备份，Rsync可以根据数据的变化差异备份，从而减少数据流量，提高工作效率。你可以使用它进行本地数据或远程数据的复制，Rsync可以使用SSH安全隧道进行加密数据传输。Rsync服务端定义源数据，Rsync客户端仅在源数据发生改变后才会从服务器上实际复制至本地，如果源数据在服务端被删除，则客户端数据也会被删除，以确保主机之间的数据是同步的。Rsync使用TCP 873端口。

### 1，搭建Rsync服务器

部署Rsync服务器需要创建至少一个配置文件，默认在系统中并不存在Rsync配置文件，对于服务器而言，配置文件创建完成后，使用守护进程模式启动rsync程序即可。

Rsync是Andrew Tridgell（Samba的作者）与Paul Mackerras 合作开发的软件，所以Rsync配置文件与Samba配置文件比较相似。

Rsync只要分为三个配置文件，分别是：

rsyncd.conf（主配置文件）

rsyncd.secrets（密码文件）

rsyncd.motd（服务器信息文件）

[root@centos ~]# yum -y install rsync # 安装rsync

[root@centos ~]# mkdir /common # 创建共享目录

[root@centos ~]# cp /etc/init.d/\* /common/

[root@centos ~]# vim /etc/rsyncd.conf

配置/etc/rsyncd.conf配置文件：

#/etc/rsyncd.conf

motd file = /etc/rstncd.motd

# 设置服务器信息提示文件名称，在该文件中编写提示信息

transfer logging = yes

# 开启Rsync数据传输日志功能

log file = /var/log/rsyncd.log

# 设置日志文件名称，可以通过log format参数设置日志格式

pid file = /var/run/rsyncd.pid

# 设置Rsync进程保存文件名称

locak file = /var/run/rsync.lock

# 设置文件锁名称

port = 873

# 设置服务器监听的端口号，默认为873

address = 192.168.0.106

# 设置服务器所监听的网卡接口的IP地址，这里服务器IP地址为192.168.0.106

uid = noboay

# 设置进行数据传输时所使用的账户名称或ID号，默认使用noboay

git = nobody

# 设置进行数据传输时所使用的组名称或GID号，默认使用nobody

user chroot = no

# 设置user chroot为yes后，rsync会首先进行chroot设置，将根映射到path参数路径下，对客户端而言，系统的根就是path参数所指定的路径。但这样做需要root权限，并且在同步符号连接资料时仅会同步名称，而内容将不会同步。

read only = yes

# 是否允许客户端上传数据，这里设置为只读

max connections = 10

# 设置并发连接数，0代表无限制。超出并发数后，如果依然有客户端连接请求，则将会收到稍后重试的提示消息

[common]

# 模块，Rsync通过模块定义同步的目录，模块已[name]的形式定义，这与Samba定义共享目录时一样的效果。在Rsync中也可以定义多个模块。

comment = Web content

# comment 定义注释说明字串

path = /common

# 同步目录的真实路径通过path指定

ignore errors

# 忽略一些I/O错误

#exclude = test/

# exclude可以指定列外的目录，即将common目录下的某个目录设置为不同步数据

auth users = tom,jerry

# 设置允许连接服务器的账户，账户可以是系统中不存在的用户

secrets file = /etc/rsyncd.secrets

# 设置密码验证文件名称，注意该文件的权限要求为只读，建议权限为600，仅在设置auth users参数后有效

hosts allow = 192.168.0.0/255.255.255.0

# 设置允许那些主机可以同步数据，可以是单个IP，也可以是网段，多个IP与网段之间使用空格分隔

host deny = \*

# 设置拒绝所有（除hosts allow定义的主机外）

list = false

# 客户端请求显示模块列表时，本模块名称是否显示，默认为true

接下来:

创建密码文件/etc/rsyncd.secrets,在该文件中输入配置文件中定义的两个用户名，并设置密码。

[root@centos ~]# echo "tom:pass" > /etc/rsyncd.secrets

[root@centos ~]# echo "jerry:111" >> /etc/rsyncd.secrets

[root@centos ~]# chmod 600 /etc/rsyncd.secrets

需要注意的是，密码文件不可以对所有的人开放可读权限，为了安全，建议设置权限为600。

创建服务器提示信息文件，并想文件中导入欢迎词

[root@centos ~]# echo "welcome to access" > /etc/rsyncd.motd

Rsync默认不是开机启动服务，为了实现开启启动Rsync服务，可以通过echo 将rsync --daemon 追加至开机启动文件/etc/rc.local,最后设置防火墙开启873端口的入展数据请求。

[root@centos ~]# rsync --daemon #开启服务

[root@centos ~]# echo "/usr/bin/rsync --daemon" >> /etc/rc.local

[root@centos ~]# firewall-cmd --permanent --add-port=873/tcp

#添加防火墙规则，允许873端口的数据访问

### 2，客户端同步数据

客户端也需要安装rsync软件，但客户端主机不需要--daemon选项。

[root@centos ~]# yum -y install rsync

[root@centos ~]# rsync -vzrtopg --progress [tom@192.168.0.106::common](mailto:tom@192.168.0.106::common) /test

rsync命令的描述和用法如下：

描述：一个快速、多功能的远程（或本地）数据复制工具

用法：SRC表示源路径，DEST表示目标路径

本地复制

rsync [选项] SRC ... [DEST]

通过远程shell复制

下载数据 rsync [[选项] [USER@] HOST:SRC... [DEST]](mailto:[选项][USER@]HOST:SRC...[DEST])

上传数据 rsync [选项] SRC... [USER@] HOST:DEST

通过rsync进程复制

下载数据 rsync [[选项] [USER@] HOST:SRC... [DEST]](mailto:[选项][USER@]HOST:SRC...[DEST])

rsync [选项] rsync://[USER@] HOST[:PORT]/SRC... [DEST]

上传数据 rsync [选项] SRC... [USER@] HOST:DEST

rsync [选项] SRC... rsync://[USER@]HOST[:PORT]/DEST

选项：

-v 显示详细信息

-q 静默模式，无错误信息

-a 归档模式，主要保留文件属性，等同于-rlptgoD

-r 递归

-b 如果目标路径已经存在同名文件，将旧的文件重命名为 ~filename,可以使用--suffix指定不同的备份前缀

--back-dir 将备份文件保存得指定目录

--suffix 指定备份文件前缀

-u 如果目标地址中的文件比将要下载的文件新，则不执行 同步。

也就是说，不会用旧的文件覆盖新的文件

-l 保留符号连接

-p 保留文件权限属性

-H 保留硬连接

-A 保留ACL权限

-X 保留文件附加属性

-o 保留文件所有者属性

-g 保留文件所属组属性

--devices 保留设备文件

--specials 保留特殊文件

-D 等同于--devices --specials

-t 保留修改时间属性

-W 不做增量检查，直接复制全部文件

-e 指定远程shell

--existing 仅同步目标路径中已经有的文件，不下载源路径下新的 文件

--delete 删除那些仅在目标路径中存在的文件（源路径中不存在）

-z 传输过程中对数据进行压缩

--include=PATTERN 匹配不排除的文件

--exclude=PATTERN 匹配需要排除的文件

--progress 显示数据传输的进度信息

--partial 保留因故障未传输完成的文件

-P 等同于--progress --partial

--password-file=FILE 指定密码文件，将密码写入文件，实现非交互式数据同 步

--list-only 仅列出服务器模块列表，需要rsync服务器设置list = true

实例：

[root@centos ~]# rsync -t \*.c foot:src/

# 将本机当前目录下的以.c结尾的文件复制至foo主机的src目录下

[root@centos ~]# rsync -avz foo:src/bar /data/tmp

# 从foo主机上以递归方式将src/bar目录复制至本机/data/tmp目录

[root@centos ~]# rsync -avz foot:src/bar/ /data/tmp

# 从foot主机以上以递归方式将src/bar 目录下的所有内容复制至本机/data/tmp目录，但在/data/tmp目录下不会创建bar目录

[root@centos ~]# rsync -avz /src/foo /dest

# 将本机/src/foo 目录复制至/dest目录

[root@centos ~]# rsync -avz [tom@192.168.0.106::common](mailto:tom@192.168.0.106::common) /test3

# 使用tom账户连接远程192.168.0.106主机的rsync进程，将common模块定义的path路径下载至本地test3目录

[root@centos ~]# rsync -avz tom@192.168.0.106：：common /dest

# 匿名下载192.168.0.106服务器的common模块至本地/dest目录

[root@centos ~]# rsync --list-only [tom@192.168.0.106::](mailto:tom@192.168.0.106::)

# 显示192.168.0.106服务器的所有模块名称，需要服务器端配置list = true才会显示

[root@centos ~]# echo "pass" > rsync.pass

[root@centos ~]# rsync -avz --delete --password-file=rsync.pass \

> [tom@192.168.0.106::comm](mailto:tom@192.168.0.106::comm) /dest

# 使用指定密码文件，操作远程rsync服务端的数据

客户端定期对Rsync服务器数据进行备份，编写Shell脚本，通过计划任务crontab实现数据的定期被封工作。

#!/bin/bash

#This script does backup throug rsync.

#Data:2015-10-1

#Version:1.0 beta

#

export PATH=/bin:/usr/bin:/usr/local/bin

SRC=common

DEST=/data

Server=192.168.0.106

User=tom

#password file must not be other-accessible.

Passfile=/root/rsync.pass

#If the DEST directory not found,then create one

[ ! -d $DEST ] && mkdir $DEST

[ ! -e $Passfile ] && exit 2

rsync -az --delete --password-file=$Passfile ${User}@${Server}::$SRC $DEST/$(date +%Y-%m-%d)

## Rsync+Inotify实现文件自动同步

前言：

如果仅使用Rsync进行数据同步，只能满足企业对数据实时性要求不高的环境，即使使用计划任务也仅可以实现定期的数据同步。而且使用Rsync在进行数据同步前需要对所有的文件进行对比，然后进行差异数据同步，然而在大数据的情况下，同步数据的速度非常慢，效率低下。这时就需要结合Inotify（inode notify）工具。目前Inotify已经被集成到Linxu内核中，Inotify为用户的应用程序提供了文件系统事件告知机制。比如，当发生文件的访问、修改以及删除等事件时，可以立刻通过给用户的应用程序，通过Inotify可以实时了解文件系统发生的所有变化。

Inotify可以监控的部分常见文件系统事件如下：

事件名称 描述

IN\_ACCESS 文件访问事件

IN\_MODIFY 文件修改事件

IN\_ATTRIB 文件属性修改事件

IN\_OPEN 文件打开事件

IN\_CLOSE\_WRITE 可写文件被关闭事件

IN\_CLOSE\_NOWRITE 不可写文件被关闭事件

IN\_MOVED\_FROM 文件移动或重命名事件

IN\_MOVED\_TO 文件移动或重命名事件

IN\_DELETE 文件或目录删除事件

IN\_CREATE 文件或目录创建事件

IN\_DELETE\_SELF 自删除事件

利用Inotify的这种事件通知机制，用户态的应用程序就可以实时监控文件系统的变化，然而Inotify仅是内核提供的一种系统功能，用户如果需要使用该功能，还需要安装用户态软件，在CentOS 7 系统中可以使用inotify-tools 来实现文件系统的实时监控，该软件可以从https://github.com/rvoicilas/inotify-tools 中下载。

1，源码安装软件

去官网下载源码包安装部署该软件，下载源码文件解压后，可以查看对应的INSTALL安装说明文档，根据新版本的文档说明，需要使用 autogen.sh 脚本读取 configure.ac文件来创建configure安装脚本。而此步骤需要依赖于automake 以及 libtool软件包，如果采用的是yum方式安装这凉饿依赖软件包，系统会将编译inotify-tool时所需要的gcc软件也一起安装。

[root@centos ~]# wget <https://github.com/rvoicilas/inotify-tools/archive/v3.14.tar.gz>

[root@centos ~]# tar -zxvf inotify-3.14.tar.gz -C /usr/src/

[root@centos ~]# cd /usr/src/inotify-3.14

[root@centos ~]# yum -y install automake libtool

[root@centos ~]# ./configure

[root@centos ~]# make && make install

2，监控数据

inotify-tool提供了两个应用程序，分别为inotifywait 与 inotifywatch。其中inoyify的命令和描述如下：

描述：使用inotify机制等待文件系统事件，该命令非常适合实时监控文件系统的变化。

用法：inotifywait [-hcmrq] [-e <event>] [-t <seconds>] [--format <fmt>] [--timefmt <fmt>] <file> ...

选项：

-h 显示帮助信息

@<file> 指定监控路径中的例外文件，应用于不需要监控文件

--fromfile <file> 从文件中读取需要于例外的文件名称，每行一个文件名，

如果文件名称以@揩油，则表示例外文件

-m 接收到事件后不退出，默认程序在接收一个事件信息后会

退出

-d 与-m类似，但程序会进入后台执行，需要通过--outfile指定

事件信息的输出文件

-o <file> 将事件信息输出至文件，默认输出至标准输出

-s 将错误信息输出至syslog系统日志，默认输出至标准错误

输出

-r 递归监控

-q 静默模式，不输出信息

--exclude <pattern> 使用正则表达式匹配例外文件，区分大小写

--excludei <pattern> 使用正则表达式匹配例外文件,不区分大小写

-t <seconds> 在指定的时间没有发生事件，则退出程序

-e <event> 仅监控指定的事件

-c 使用CSV格式输出

--timefmt <fmt> 设置事件格式，即--format指定的%T格式

--format <fmt> 指定输出信息格式，具体格式参考man手册

实例一：

创建测试目录/test和测试文件/etc/foo，运行inotifywait命令监控/test目录，然后开启一个终端窗口运行命令 cat /test/foo,验证当发生查看文件事件时，是否会有事件通知。

[root@centos ~]# mkdir -p /test; echo "hello" > /test/foo

[root@centos ~]# inotifywait /test/

Setting up watches

Watches established.

/test/ OPEN foo

实例二：

编写一个脚本实时监控NetworkManager相关日志信息。

[root@centos ~]# cat monitor.sh

#!/bin/bash

while inotifywait -e modify /var/log/message

do

if tail -n1 /var/log/message | grep NetworkManager

then

echo Love

fi

done