

NSD ADMIN DAY05

1. [案例1：配置用户和组账号](#)
2. [案例2：创建一个备份包](#)
3. [案例3：配置NTP网络时间客户端](#)
4. [案例4：配置一个cron任务](#)

1 案例1：配置用户和组账号

1.1 问题

本例要求创建下列用户、组以及组的成员关系：

1. 新建用户 alex，其用户ID为3456，密码是flectrag
2. 创建一个名为 adminuser 的组
3. 创建一个名为 natasha 的用户，其属于 adminuser 组，这个组是该用户的从属组
4. 创建一个名为 harry 的用户，其属于 adminuser 组，这个组是该用户的从属组
5. 创建一个名为 sarah 的用户，其在系统中没有可交互的 Shell，并且不是 adminuser 组的成员
6. natasha、harry、sarah 的密码都要设置为 flectrag

1.2 步骤

实现此案例需要按照如下步骤进行。

步骤一：创建组账号

```
01 [root@server0 ~]# groupadd adminuser
```

步骤二：按照要求的属性创建用户账号

```
01 [root@server0 ~]# useradd -u 3456 alex
02.
03. [root@server0 ~]# useradd -G adminuser natasha
04. [root@server0 ~]# useradd -G adminuser harry
05. [root@server0 ~]# useradd -s /sbin/nologin sarah
```

步骤三：为用户设置登录密码

```
01 [root@server0 ~]# echo flectrag | passwd --stdin alex
02. 更改用户 alex 的密码。
03. passwd：所有的身份验证令牌已经成功更新。
04.
```

[Top](#)

05. `[root@server0 ~]# echo flectrag | passwd --stdin natasha`
06. 更改用户 natasha 的密码。
07. passwd：所有的身份验证令牌已经成功更新。
08. `[root@server0 ~]# echo flectrag | passwd --stdin harry`
09. 更改用户 harry 的密码。
10. passwd：所有的身份验证令牌已经成功更新。
11. `[root@server0 ~]# echo flectrag | passwd --stdin sarah`
12. 更改用户 sarah 的密码。
13. passwd：所有的身份验证令牌已经成功更新。

2 案例2：创建一个备份包

2.1 问题

本例要求使用 tar 工具完成以下备份任务：

1. 创建一个名为 /root/backup.tar.bz2 的归档文件
2. 其中包含 /usr/local 目录中的内容
3. tar 归档必须使用 bzip2 进行压缩

2.2 方案

制作归档压缩包：

- `tar -zPcf 备份文件.tar.gz 文档....`
- `tar -jPcf 备份文件.tar.bz2 文档....`
- `tar -JPcf 备份文件.tar.xz 文档....`

查看归档压缩包：

- `tar -tf 备份文件`

释放归档压缩包：

- `tar -xf 备份文件 [-C 目标目录]`

2.3 步骤

实现此案例需要按照如下步骤进行。

步骤一：创建备份文件

使用tar命令制作归档备份，结合-j选项调用bzip2压缩工具，保留绝对路径：

01. `[root@server0 ~]# tar jcPf /root/backup.tar.bz2 /usr/local/`

步骤二：确认结果

[Top](#)

```
01. [root@server0 ~]# ls -lh /root/backup.tar.bz2 //确认文件
02. -rw-r--r--. 1 root root 1.9K 12月 23 23:22 /root/backup.tar.bz2
03.
04. [root@server0 ~]# tar tPf /root/backup.tar.bz2 //确认内容
05. /usr/local/
06. /usr/local/bin/
07. /usr/local/bin/lab
08. /usr/local/etc/
09. /usr/local/games/
```

3 案例3：配置NTP网络时间客户端

3.1 问题

本例要求配置虚拟机 server0，能够自动校对系统时间。相关信息如下：

1. NTP服务器位于 classroom.example.com
2. 此客户机的系统时间应当与NTP服务器的时间保持同步

3.2 方案

NTP服务端可以为客户端提供标准的日期时间。

在RHEL7主机中可以配置软件包chrony来使用NTP时间同步。

3.3 步骤

实现此案例需要按照如下步骤进行。

步骤一：确认已安装NTP客户端软件包chrony

通常系统会默认安装此软件包：

```
01. [root@server0 ~]# rpm -q chrony
02. chrony-1.29.1-1.el7.x86_64
```

如果检查发现此软件包没有安装，请通过yum命令安装：

```
01. [root@server0 ~]# yum -y install chrony
02. ...
```

如果发现此软件包已经安装，但配置文件/etc/chrony.conf损坏或信息丢失，可删除此文件后重新安装chrony软件包：

[Top](#)

```

01. [root@server0 ~]# rm -rf /etc/chrony.conf
02. [root@server0 ~]# yum -y reinstall chrony
03. 已加载插件：langpacks
04. 正在解决依赖关系
05. --> 正在检查事务
06. ---> 软件包 chrony.x86_64.0.1.29.1-1.el7 将被 重新安装
07. --> 解决依赖关系完成
08. ...

```

步骤二：调整NTP客户端配置

1) 修改/etc/chrony.conf文件内的server配置

移除不可用的NTP服务器记录，正确添加可用的记录：

```

01. [root@server0 ~]# vim /etc/chrony.conf
02. ...
03. #server 0.rhel.pool.ntp.org iburst //注释掉默认的server配置，
04. #server 1.rhel.pool.ntp.org iburst
05. #server 2.rhel.pool.ntp.org iburst
06. #server 3.rhel.pool.ntp.org iburst
07. server classroom.example.com iburst //添加新的配置
08. ...

```

2) 开启NTP时间同步

```

01. [root@server0 ~]# timedatectl //查看现有状态
02. ...
03. NTP enabled: no //NTP可能尚未启用
04. NTP synchronized: no //尚未完成过一次NTP同步
05. ...
06. [root@server0 ~]# timedatectl set-ntp yes //启用NTP同步
07. [root@server0 ~]# timedatectl //查看启用后的状态
08. ...
09. NTP enabled: yes //NTP已经启用
10. NTP synchronized: no //尚未完成过一次NTP同步
11. ...

```

[Top](#)

步骤三：测试NTP时间同步

1) 先设置一个错误的系统日期时间

```

01. [root@server0 ~]# date -s '2001-09-11 11:30:00' //调整日期时间
02. 2001年 09月 11日 星期二 11:30:00 CST
03. [root@server0 ~]# date //确认调整结果
04. 2001年 09月 11日 星期二 11:30:01 CST

```

2) 启动系统服务chronyd，并设为开机自启

```

01. [root@server0 ~]# systemctl restart chronyd
02. [root@server0 ~]# systemctl enable chronyd

```

3) 重新查看当前的系统时间

重启chronyd服务后稍等片刻，当前系统的日期时间应该恢复正常（与NTP服务器保持一致）：

```

01. [root@server0 ~]# date
02. 2016年 12月 23日 星期五 23:44:53 CST

```

再次执行timedatectl查看，会发现NTP synchronized的值已经变成yes：

```

01. [root@server0 ~]# timedatectl
02. ...
03. NTP enabled: yes //NTP已经启用
04. NTP synchronized: no //已经完成过一次NTP同步
05. ...

```

4 案例4：配置一个cron任务

4.1 问题

本例要求为用户 natasha 配置一个定时任务，具体要求如下：

1. 每天在本地时间 14:23 执行
2. 需要完成的任务操作为 /bin/echo hiya

4.2 方案

配置格式可参考 /etc/crontab 文件：

[Top](#)

01 分 时 日 月 周 任务命令行 (绝对路径)

在表示各段的时间点时，除了明确的数值以外，还可以参考以下形式：

- *：匹配范围内任意时间
- ,：分隔多个不连续的时间点
- -：指定连续时间范围
- /n：指定时间频率，每n ...

4.3 步骤

实现此案例需要按照如下步骤进行。

步骤一：配置crontab任务记录

1) 确保系统服务crond可用

```
01. [root@server0 ~]# systemctl restart crond
02. [root@server0 ~]# systemctl enable crond
```

2) 为用户natasha添加计划任务

```
01. [root@server0 ~]# crontab -e -u natasha
02. 23 14 * * * /bin/echo hiya
```

步骤二：检查任务是否执行

1) 将系统日期时间临时调整到任务时间点前10秒左右

```
01. [root@server0 ~]# date -s '14:22:50' //设置
02. Sat Nov 26 14:22:50 CST 2016
03. [root@server0 ~]# date //确认日期时间
04. Sat Nov 26 14:22:55 CST 2016
```

2) 等待10秒后查看/var/log/cron日志，应该会有执行记录

```
01. [root@server0 ~]# tail /var/log/cron
02. ...
03. Nov 26 14:23:02 localhost CROND[3818]: (natasha) CMD (/bin/echo hiya)
```

[Top](#)

3) 恢复系统日期时间

01. [root@server0 ~] # hwclock -s //重设为系统时钟
02. [root@server0 ~] # date //确认日期时间
03. Sat Nov 26 05:05:23 CST 2016

[Top](#)