

Solrex Linux Cheatsheet

© 2009 — Solrex Yang — License CC [by nc sa]

目录

环境变量.....1	进程管理.....2
修改环境变量.....1	系统服务控制.....2
基本命令.....1	声音控制.....2
查看帮助.....1	Bash Shortcuts.....2
目录操作.....1	VIM.....2
文件操作.....1	Emacs.....2
文件搜索.....1	SED & AWK.....2
输入、输出和重定向.....1	SED.....2
压缩和解压.....1	AWK.....3
网络控制.....1	LFTP.....3
网络管理.....1	SSH.....3
网络应用.....2	GCC.....3
软件管理.....2	GDB.....3
文本操作.....2	
定时执行命令.....2	

环境变量

EDITOR	偏好的文本编辑器, 被 <code>svn</code> 等程序读取
HOME	当前用户的家目录
JAVA_HOME	Java 程序主目录
LANG	用户使用的语言, 被 <code>man</code> 等程序读取
LD_LIBRARY_PATH	动态链接库搜索路径, 用冒号分隔
PATH	可执行程序搜索路径, 用冒号分隔
PWD	当前目录
TERM	终端类型
USER	当前用户名
http_proxy	http 代理: <code>http://user:passwd@host:1080</code>
sock_proxy	sock 代理: 同上

修改环境变量

<code>export PATH=~/bin:\$PATH</code>	添加~/bin到环境变量PATH中
<code>export LANG=zh</code>	设置环境变量LANG为zh中文
<code>export</code>	显示所有 <code>export</code> 的环境变量
<code>unset http_proxy</code>	删除环境变量 <code>http_proxy</code>
<code>set</code>	显示所有 <code>shell</code> 变量和函数

基本命令

查看帮助

<code>man lftp</code>	查看 <code>lftp</code> 命令的手册, <code>q</code> 退出 (下以 <code>lftp</code> 为例)
<code>man -k lftp</code>	搜索匹配 <code>lftp</code> 命令的手册, 正则表达式
<code>lftp --help</code>	打印 <code>cmd</code> 的帮助信息, 适用于大多数命令
<code>info lftp</code>	查看 <code>lftp</code> 的 <code>info</code> 文档, 比 <code>man</code> 更详细

目录操作

<code>cd</code>	回到家目录
<code>cd -</code>	回到前一目录
<code>cd dir</code>	进入 <code>dir</code> 目录
<code>dirs -v</code>	列出当前的目录栈, 前面加上序号
<code>mkdir dir</code>	创建 <code>dir</code> 目录
<code>mkdir -p dir</code>	递归创建目录树
<code>popd +n</code>	删除目录栈第 <code>n</code> 项

1a

<code>pushd +n</code>	进入目录栈第 <code>n</code> 项, 并将该项重新压入栈顶
<code>pushd dir</code>	进入 <code>dir</code> 目录, 并将 <code>dir</code> 压入目录栈
<code>rm -rf dir</code>	强制删除 <code>dir</code> 目录及所有内容
<code>rmdir dir</code>	删除 <code>dir</code> 目录 (空)

文件操作

<code>chmod u+x file</code>	为 <code>file</code> 拥有者增加可执行权限, 另外 <code>u</code> 可替换为: <code>g</code> 组, <code>o</code> 其他人, <code>a</code> 所有人; <code>+</code> 可替换为 <code>-</code> 去除; <code>x</code> 可替换为 <code>r</code> 读, <code>w</code> 写, <code>X</code> 仅目录。
<code>chmod 0755 file</code>	设置 <code>file</code> 的权限位为 <code>755:wxr-xr-x</code>
<code>chgrp grp file</code>	改变 <code>file</code> 所属组为 <code>grp</code>
<code>chown user file</code>	改变 <code>file</code> 拥有者为 <code>user</code>
<code>diff file1 file2</code>	比较两个文件内容
<code>diff -Naur old new</code>	生成 <code>patch</code> 文件的标准选项
<code>file file</code>	查看文件类型
<code>rm file</code>	删除文件 <code>file</code>
<code>rm *.txt</code>	删除所有以 <code>.txt</code> 结尾的文件
<code>touch file</code>	更新 <code>file</code> 文件时间戳, 或创建空文件 <code>file</code>
<code>ln -s file link</code>	建立到 <code>file</code> 的链接 <code>link</code> , 操作数亦可为目录

文件搜索

<code>find dir -iname "*.c"</code>	查找 <code>dir</code> 目录及子目录中以 <code>.c</code> 结尾的文件或目录, 忽略文件名大小写
<code>find dir -type f</code>	查找 <code>dir</code> 目录及子目录中所有文件
<code>locate -b menu.lst</code>	列出文件名中包含 <code>menu.lst</code> 的文件
<code>ls</code>	列出当前目录下文件
<code>ls --color=auto</code>	用不同颜色高亮显示不同类型的文件
<code>ls -a</code>	列出所有文件
<code>ls -l</code>	列出文件的详细信息
<code>ls -t</code>	以时序列出文件
<code>ls -h</code>	使用用户友好的文件大小: <code>1K</code> , <code>2M</code> , <code>3G</code>
<code>ls dir</code>	列出 <code>dir</code> 目录下文件
<code>tree dir</code>	列出以 <code>dir</code> 为根的目录树

输入、输出和重定向

<code>0</code>	标准输入 <code>stdin</code> 的文件描述符
<code>1</code>	标准输出 <code>stdout</code> 的文件描述符
<code>2</code>	标准错误 <code>stderr</code> 的文件描述符
<code>cat file</code>	输出 <code>file</code> 内容
<code>cat file1 file2</code>	连续输出 <code>file1</code> 和 <code>file2</code>
<code>echo "hello"</code>	输出 <code>hello</code> 并换行
<code>echo -n "hello"</code>	输出 <code>hello</code> , 不换行
<code>echo -e "hello\tyou"</code>	开启反斜线转义字符识别
<code>echo \$PATH</code>	输出 <code>PATH</code> 环境变量内容
<code>read name</code>	读取输入并赋给 <code>name</code> 变量
<code>tee file</code>	将 <code>stdin</code> 同时复制到 <code>stdout</code> 和 <code>file</code>
<code> </code>	管道符, 用于连接两个程序。示例: 将 <code>ps</code> 结果通过管道传递给 <code>tee</code> , 然后输出到屏幕, 同时保存到 <code>file</code>
<code>ps -e tee file</code>	
<code>ls > fnile</code>	将 <code>ls</code> 的 <code>stdout</code> 重定向到 <code>file</code>
<code>ls >> file</code>	将 <code>ls</code> 的 <code>stdout</code> 重定向到 <code>file</code> (追加)
<code>ls &> file</code>	将 <code>ls</code> 的 <code>stdout</code> 和 <code>stderr</code> 重定向到 <code>file</code>
<code>ls > file 2>&1</code>	同上
<code>ls > file1 2>file2</code>	将 <code>ls</code> 的 <code>stdout</code> 重定向到 <code>file1</code> , <code>stderr</code> 重定向到 <code>file2</code>
<code>sort < file</code>	将 <code>file</code> 内容重定向到 <code>stdin</code> 进行排序
<code>sort < file1 > file2</code>	将排序结果重定向到 <code>file2</code>

1b

压缩和解压

<code>bzip2 -c file > file.bz2</code>	将 <code>file</code> 压缩为 <code>file.bz2</code>
<code>bzip2 -d file.bz2</code>	解压 <code>file.bz2</code>
<code>gzip -c file > file.gz</code>	将 <code>file</code> 压缩为 <code>file.gz</code>
<code>gzip -d file.gz</code>	解压 <code>file.gz</code>
<code>tar -cvf dir.tar dir</code>	将 <code>dir</code> 打包为 <code>dir.tar</code>
<code>tar -czvf dir.tar.gz dir</code>	将 <code>dir</code> 打包并用 <code>gzip</code> 压缩
<code>tar -cjvf dir.tar.bz2 dir</code>	将 <code>dir</code> 打包并用 <code>bzip2</code> 压缩
<code>tar -xvf dir.tar</code>	在当前目录下解开 <code>tar</code> 包
<code>tar -xzvf dir.tar.gz</code>	在当前目录下解开 <code>tar.gz</code> 包
<code>tar -xjvf dir.tar.bz2</code>	在当前目录下解开 <code>tar.bz2</code> 包
<code>zip -r dir.zip dir</code>	将 <code>dir</code> 打包并用 <code>zip</code> 压缩
<code>unzip dir.zip</code>	解压 <code>dir.zip</code>

网络控制

<code>dhclient wlan0</code>	在 <code>wlan0</code> 接口上使用 <code>dhcp</code> 自动获取 IP 和网络配置
<code>ifconfig eth0 192.168.0.2 netmask 255.255.255.0 up</code>	配置网络接口 <code>eth0</code> 的 IP 为 <code>192.168.0.2</code> , 子网掩码为 <code>255.255.255.0</code> , 并启动之
<code>ifconfig eth0 down</code>	关闭网络接口 <code>eth0</code>
<code>ifconfig eth0 dhcp</code>	配置网络接口 <code>eth0</code> 为 <code>dhcp</code>
<code>iwlist wlan0 scan</code>	在 <code>wlan0</code> 接口上扫描无线网络
<code>iwconfig wlan0 essid TPLINK-HOME key s:password</code>	配置 <code>wlan0</code> 接口连接到名为 <code>TPLINK-HOME</code> 的 WEP 无线网络, 使用 <code>ascii</code> 密钥 <code>password</code>
<code>route add default gw 192.168.0.1</code>	添加 <code>192.168.0.1</code> 为缺省的网关
<code>/etc/init.d/network[ing] start stop restart</code>	启动、停止、重启所有网络接口
<code>/etc/network/interfaces</code>	保存网络配置的文件
<code>/etc/resolv.conf</code>	保存 DNS 服务器地址列表的文件

/etc/network/interfaces 示例:

```
# The loopback network interface
auto lo
iface lo inet loopback
auto eth0
#iface eth0 inet dhcp
iface eth0 inet static
address 192.168.0.2
netmask 255.255.255.0
gateway 192.168.0.1
#auto wlan0
#iface wlan0 inet dhcp
#wireless-essid TPLINK-HOME
#wireless-key s:password
```

/etc/resolv.conf 示例:

```
# Generated by NetworkManager
nameserver 192.168.0.1
```

网络管理

<code>ifconfig</code>	显示接口, IP, MAC 等网络信息
<code>ping 192.168.0.1</code>	向某 IP 发送 ICMP 报文, 诊断连接
<code>ping6 ipv6.google.com</code>	向某 IPv6 发送 ICMP 报文
<code>tracpath baidu.com</code>	显示到某 v4 主机的路由路径
<code>tracpath6 ipv6.google.com</code>	显示到某 v6 主机的路由路径
<code>netstat -a</code>	显示所有网络连接信息

1c

netstat -tl	显示本机监听的 TCP 端口连接
host -a solrex.cn	查询 solrex.cn 域名对应 IP
nslookup solrex.cn	查询 solrex.cn 域名对应 IP
dig solrex.cn	同上,但更强大,因而更难用
nslookup solrex.cn server	向 DNS 服务器 server 查询域名
nslookup ip	通过 ip 反查域名
whois solrex.cn	查询 solrex.cn 的域名信息
lsof -i:22	查看监听端口 22 的进程,root 权限

网络应用	
wget url	下载 url 指向的网页或者文件
wget -c url	同上,但支持断点续传
curl url > file	下载 url,保存到 file
curl -O abc.cn/[1-9].jpg	批量下载文件
w3m url	命令行网页浏览器
lynx url	命令行网页浏览器
w3m -dump url	将网页 dump 为文本文件

软件管理	
Debian,Ubuntu,Knoppix(root 权限或 sudo 执行)	
apt-get install curl	从软件仓库中下载并安装软件 curl
aptitude install curl	同上,apt 系列命令一般需要网络连接
apt-get remove curl	移除 curl,
aptitude remove curl	同上
apt-get purge curl	移除 curl 并配置文件
aptitude purge curl	同上
apt-get update	更新可用软件列表
apt-get upgrade	更新系统中所有软件
apt-get dist-upgrade	跨版本升级所有软件
apt-cache show curl	显示 curl 软件包信息
apt-cache pkgnames fi	显示以 fi 开头的软件包
apt-cache search curl	在软件包信息中搜索包含 curl 的包
dpkg -i curl-xx-x.deb	安装手动下载的 deb 软件包
dpkg --get-selections	将系统中所有软件的选择状态输出到
> install.list	install.list
dpkg --set-selections <install.list	依据 install.list 标记软件选择状态,使用下面一条命令应用修改
apt-get dselect-upgrade	根据选择状态的修改更新所有软件
Redhat,Fedora,红旗(root 权限或 sudo 执行)	
yum	网络安装命令
rpm -ivh curl-x-xx.rpm	安装手动下载的 rpm 软件包
rpm -Uvh curl-x-xx.rpm	用下载的 rpm 软件包升级某软件

文本操作	
sort file	对文件内容排序
head file	输出文件头,默认为前 10 行
tail file	输出文件尾,默认为末 10 行
more file	分页查看文件内容,q 退出
less file	插件文件内容,q 退出
cut	文本截取工具,适用以字符和字节单位截取字符串
grep	常用文本搜索命令,详见章节: GREP
sed	常用文本搜索替换命令,详见章节: SED & AWK
awk	常用列表文本处理命令,详见章节: SED & AWK

定时执行命令	
echo “beep” at 21:00	在 21 点执行 beep 命令响铃
crontab -e	编辑 crontab 任务列表

crontab 内容示例:

m h dom mon dow command
每天 0:00 执行 beep
0 0 * * * /usr/bin/beep
每天 8 点 15 和 45 执行~/bin/getup.sh 脚本
15,45 8 * * 5 ~/bin/getup.sh
每 5 分钟执行一次 beep
*/5 * * * * /usr/bin/beep

进程管理	
ps -e	列出所有进程
ps aux	列出所有进程的详细信息
kill 2712	杀死 PID 为 2712 进程
kill -9 PID	将信号 9(KILL)发送给进程 PID
pgrep firefox	查询 firefox 的进程 PID
kill `pgrep ssh`	杀死名称中包含 ssh 的进程
killall firefox	杀死所有命令名中包含 firefox 的进程
killall -e ps	杀死所有命令名精确匹配 ps 的进程
top	列出正在运行的进程,动态刷新
<alt>-F2	调出 KDE 或 GNOME 的程序程序启动器
<ctrl>-<esc>	调出 KDE 的任务管理器
lsof -p PID	查看进程 PID 打开的所有文件
lsof -c proc	查看进程名 proc 打开的所有文件

系统服务控制	
sysv-rc-conf	管理/ect/rc.0~5 启动项, q 退出

声音控制	
alsamixer	alsa 混音器, 调整音量和话筒输入等

BASH SHORTCUTS	
在 bash 下可用的快捷键。	
<esc>-d	从当前位置删除到行尾
<esc>-t	交换光标前面两个词
<ctrl>-a	将光标移动到行首
<ctrl>-e	将光标移动的行尾
<ctrl>-k	从当前位置删除到行尾
<ctrl>-u	从当前位置删除到行首
<ctrl>-w	删除前一个词
<ctrl>-t	交换光标前面两个字符
<ctrl>-r	在命令历史中搜索
<ctrl>-l	清屏
<ctrl>-z	停止当前运行程序并将其转入后台
<ctrl>-d	结束当前运行程序,前台
<ctrl>-v	输入特殊字符,如^M, ^I 等
<ctrl>-d	输入命令时,删除当前字符
<ctrl>-d	输入内容时,代表 EOF
<shift>-Up,PgUp	在虚拟终端中向下移, 或者向上翻页
<shift>-Dn,PgDn	在虚拟终端中向上移, 或者向下翻页

VIM	
vim 是*nix 系统最流行的文本编辑器之一。	
vim file	使用 vim 打开某文件
gvim file	使用 GUI 界面 vim 打开某文件
vimdiff file1 file2	使用 vim 比较两个文件
<Esc>	切换为普通模式
i	切换为插入模式
v	切换为可视模式
V	切换为块模式

/abc	在下文中查找 abc
?abc	在上文中查找 abc
n N	继续查找:下一个 上一个
:1,\$s/abc/def/g	在全文中替换 abc 为 def, 正则表达式
:make	在 vim 中执行 make 命令
:w	保存修改
:q	退出
:q!	强制退出
:wq	保存然后退出
:set nobackup	设置不保存备份文件
:set fencs=utf-8,cp936	设置自动检测 GBK 中文编码
:set ignorecas	设置搜索时忽略大小写
:set number	设置显示行号
:set tabstop=4	设置 tab 占位等同 4 空格, 默认 8
:set list	显示非打印字符
:syntax on off	打开 关闭语法高亮
:colorscheme seashell	使用不同的颜色主题, 需手动下载
~/vimrc	vim 的运行控制文件(run control)

EMACS	
Emacs 是*nix 系统最流行的文本编辑器另外一个之一。	
emacs file	使用 Emacs 打开某文件
emacs -nw file	使用文本界面 Emacs 打开某文件
C-x C-f	创建/打开文件
C-x C-c	退出 Emacs
C-x b	切换编辑缓冲区
C-x k	关闭编辑缓冲区
C-x C-s	保存文件
C-s	渐近式搜索
C-r	反向渐近式搜索
C-M-s	正则表式达搜索
M-x replace-string	字符串替换
C-@	开始标记
C-w	剪切
M-w	复制
C-y	粘贴
M-x font-lock-mode	切换关键字高亮模式
C-g	中断命令执行
C-	撤消操作
C-k	删除行
C-x C-q	切换编辑缓冲区只读模式
~/emcs	Emacs 的配置文件

SED & AWK	
SED	
sed 是对文本流进行过滤处理的工具,依赖正则表达式 (PTN)	
基本应用	
sed 'CMD' in > out	用 CMD 处理输入 in,输出到 out
CMD:=[定位命令]编辑命令	CMD 一般由定位命令和编辑命令组成
	定位命令
x	第 x 行
x,y	第 x 到第 y 行
x,y!	不包括第 x 到 y 行
/pattern/	匹配模式 pattern 的行
x,/pattern/	第 x 行到匹配 pattern 行
/pattern/,x	匹配 pattern 行到第 x 行
/PTN1/,/PTN2/	从匹配 PTN1 行到匹配 PTN2 行

编辑命令	
p	打印定位行
=	打印定位行每行行号
a \TXT	在定位行每行后附加文本行 TXT
i \TXT	在定位行每行前插入文本行 TXT
d	删除定位行
c \abc	用新文本 abc 替换定位行文本
s/PTN/abc/	用新文本 abc 替换匹配模式 PTN
q	匹配第 1 个模式后即退出
r file	附加 file 中文本
R file	同 r , 但仅附加一行
应用举例	
sed -n '5,\$p' in	打印第 5 到最后一行
sed -n '/2008/= ' in	打印包含 2008 的行号
sed '\$a \#END' in	在最后一行后添加一行 #END
sed '12i \#include' in	在第 1 行前添加一行 #include
sed '1,3d' in	删除输入中的 1 到 3 行
sed '/2008/d' in	删除包含 2008 的行, 正则表达式
sed '1,4c \#include' in	将第 1 到 4 行替换为 #include
sed 's/2008/2009/g' in	替换输入中 2008 为 2009
sed '5,\$s/2008/2009/g' in	替换操作应用于第 5 行到最后一行
sed 's/2008/"May "&/g' in	在 2008 前面添加 May
sed 's/May/&" 2008"/g' in	在 May 后面添加 2008
sed '5r a.txt' in	在第 5 行后插入 a.txt 中内容
sed '/2008/r a.txt' in	在包含 2008 下一行插入 a.txt 内容
sed '/2008/R a.txt' in	插入 a.txt 的第一行

AWK	
awk '{print \$2}'	提取标准输入流每行第二个字段值
LFTP	

lftp 是 Linux 下总常用也最强大的命令行 ftp 工具。	
lftp -u root 192.168.0.1	使用 root 用户登录 FTP 站
lftp -p 8021 192.168.0.1	使用指定端口登录 FTP 站
> user solrex	用户名 solrex, 等待提示密码输入
> set ftp:charset GBK	设置站点文件名编码为 GBK
> set file:charset UTF-8	设置本地文件名编码为 UTF-8
> ls	列出站点目录
> !ls	列出本地目录
> !cmd	cmd 加 ! 即为在本地执行
> lcd	改变本地目录
> get file	下载文件 file 到本地目录
> mget *.doc	下载匹配通配符的所有文件
> pget file	多线程下载文件, 用处不大
> mirror dir	镜像 FTP 目录到本地, 慎用
> put file	上传文件 file 到远程目录
> mput *.doc	上传匹配通配符的所有文件
> mirror -R dir	逆 mirror, 即上传
> help	打印帮助信息
> help cmd	打印关于 cmd 命令的帮助信息
> exit	退出 lftp
~/.ftprc	用户运行控制文件

SSH	
ssh 是远程登录 ssh server 的工具	
ssh me@aix.unix-center.net	登录 Unix-Center 体验中心
ssh user@host	以 user 登录某主机, 默认端口 22
ssh user@host -p 2222	以默认端口 2222 登录某主机

ssh -C -f -N -g -R port1:127.0.0.1:port2 user@ip1	将远程 ssh 服务器主机 ip1 端口 port1:127.0.0.1:port2 port1 映射到本机端口 port2
ssh -C -f -N -g -L port1:ip2:port2 user@ip1	将本机端口 port1 映射到服务器主机 ip2 端口 port2
ssh -C -f -N -g -D port1 user@ip1	在本机 port1 建立一个 socks 代理, 所有通信被转发到服务器 ip1
scp file user@ip1:~/	将 file 通过 ssh 上传到 ip1 上
scp user@ip1:~/file .	将 file 通过 ssh 下载到本机
scp -r dir/ user@ip1:~/	将目录 dir 上传到 ip1 上
scp -P 2222 user@ip1:~/dir/ .	将目录 dir 下载到本机, 使用非默认 ssh 端口 2222
sftp user@host	以 sftp 方式登录 ssh 主机

GCC	
gcc 是 *nix 下最流行的编译器集合。	
gcc test.c	编译 test.c 为可执行文件 a.out
gcc -g test.c	编译时添加调试信息
gcc test.c -o test	编译的可执行文件输出为 test
gcc -O2 test.c	以优化水平 2 编译 test.c
g++ test.cpp	编译 test.cpp, C++ 程序
gcc -std=c99 test.c	以 ISO C99 兼容标准编译, 默认为 gnu89
gcc -c test.c	编译成目标文件, 默认输出到 test.o
gcc -S test.c	编译成汇编代码, 默认输出到 test.s
gcc test.c -lm	编译时链接数学库

GDB	
gdb 是 GNU 的调试工具。	
gdb a.out	用 gdb 开始调试 a.out, 需用 gcc -g 编译
gdb --pid=2002	用 gdb 调试正在运行的 PID 为 2002 的进程
> r	开始运行程序
> c	继续运行程序
> b 10	在当前源文件第十行设置断点
> b foo	在函数 foo 入口设置断点
> b a.c:10	在 a.c 的第十行设置断点
> b *0xabcdefgh	在内存的 0xabcdefg 位置设置断点
> s	单步执行
> stepi	单指令执行
> list	列出当前运行位置附近源文件内容
> disassemble	反汇编当前运行位置附近机器指令
> p var	打印变量 var 内容
> p/x var	十六进制打印变量 var 内容
> info registers	打印计算机寄存器信息
> p/x \$eax	十六进制打印寄存器 eax 内容
> x/10 0xabcdef	查看内存位置 0xabcdef 及后 10 字的内容



©2009 Solrex Yang(杨文博) <<http://solrex.cn>>
版本: 零点零三
二零零九年六月九日
协议: 创作共享协议 (署名-非商业-相同方式共享)
主页: <http://share.solrex.cn/cheatsheet/slc>