Svn备份

1.在根目录data下新建备份文件backup,备份日志log,备份脚本script目录

2.复制svn备份脚本文件到新服务器root目录下

scp svnbackup.sh 192.168.10.22 : /data/script/

3.修改/etc/fstab配置文件，在最后一行添加：

//192.168.10.210/filebackup/websvn  /data/backup cifs  credentials=/root/.smbcredentials,iocharset=utf8 0 0

4.复制root下的.smbcredentials文件到新服务器root目录下：

scp .smbcredentials 192.168.10.22:/root/

5.编辑crontab

crontab -e

在最后一行下面添加两条语句：

0 23 \*\*6  /data/script/svnbackup.sh full /data/svn /data/backup>> /data/log/svnbackup.log

#每周六23点全量备份

0 4   \*\*\* /data/script/svnbackup.sh inc /data/svn /data/backup>> /data/log/svnbackup.log

#每天4点增量备份

6.添加cifs-utils包

apt-get install cifs-utils

7.reboot重启

8.df -h验证共享磁盘是否挂上

/data/backup的挂载点是否为系统文件

//192.168.10.210/filebackup/websvn

注：websvn目录要修改为对应的需要备份的文件。

其中，备份脚本文件svnbackup.sh内容如下：

#!/bin/bash //#!是特殊表示符，后面是脚本的shell的路径。

// 条件语句 if……then……fi

if [ $# != 3 ]; then //$#传给脚本或函数的参数个数，即位置变量数减1，不含脚本名称。

echo "usage: svn-backup ['full'|'ins'] [svn root] [backup dir root]" //echo是显示信息

echo "example 1: svn-backup full /opt/svn-repos /opt/svn-backup"

exit

fi

mode=$1 //定义备份类型为参数1

svn\_root=$2 //定义svn路径为参数2

backup\_root=$3 //定义svn备份路径为参数3

current\_date=`date +"%Y%m%d"` //定义当前日期，日期格式为四位年两位月两位日。

// for语句 for……in……do……done

for project in `ls -F $svn\_root | grep /$ | sed 's/\///g'`

// ls –F 显示子目录时按类型归类，并在末尾加一个“/“。grep /$ 搜索以/结尾的行，也就是标识目录。sed’s/\///g’ 替换行中所有的“\”为”/”。Sed’s/字符a 字符b/ 语法是把字符a替换为字符b。/g global可选标志，替换将对所有匹配的串进行。

do

project\_dir="$svn\_root/$project" //定义项目路径

backup\_dir="$backup\_root/$project" //定义备份路径

backup\_tmp\_dir="$backup\_root/$project/tmp" //定义备份临时路径

backup\_info="$project\_dir/backup\_info" //定义版本信息

last\_version=`svnlook youngest $project\_dir` //定义svn最新版本

//svnlook 查看最新的版本或Head. Svnlook youngest 不需选项，只显示Head版本号。

echo "`date +"%F %R:%S"` Clear old backup temp file"

// %F 等同于%Y%m%d。 %R 24小时制。%S 2位秒。

`rm -rf $backup\_tmp\_dir` //强制删除备份所有临时文件。-r 下一级。-f 强制。

`mkdir -p $backup\_tmp\_dir` //创建备份临时文件。 –p 可以是一个路径名称，一次可创建多个目录。

if [ ! -f $backup\_info ]; then //如果版本信息里没有时间。-f 显示文件中的每行时间。

echo 0 > $backup\_info

//把0输出到版本信息里。>是定向输出到文件，如果文件不存在，就创建文件；如果文件存在，就将其清空。

fi

last\_backup\_version=`cat $backup\_info` //定义最新备份版本。 Cat 显示的意思。

if [ $mode != 'full' ] && [ $last\_backup\_version == $last\_version ]; then

// && 用这种方法可以同时执行多条命令，当碰到执行出错的命令后将不执行后面的命令，如果一直没有出错则一直执行完所有命令；

echo "`date +"%F %R:%S"` '$project' nothing to do"

// 如果备份模式不为全备且最新备份版本与svn最新版本一致，显示不需要做操作。

else

if [ $mode == 'full' ] || [ $last\_backup\_version == '0' ] || [ ! -d $backup\_dir ]; then

// 如果备份模式为全备或者最新备份版本为0或者不存在备份路径，就执行全备。

// || 用这种方法可以同时执行多条命令，当碰到执行正确的命令后将不执行后面的命令，如果没有出现正确的命令则一直执行完所有命令；

// -d 存在目录。

target\_filename="svnbak\_${project}\_${current\_date}\_full.tar.bz2"

// 定义完全备份压缩包名，格式如：svnbak\_项目a\_20140404085620\_full.tar.bz2

echo "`date +"%F %R:%S"` Start full backup '$project'"

svnadmin hotcopy $project\_dir $backup\_tmp\_dir

// svnadmin 提供创建版本库功能，也可维护这些版本库。Hotcopy随时备份不管是否在使用。

echo "`date +"%F %R:%S"` Archiveing backup file '${target\_filename}'"

//archiving 存档，也即备份。

cd ${backup\_tmp\_dir}

tar jcf - \* | openssl des3 -salt -k JkeF8vx | dd of=../${target\_filename}

//压缩备份文件。j bzi2压缩方式。c 创建。f 文件。- 输出到标准输出。

last\_version=`svnlook youngest $backup\_tmp\_dir

rm -Rf $backup\_tmp\_dir` //删除备份临时文件

else //否则进行增量备份

target\_filename="svnbak\_${project}\_${current\_date}\_inc.tar.bz2"

//定义增量备份压缩包名，格式如：svnbak\_项目a\_20140404085620\_inc.tar.bz2 backup\_file="svnbak\_${project}\_${current\_date}\_inc\_${last\_backup\_version}\_${last\_version}.bak"

// 定义增量备份文件名为 svnbak\_项目a\_20140404085620\_inc\_最新备份版本号\_最新版本号.bak

cd ${backup\_tmp\_dir}

echo "`date +"%F %R:%S"` Start incremenal backup '$project' version $last\_backup\_version to

$last\_version"

svnadmin dump --incremental -q -r $last\_backup\_version:$last\_version $project\_dir > $backup\_file

// svnadmin dump 将文件内容转储到标准输出。Dump 导出版本库修订一定范围内的内容，使用可移植转储格式。--incremental 在一个转储文件中增加新的转储修订版本。Dump –incremenal 增量转储。-q 不弹出提示窗口。-r 开启重建文件系统。在空文件系统上操作，并将它还原为已保存的状态。

echo "`date +"%F %R:%S"` Archiveing backup file '${target\_filename}'"

tar jcf $backup\_file | openssl des3 -salt -k JkeF8vx | dd of=../${target\_filename}

fi

echo $last\_version > $backup\_info //把最新版本信息输出到版本信息里。

fi

done

echo "`date +"%F %R:%S"` done"

echo ""