智能电网调度控制系统实操技术

王顺江　李　婷等　编著

ZHINENG DIANWANG
DIAODU KONGZHI XITONG
SHICAO JISHU

中国电力出版社
CHINA ELECTRIC POWER PRESS

智能电网
调度控制系统
实操技术

王顺江　李　婷等　编著

ZHINENG DIANWANG
DIAODU KONGZHI XITONG
SHICAO JISHU

中国电力出版社
CHINA ELECTRIC POWER PRESS

内 容 提 要

本书立足于目前智能电网调度控制系统最主流产品D5000系统，对系统的基本原理和基本功能做了简单介绍，对系统的各项操作做了详细阐述，满足主站自动化运维和管理人员的需求，本书共分10章，包括操作系统、数据库、基础平台、稳态监控、数据采集与交换、网络分析、电网动态运行监视系统、自动电压控制、自动发电控制系统、综合智能告警，涵盖了省级和地市级主站自动化人员运维和管理的主要内容。本书适合变电站自动化专业人员阅读，也可供变电站自动化相关院校师生学习参考使用。

图书在版编目（CIP）数据

智能电网调度控制系统实操技术/王顺江等编著. —北京：中国电力出版社，2018.10

ISBN 978-7-5198-2192-0

Ⅰ. ①智… Ⅱ. ①王… Ⅲ. ①智能控制－电网－电力系统调度 Ⅳ. ①TM76

中国版本图书馆CIP数据核字（2018）第145257号

出版发行：中国电力出版社

地　　址：北京市东城区北京站西街19号（邮政编码100005）

网　　址：http://www.cepp.sgcc.com.cn

责任编辑：孙　芳（010-63412381）

责任校对：黄　蓓　太兴华

装帧设计：赵姗姗

责任印制：蔺义舟

印　　刷：北京天宇星印刷厂

版　　次：2018年10月第一版

印　　次：2018年10月北京第一次印刷

开　　本：787毫米×1092毫米　16开本

印　　张：23.25

字　　数：512千字

印　　数：0001—1500册

定　　价：85.00元

编　委　会

王顺江　李　婷　苏安龙　石玉东　谭洪恩　董恩伏
钟丽波　沈　力　高　凯　王永福　朱宏超　南贵林
刘金波　葛维春　徐纯烈　孟祎南　施毅斌　赵　军
李　赫　南　阳　郑　璐　葛延峰　李　铁　钟奇桐
金　妍　张忠林　句荣滨　李劲松　金宜放　王　铎
凌兆伟　寿　增　许睿超　狄跃斌　刘嘉明　眭　冰
李正文　闫春生　赵德仁　王洪哲　殷鸿雁　于　游
张艳军　夏玉珏　高永俊　李典阳　罗桓桓　孙福斌
范新野　吴　昕　殷艳红　岳　伶　魏秀岩　张宏宇
丛海洋　张　建　崔　岱　姜　峰　王　刚　杨晓明
那广宇　韩子娇　李家珏　丛培贤　高　潇　黄　刚
马文全　路　明　李　森　王永清　张　琦　卫　亮
于　宁　池　洋　王忠波　潘　亮　刘　爽　唐宏丹
金　苗　纪　翔　孙　乔　周桂平　金　彪　孙畅岑
于　淼　宋　健　李　涵　李　朋

前 言

近年来，智能电网调度控制系统发展迅速，更新换代频繁，运维管理难度较大。而调度控制系统的重要性逐渐提升，与电网安全关联日趋紧密，若不能提升运维能力，必将导致系统可靠性下降，给电网安全稳定运行带来安全隐患。

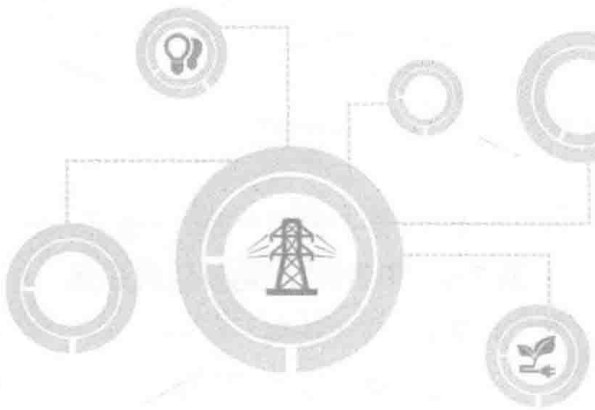
现阶段，智能电网调度控制系统主流产品是 D5000，市场占有率超过了 90%，但目前的书籍都注重于介绍理论和系统结构，没有直接介绍日常运维工作的操作方法，不利于智能电网调度控制系统运维人员的培养，因此本书基于主流智能电网调度控制系统 D5000，深入浅出地介绍系统实际操作技术，为目前主站自动化运维和管理提供有力支撑。

在编写组全体成员的共同努力下，历经 2 年多时间，经过初稿编写、轮换修改、集中会审、送审、定稿、校稿等多个阶段，终于完成了本书的编写和出版工作，感谢各位编写组成员的辛勤付出。

本书适合变电站自动化专业人员阅读，希望各位读者通过阅读本书，提升变电站监控系统故障缺陷处理能力，为日常的故障缺陷处理工作带来帮助。本书编辑时间较短，若有错漏，请各位读者批评指正。

编 者

2018 年 10 月



目 录

第1章

操 作 系 统

1.1 概 述

麒麟（Kylin）操作系统由操作系统工程研究中心研制，秉承了国家十五 863 银河麒麟操作系统多项关键技术。系统具有高性能，高安全，高可用，广兼容，易使用等特点，适用于政府、国防、金融、电信、能源、教育等领域。

1.2 特 点

1.2.1 广兼容（软硬件）

与国内多家主流板卡生产商、系统集成商、独立软件开发商紧密合作，进行大量优化改进，确保了系统与主流硬件设备的高度兼容性；支持 X86、AMD64、EM64T 多种硬件平台，并针对多核、多处理器结构进行优化，使系统性能达到最优。系统集成了丰富的开源软件，包括 MySQL、Apache 服务器、FTP 工具、多媒体播放工具、中间件、邮件服务系统，办公系统等。同时，系统还对 Oracle、Sybase、WebLogic、WebSphere 等主流商用软件提供良好的支持，为用户构建了一个功能强大的应用支撑环境。

1.2.2 高安全

强化的内核安全机制能够有效地防止病毒或黑客等的侵袭；系统采用内核与密码机制相融合的整体设计技术，增强的用户身份鉴别技术，细粒度的自主访问控制技术，这些涉及了操作系统和应用的一体化安全体系，并采用加密文件系统、角色定权等特色安全机制技术，使系统的安全性得到巩固。

1.2.3 高可靠

系统将内核可靠性和核外优化配置相结合，重点针对系统级容错、高可靠的 I/O 存储、检查点和集群高可用支持进行了详细设计，支持文件系统的日志功能和元数据保护功能，支持设备热插拔功能，同时也支持双机热备份功能。

1.2.4 网络化

面向新一代网络环境下应用的需求，提供对各类基于网络环境的服务的内核级支持和优化，对 StarBus、J2EE 和 WebLogic 等中间件提供强大支持。

1.2.5 简单易用

系统安装快捷，使用方便，配置简易，增强了对即插即用硬件设备的支持；网络管理器能够为有线和无线的连接进行自动配置；提供了基于 yum 的软件包安装、升级机制；具有类 Windows 的桌面风格，实现了多种用户友好的易用性设计， 用户可轻松地网上冲浪、收发邮件、撰写文档、即时聊天、欣赏电影。

1.3 常用操作命令

麒麟操作系统有着良好的图形化操作界面，但是对于经常使用麒麟系统的人来说，掌握常用的操作命令可以明显地提高工作效率，下面就通过一些实例来介绍一些日常操作中常用操作命令用法。

（1）查找本机 process.ini 文件所在位置。

答案：locate process.ini

详解：查找文件通常使用 find 命令，但是该命令是对目录进行搜索，搜索速度较慢，locate 命令则要比 find 命令快得多，locate 不搜索具体目录，只搜索数据库“/var/lib/locatedb”，该数据库包含本地文件信息，但是该库是每天更新一次，所以 locate 无法查到新变动过的文件。所以如果想要用 locate 命令来查找文件，可以提前执行 updatedb 命令来手动更新数据库，再用 locate 进行查找，如图 1-1 所示。

```
[root@sg20-sca1 test]# locate process.ini
[root@sg20-sca1 test]# updatedb
[root@sg20-sca1 test]# locate process.ini
/home/d5000/gwxy/test/process.ini
```

图 1-1 查找文件所在位置

（2）查看 ceshi 路径下有哪些文件的内容中包括“host”字样。

答案：grep host *

详解：如图 1-2 所示，要搜索包含某种内容的文件都有哪些，通常使用 grep 命令，该命令后接两个参数，第一个参数是要检索的内容，第二个参数是检索的范围。答案中的*表示模糊匹配，即当前路径下的所有文件，所以答案所示命令的意思就是在当前路径的所有文件中搜索内容含有“host”的文件并显示出来，结果中从左到右会显示包含“host”字样的文件名，“host”所在行数和“host”所在行内容。

```
//KS1 sg20-sca1:/home/d5000/gwxy/test % grep host *
mulu:4:hostname=localname
name:1:hosts1=one
name:2:hosts2=two
name:3:hosts3=three
```

图 1-2 grep 命令用法

（3）比较 mng_proc.ini 和 mng_proc.txt 文件内容有什么不同。

答案：vim-d mng_proc.ini mng_proc.txt

详解：要比较两个文件有哪些内容是不同的可以用 vim-d 命令，该命令后接想要对比的两个文件的名称即可，如图 1-3 所示。

图 1-3 vim-d 命令用法

该命令会对两个文件内容进行对比，并将不同的地方做出明显标示。从结果中可以看出 mng_proc.txt 文件相比 mng_proc.ini 缺少第一行的内容，并且在最后一行两个文件的内容也不一样。

（4）将文件 name.txt 文件中的“a”批量修改为“3”。

答案：如图 1-4 所示，用 vi 编辑器打开 name.txt 文件

```
//KS1 sg20-sca1:/home/d5000/gwxy/test % vi name.txt
```

图 1-4 vi 编辑器

然后如图 1-5 打开后输入：%s#a#3#g。

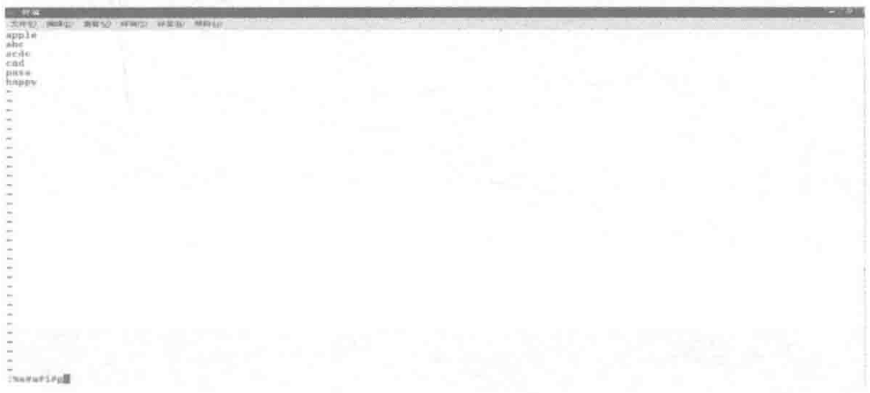


图 1-5 批量修改文件（一）

回车执行就可以看到修改成功了，如图 1-6 所示。

（5）修改文件名以“windows”开头的文件，将文件名称中的“windows”批量替换为“linux”。

答案：rename windows linux windows*

详解：当前路径下有多个以 windows 开头的文件，如图 1-7 所示

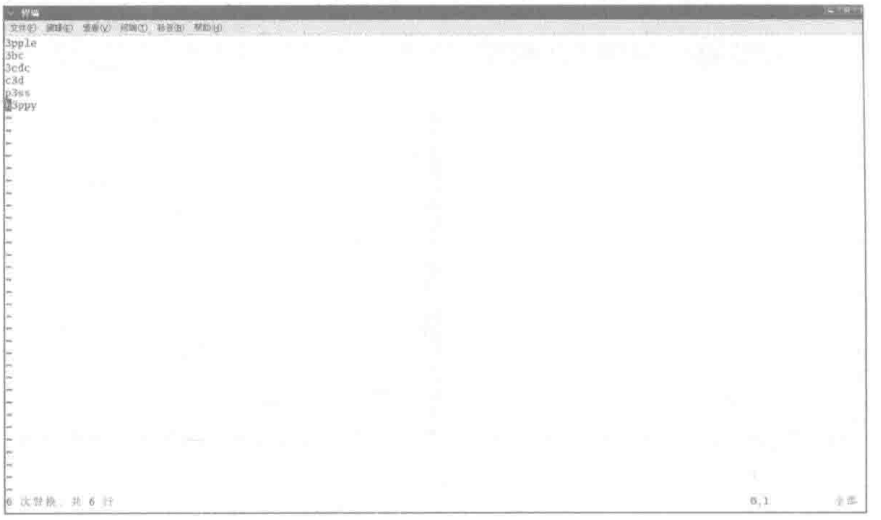


图 1-6　批量修改文件（二）

图 1-7　ls 命令查看当前目录下文件

批量修改文件名称一般使用 rename 命令，答案中的 rename 后接三个参数，第一个参数为文件名中想要修改的内容，第二个参数为文件名修改后内容，第三个参数表示想要对其进行修改的文件。修改后结果如图 1-8 所示。

图 1-8　rename 命令用法

（6）统计文件 abc 的行数、字数、字节数。

答案：wc abc

详解：如图 1-9 所示，wc 命令后面接所要统计的文件名称，可以统计出该文件的行数、字数、字节数和文件名，输出结果显示文件 abc 的行数为 3 行，字数为 3，字节数为 29，文件名为 abc。

```
//KS1 sg20-sca1:/home/d5000/gwxy/test % wc abc
 3  3 29 abc
```

图 1-9　wc 命令用法

（7）将 conf 文件夹进行打包，压缩包名称为 test.tar.gz。

答案：tar-czvf test.tar.gz conf

详解：在 linux 操作系统中，常用 tar 命令来压缩文件，答案中 tar 后接了多个参数，其中：

–c 表示建立压缩档案

–z 表示有 zip 属性

–v 表示显示所有过程

–f 表示使用档案名字，这里注意 f 参数一定要放到最后，后面接压缩后的文件名称，即题目中的 test.tar.gz

执行命令前还要注意被压缩文件和压缩后文件名的先后顺序，被压缩文件名一定要放在最后，如图 1-10 所示。

```
//KS1 sg20-sca1:/home/d5000/gwxy/test % tar -czvf test.tar.gz conf
conf/
conf/3.txt
conf/2.txt
conf/1.txt
```

图 1-10　tar 命令用法

压缩执行完成后在该路径下就会生成一个最新的压缩文件，即 test.tar.gz，如图 1-11 所示。

```
//KS1 sg20-sca1:/home/d5000/gwxy/test % ls -tlr
总计 116
-rw-rw-r-- 1 d5000 d5000    0 12-27 15:52 linux_20171227.log
-rw-rw-r-- 1 d5000 d5000    0 12-28 00:00 linux_20171228.log
-rw-rw-r-- 1 d5000 d5000    0 01-15 00:00 linux_20180115.log
-rw-rw-r-- 1 d5000 d5000    0 01-16 00:00 linux_20180116.log
-rw-rw-r-- 1 d5000 d5000    0 01-17 00:00 linux_20180117.log
-rw-rw-r-- 1 d5000 d5000    0 01-18 00:00 linux_20180118.log
-rw-rw-r-- 1 d5000 d5000    0 01-19 00:00 linux_20180119.log
-rw-rw-r-- 1 d5000 d5000    0 01-20 00:00 linux_20180120.log
-rw-rw-r-- 1 d5000 d5000    0 01-21 00:00 linux_20180121.log
-rw-rw-r-- 1 d5000 d5000    0 01-22 00:00 linux_20180122.log
-rw-rw-r-- 1 d5000 d5000    0 01-23 00:00 linux_20180123.log
-rw-rw-r-- 1 d5000 d5000    0 01-24 00:00 linux_20180124.log
-rw-r--r-- 1 root  root     0 01-28 16:58 process.ini
-rw-rw-r-- 1 d5000 d5000   73 01-28 17:17 mulu
-rw-rw-r-- 1 d5000 d5000   55 01-28 17:59 mng_proc.ini
-rw-rw-r-- 1 d5000 d5000   50 01-28 18:01 mng_proc.txt
-rw-rw-r-- 1 d5000 d5000   30 01-28 18:23 name.txt
-rw-rw-r-- 1 d5000 d5000   35 01-28 18:42 name
-rw-rw-r-- 1 d5000 d5000   29 01-28 18:45 abc
drwxrwxr-x 2 d5000 d5000 4096 01-28 18:51 conf
-rw-rw-r-- 1 d5000 d5000  158 01-28 18:53 test.tar.gz
```

图 1-11　压缩结果

（8）将 break.log 文件转换为隐藏文件。

答案：mv break.log .break.log

详解：linux 系统中文件和文件夹是通过“.”来进行隐藏的，以“.”开头的文件被认为是隐藏文件，通过 ls 命令无法看到，如果想要看到隐藏文件，则需要用 ls-a 命令来查看。如图 1-12 所示，当前路径下有 break.log 文件，此时通过 ls 命令可以看到该文件。

```
//KS1 sg20-sca1:/home/d5000/gwxy/test % ls
abc         linux_20171227.log  linux_20180116.log  linux_20180119.log  linux_20180122.log  mng_proc.ini  name       test.tar.gz
break.log   linux_20171228.log  linux_20180117.log  linux_20180120.log  linux_20180123.log  mng_proc.txt  name.txt
conf        linux_20180115.log  linux_20180118.log  linux_20180121.log  linux_20180124.log  mulu          process.ini
```

图 1-12　ls 命令查看文件

当执行完答案所示命令后，再执行 ls 命令后无法查看到该文件，如图 1-13 所示。

图 1-13　再次执行 ls 命令

只有执行 ls-a 命令才能看到该隐藏文件，如图 1-14 所示。

图 1-14　ls-a 命令查看隐藏文件

（9）修改 luser 用户密码，将密码修改为 abcd.2008。

答案：passwd luser，根据提示输入修改后的密码

详解：linux 系统中如果想要修改用户密码通常使用 passwd 命令，一般使用 root 用户执行该命令，passwd 命令后接想要修改密码的用户名称即可，如图 1-15 所示，该命令执行后会要求输入新密码，并确认新密码，执行完上述操作后系统会提示密码修改成功。

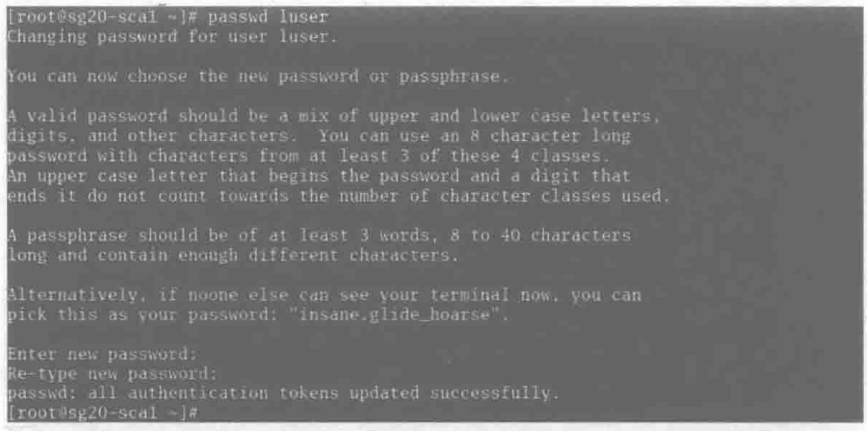


图 1-15　passwd 命令用法

（10）编写定时任务，将/home/d5000/gwxy 磁盘空间使用情况在每天的 09：00 写入到 kong.txt 中，要求该文件只显示最后一次的执行结果。

答案：crontab-e 编写定时任务，内容为：00 9 * * * df-h >kong.txt

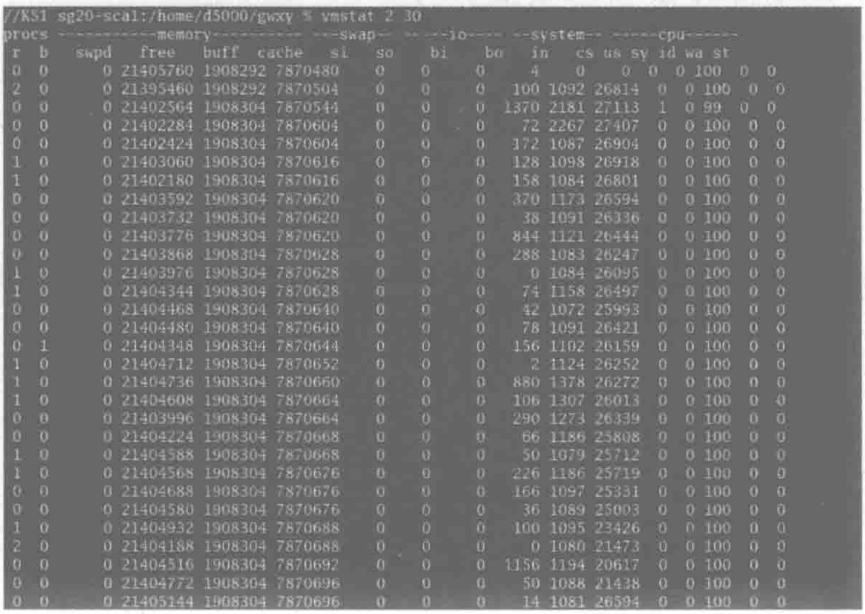
详解：linux 系统中添加定时任务的命令是 crontab-e，执行完该命令会显示任务编辑界面，在该界面根据规定格式添加任务。格式如图 1-16 所示，包含六个参数，前五个参数为需要执行该任务的时间点，从左到右分别对应分（0～59）、时（0～23）、日（1～31）、月（1～12）、星期（1～7），最后一个参数是要执行的命令或者脚本。将命令打印的内容写入对应文件可以用“>”和“>>”，“>”表示覆盖写入，“>>”表示追加写入，这里需要注意的是题中要求只显示最后一次结果，所以用 df-h>kong.txt。

图 1-16 crontab-e 编写定时任务

（11）利用 vmstat 查看系统资源使用情况，以 2 秒钟为间隔，将持续 1 分钟的资源使用情况显示出来。

答案：vmstat 2 30

详解：vmstat 命令可对操作系统的虚拟内存、进程、IO 读写、CPU 活动等进行监视。根据题中要求，在 vmstat 后接两个参数，“2”表示每两秒统计一次信息，“30”表示共统计 30 次，执行后的结果即是一分钟内的统计信息详情，如图 1-17 所示。



```
//KS1 sg20-sca1:/home/d5000/gwxy % vmstat 2 30
procs -----------memory---------- ---swap-- -----io---- --system-- -----cpu------
 r  b   swpd   free   buff  cache   si   so    bi    bo   in   cs us sy id wa st
 0  0      0 21405760 1908292 7870480    0    0     0     4    0    0  0  0 100  0  0
 2  0      0 21395460 1908292 7870504    0    0     0     0  100 1092 26814  0  0 100  0  0
 0  0      0 21402564 1908304 7870544    0    0     0     0 1370 2181 27113  1  0 99  0  0
 0  0      0 21402284 1908304 7870604    0    0     0     0   72 2267 27407  0  0 100  0  0
 0  0      0 21402424 1908304 7870604    0    0     0     0  372 1087 26904  0  0 100  0  0
 1  0      0 21403060 1908304 7870616    0    0     0     0  128 1098 26918  0  0 100  0  0
 1  0      0 21402180 1908304 7870616    0    0     0     0  158 1084 26801  0  0 100  0  0
 0  0      0 21403592 1908304 7870620    0    0     0     0  370 1173 26594  0  0 100  0  0
 0  0      0 21403732 1908304 7870620    0    0     0     0   38 1091 26336  0  0 100  0  0
 0  0      0 21403776 1908304 7870620    0    0     0     0  844 1121 26444  0  0 100  0  0
 0  0      0 21403868 1908304 7870628    0    0     0     0  288 1083 26247  0  0 100  0  0
 1  0      0 21403976 1908304 7870628    0    0     0     0    0 1084 26095  0  0 100  0  0
 1  0      0 21404344 1908304 7870628    0    0     0     0   74 1158 26497  0  0 100  0  0
 0  0      0 21404468 1908304 7870640    0    0     0     0   42 1072 25993  0  0 100  0  0
 0  0      0 21404480 1908304 7870640    0    0     0     0   78 1091 26421  0  0 100  0  0
 0  1      0 21404348 1908304 7870644    0    0     0     0  156 1102 26159  0  0 100  0  0
 1  0      0 21404712 1908304 7870652    0    0     0     0    2 1124 26252  0  0 100  0  0
 1  0      0 21404736 1908304 7870660    0    0     0     0  880 1378 26272  0  0 100  0  0
 1  0      0 21404608 1908304 7870664    0    0     0     0  106 1307 26013  0  0 100  0  0
 0  0      0 21403996 1908304 7870664    0    0     0     0  290 1273 26339  0  0 100  0  0
 0  0      0 21404224 1908304 7870668    0    0     0     0   66 1186 25808  0  0 100  0  0
 1  0      0 21404588 1908304 7870668    0    0     0     0   50 1079 25712  0  0 100  0  0
 1  0      0 21404568 1908304 7870676    0    0     0     0  226 1186 25719  0  0 100  0  0
 0  0      0 21404688 1908304 7870676    0    0     0     0  166 1097 25331  0  0 100  0  0
 0  0      0 21404580 1908304 7870676    0    0     0     0   36 1089 25003  0  0 100  0  0
 1  0      0 21404932 1908304 7870688    0    0     0     0  100 1095 23426  0  0 100  0  0
 2  0      0 21404188 1908304 7870688    0    0     0     0    0 1080 21473  0  0 100  0  0
 0  0      0 21404516 1908304 7870692    0    0     0     0 1156 1194 20617  0  0 100  0  0
 0  0      0 21404772 1908304 7870696    0    0     0     0   50 1088 21438  0  0 100  0  0
 0  0      0 21405144 1908304 7870696    0    0     0     0   14 1081 26594  0  0 100  0  0
```

图 1-17 vmstat 2 30 命令用法

（12）在 messages 文件的第 13 行的句首添加“linux”。

答案：vi 编辑器打开 messages 文件，输入 13g 跳到 13 行，输入“i”进入编辑模式进行编辑

详解：输入 vi messages 打开 messages 文件，如图 1-18 所示。

```
//KS1 sg20-sca1:/home/d5000/gwxy/test % vi messages
```

图 1-18 vi 编辑器打开文件

打开后该文件内容如图 1-19 所示。

如果想要标注行数可以如图 1-20 输入：set nu，回车后会在每行的行首显示当前行数，如图 1-21 所示。

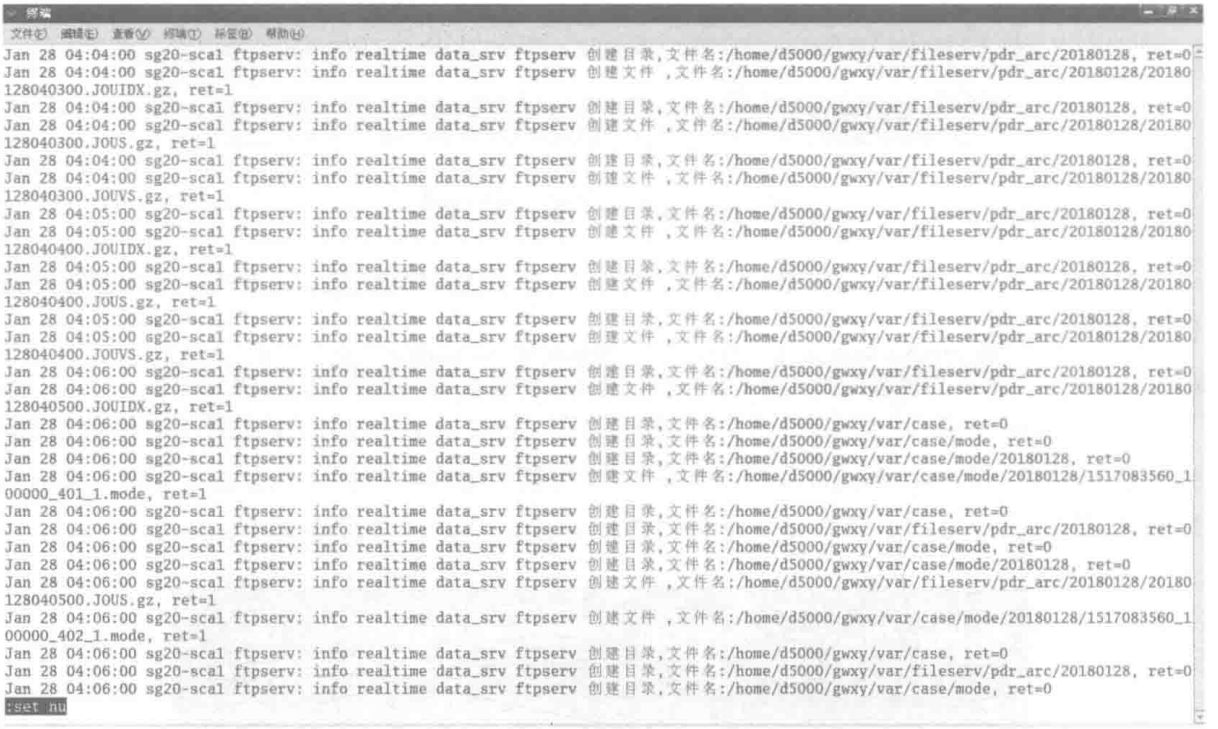


图 1-19　文件内容

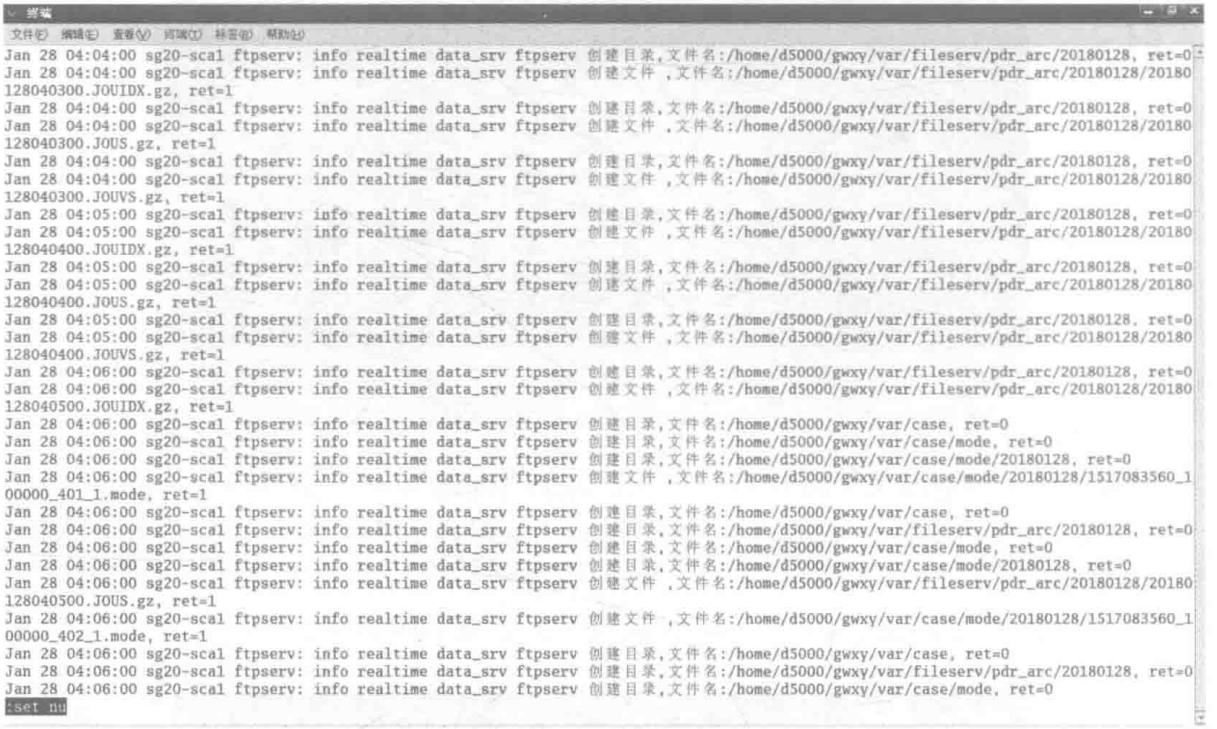


图 1-20　执行 set nu

如果想要跳到对应行可以输入 ng，其中 n 表示想要跳到的行，本题输入 13g，回车后光标回跳到 13 行，在行首输入“i“进入编辑模式，输入 linux 后按 esc 键退出编辑模式，如图 1-22 所示。

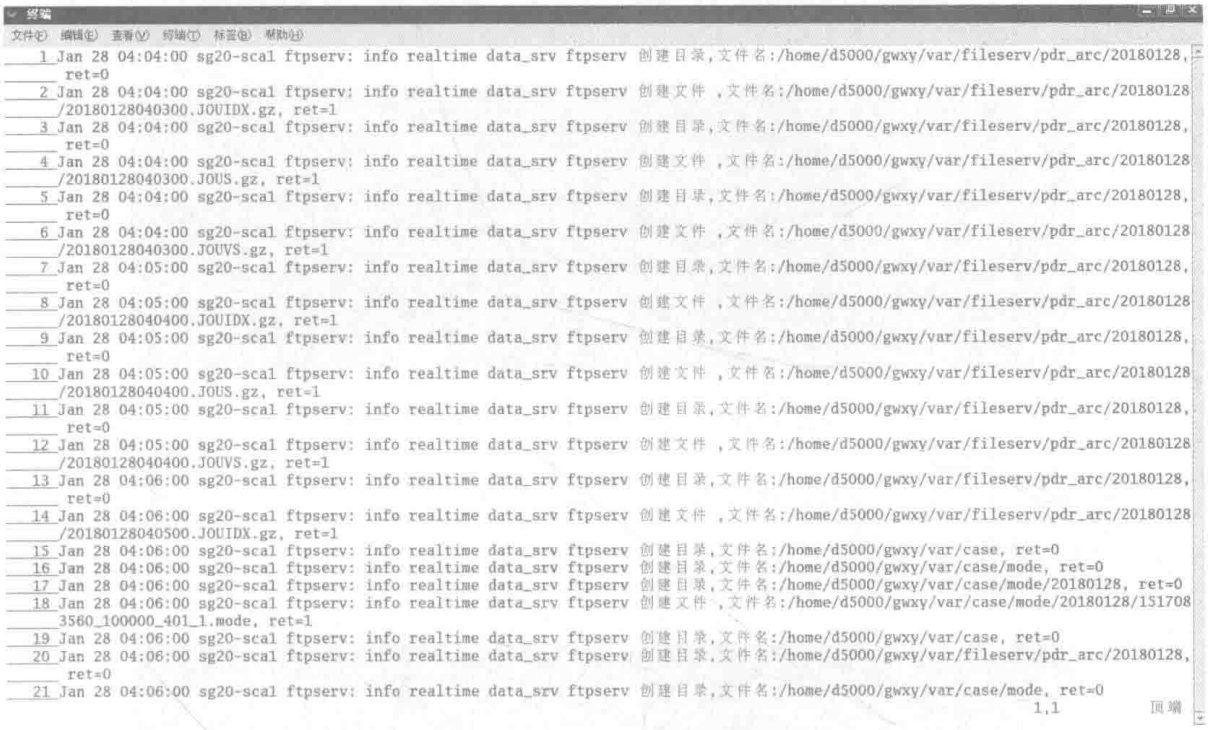


图 1-21 set nu 显示当前行数

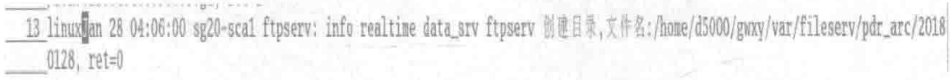


图 1-22 ngg 命令跳到对应行

然后输入 wq，再回车退出 vi 编辑器，则完成了题目所要求的修改，如图 1-23 所示。

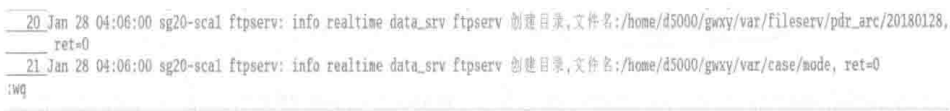


图 1-23 wq 命令退出 vi 编辑器

（13）搜索当前路径下大于 5M 并小于 10M 的文件。

答案：find. -size+5M-size-10M

详解：如图 1-24 所示，答案是 find 命令的一种用法，该命令中“.”表示当前目录，“+5M”表示大于 5M，“-10M”表示小于 10M，执行该命令即可检索出当前路径下符合条件的文件。

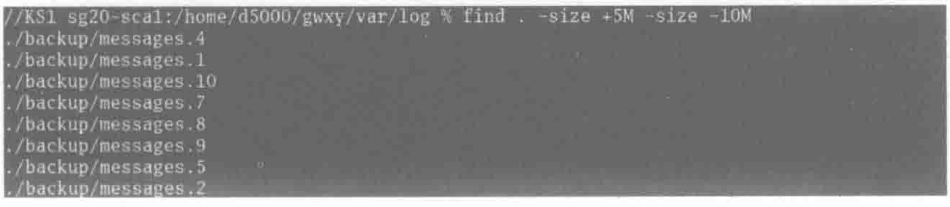


图 1-24 find 命令用法

如图 1-25 所示，可以用 du-sh 命令验证答案是否正确。

```
//KS1 sg20-sca1:/home/d5000/gwxy/var/log % du -sh ./backup/messages*
7.5M    ./backup/messages.1
7.9M    ./backup/messages.10
6.1M    ./backup/messages.2
1.1M    ./backup/messages.3
6.5M    ./backup/messages.4
5.7M    ./backup/messages.5
24K     ./backup/messages.6
6.5M    ./backup/messages.7
7.9M    ./backup/messages.8
7.9M    ./backup/messages.9
```

图 1-25　du-sh 命令验证答案正确性

（14）查看 kong.txt 的最后四行内容，并显示出来。

答案：tail-n 4 kong.txt

详解：如图 1-26 所示，kong.txt 文件内容如下所述。

```
//KS1 sg20-sca1:/home/d5000/gwxy/test % cat kong.txt
文件系统              容量  已用  可用  已用% 挂载点
/dev/sda1              48G   20G   26G  44% /
/dev/sda2             117G   46G   66G  41% /home/d5000/gwxy
/dev/sda3              76G  6.9G   65G  10% /home/d5000/gwxy/var
tmpfs                  16G   72K   16G   1% /dev/shm
文件系统              容量  已用  可用  已用% 挂载点
/dev/sda1              48G   20G   26G  44% /
/dev/sda2             117G   46G   66G  41% /home/d5000/gwxy
/dev/sda3              76G  6.9G   65G  10% /home/d5000/gwxy/var
tmpfs                  16G   72K   16G   1% /dev/shm
文件系统              容量  已用  可用  已用% 挂载点
/dev/sda1              48G   20G   26G  44% /
/dev/sda2             117G   46G   66G  41% /home/d5000/gwxy
/dev/sda3              76G  6.9G   65G  10% /home/d5000/gwxy/var
tmpfs                  16G   72K   16G   1% /dev/shm
```

图 1-26　cat 命令显示文件内容

当执行答案所示命令后，即可显示题中所要求的内容，如图 1-27 所示。

```
//KS1 sg20-sca1:/home/d5000/gwxy/test % tail -n 4 kong.txt
/dev/sda1              48G   20G   26G  44% /
/dev/sda2             117G   46G   66G  41% /home/d5000/gwxy
/dev/sda3              76G  6.9G   65G  10% /home/d5000/gwxy/var
tmpfs                  16G   72K   16G   1% /dev/shm
```

图 1-27　tail 命令用法