



马哥教育
最专业的Linux培训机构

Linux入门

<http://www.magedu.com>

- ❖ 用户登录
- ❖ 终端
- ❖ **Shell**介绍
- ❖ 执行命令
- ❖ 简单命令
- ❖ **Tab**键补全
- ❖ 命令行历史
- ❖ **bash**快捷键
- ❖ 使用**gnome-terminal**

马哥教育
www.magedu.com

❖ root 用户

一个特殊的管理帐户

也被称为超级用户

root已接近完整的系统控制

对系统损害几乎有无限的能力

除非必要,不要登录为 **root**

❖ 普通（非特权）用户

权限有限

造成损害的能力比较有限

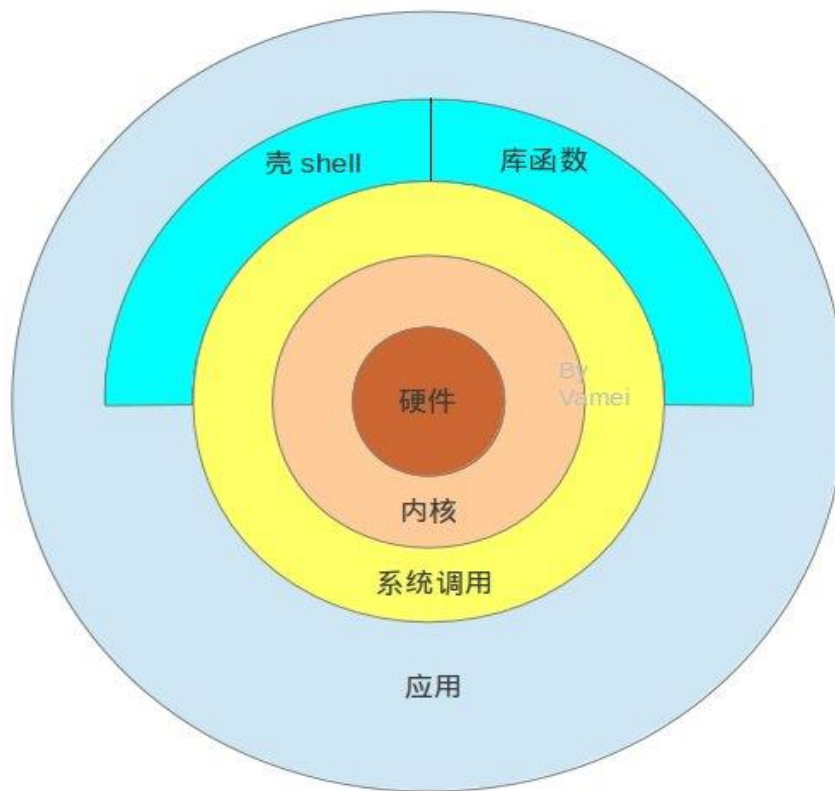
马哥教育
www.magedu.com



- ❖ 设备终端
键盘鼠标显示器
- ❖ 物理终端 (`/dev/console`)
控制台console
- ❖ 虚拟终端(`tty: teletypewriters, /dev/tty#` #为[1-6])
`tty` 可有n个, `Ctrl+Alt+F[1-6]`
- ❖ 图形终端 (`/dev/tty7`) `startx, xwindows`
`CentOS 6: Ctrl + Alt + F7`
`CentOS 7: 在哪个终端启动, 即位于哪个虚拟终端`
- ❖ 串行终端 (`/dev/ttyS#`)
`ttyS`
- ❖ 伪终端 (`pty: pseudo-tty , /dev/pts/#`)
`pty, SSH`远程连接
- ❖ 查看当前的终端设备: `#tty`

- ❖ 交互式接口：启动终端后，在终端设备附加一个交互式应用程序
- ❖ GUI: Graphic User Interface
 - X protocol, window manager, desktop
 - Desktop:
 - GNOME (C, 图形库gtk),
 - KDE (C++, 图形库qt)
 - XFCE (轻量级桌面)
- ❖ CLI: Command Line Interface
 - shell程序: sh (bourn 史蒂夫·伯恩) csh tcsh
 - ksh (korn) bash (bourn again shell)GPL zsh

- ❖ **Shell** 是Linux系统的用户界面，提供了用户与内核进行交互操作的一种接口。它接收用户输入的命令并把它送入内核去执行
- ❖ **shell**也被称为**LINUX**的命令解释器（**command interpreter**）
- ❖ **shell**是一种高级程序设计语言



- ❖ GNU Bourne-Again Shell(bash)是GNU计划中重要的工具软件之一，目前也是 Linux标准的shell，与sh兼容
- ❖ CentOS默认使用
- ❖ 显示当前使用的shell
`echo ${SHELL}`
- ❖ 显示当前系统使用的所有shell
`cat /etc/shells`

马哥教育

www.magedu.com

命令提示符

❖ 命令提示符: prompt

```
[root@localhost ~]#
```

管理员 \$ 普通用户

❖ 显示提示符格式

```
[root@localhost ~]#echo $PS1
```

❖ 修改提示符格式

```
PS1="\[\e[1;5;41;33m\][\u@\h \W]\\$\[\e[0m\]"
```

\e \033

\u 当前用户

\h 主机名简称

\H 主机名

\w 当前工作目录

\W 当前工作目录基名

\t 24小时时间格式

\T 12小时时间格式

\! 命令历史数

\# 开机后命令历史数

❖ 输入命令后回车

提请**shell**程序找到键入命令所对应的可执行程序或代码，并由其分析后提交给内核分配资源将其运行起来

❖ 在**shell**中可执行的命令有两类

内部命令：由**shell**自带的，而且通过某命令形式提供

help 内部命令列表

enable cmd 启用内部命令

enable -n cmd 禁用内部命令

enable -n 查看所有禁用的内部命令

外部命令：在文件系统路径下有对应的可执行程序文件

查看路径：**which -a | --skip-alias ; whereis**

❖ 区别指定的命令是内部或外部命令

type COMMAND

执行外部命令

❖ Hash缓存表

系统初始hash表为空，当外部命令执行时，默认会从PATH路径下寻找该命令，找到后会将这条命令的路径记录到hash表中，当再次使用该命令时，shell解释器首先会查看hash表，存在将执行之，如果不存在，将会去PATH路径下寻找。利用hash缓存表可大大提高命令的调用速率

❖ hash常见用法

hash 显示hash缓存

hash -l 显示hash缓存，可作为输入使用

hash -p path name 将命令全路径path起别名为name

hash -t name 打印缓存中name的路径

hash -d name 清除name缓存

hash -r 清除缓存

- ❖ 显示当前**shell**进程所有可用的命令别名

alias

- ❖ 定义别名**NAME**，其相当于执行命令**VALUE**

alias NAME='VALUE'

- ❖ 在命令行中定义的别名，仅对当前**shell**进程有效

- ❖ 如果想永久有效，要定义在配置文件中

仅对当前用户：**~/.bashrc**

对所有用户有效：**/etc/bashrc**

马哥教育
www.magedu.com

❖ 编辑配置给出的新配置不会立即生效

❖ **bash**进程重新读取配置文件

```
source /path/to/config_file
```

```
./path/to/config_file
```

❖ 撤消别名: **unalias**

```
unalias [-a] name [name ...]
```

-a 取消所有别名

❖ 如果别名同原命令同名，如果要执行原命令，可使用

"\COMMAND"

'COMMAND'

/PATH/COMMAND: 外部命令

❖ **COMMAND [OPTIONS...] [ARGUMENTS...]**

选项：用于启用或关闭命令的某个或某些功能

短选项：-c 例如：-l, -h

长选项：--word 例如：--all, --human-readable

参数：命令的作用对象，比如文件名，用户名等

❖ 注意：

- ➡ 多个选项以及多参数和命令之间使用空白字符分隔
- ➡ 取消和结束命令执行：Ctrl+c, Ctrl+d
- ➡ 多个命令可以用;符号分开
- ➡ 一个命令可以用\分成多行

❖ Linux的两种时钟

系统时钟：由Linux内核通过CPU的工作频率进行的

硬件时钟：主板

❖ 相关命令

date 显示和设置系统时间

date +%s

date -d @1509536033

hwclock, clock: 显示硬件时钟

-s, --hctosys 以硬件时钟为准，校正系统时钟

-w, --systohc 以系统时钟为准，校正硬件时钟

❖ 时区：/etc/localtime

❖ 显示日历：cal -y

简单命令

❖ 关机: **halt, poweroff**

❖ 重启: **reboot**

-f: 强制, 不调用**shutdown**

-p: 切断电源

❖ 关机或重启: **shutdown**

shutdown [OPTION]... TIME [MESSAGE]

-r: **reboot**

-h: **halt**

-c: **cancel**

TIME: 无指定, 默认相当于 **+1**

now: 立刻, 相当于 **+0**

+m: 相对时间表示法, 几分钟之后; 例如 **+3**

hh:mm: 绝对时间表示, 指明具体时间

❖ 用户登录信息查看命令：

whoami：显示当前登录有效用户

who：系统当前所有的登录会话

w：系统当前所有的登录会话及所做的操作

❖ nano 文本编辑

马哥教育

www.magedu.com

❖ screen命令:

创建新screen会话

```
screen -S [SESSION]
```

加入screen会话

```
screen -x [SESSION]
```

退出并关闭screen会话

```
exit
```

剥离当前screen会话

```
Ctrl+a,d 马哥教育
```

显示所有已经打开的screen会话

```
screen -ls
```

恢复某screen会话

```
screen -r [SESSION]
```

- ❖ **echo**命令
- ❖ 功能：显示字符
- ❖ 语法：**echo [-neE][字符串]**
- ❖ 说明：**echo**会将输入的字符串送往标准输出。输出的字符串间以空白字符隔开，并在最后加上换行号
- ❖ 选项：
 - ➔ **-E** （默认）不支持 \ 解释功能
 - ➔ **-n** 不自动换行
 - ➔ **-e** 启用 \ 字符的解释功能
- ❖ 显示变量
 - echo "\$VAR_NAME"** 变量会替换，弱引用
 - echo '\$VAR_NAME'** 变量不会替换，强引用

❖ 启用命令选项 **-e**，若字符串中出现以下字符，则特别加以处理，而不会将它当成一般文字输出

\a 发出警告声

\b 退格键

\c 最后不加上换行符号

\n 换行且光标移至行首

\r 回车，即光标移至行首，但不换行

\t 插入 **tab**

**** 插入 **** 字符

\Onnn 插入 **nnn**（八进制）所代表的 **ASCII** 字符

echo -e '\033[43;31;5mmagedu\033[0m'

\xHH 插入 **HH**（十六进制）所代表的 **ASCII** 字

ASCII (American Standard Code for Information Interchange)

ASCII 字符代码表 一

高四位 低四位		ASCII非打印控制字符										ASCII 打印字符												
		0000					0001					0010	0011		0100	0101		0110		0111				
		0					1					2	3		4	5		6		7				
		十进制	字符	ctrl	代码	字符解释	十进制	字符	ctrl	代码	字符解释	十进制	字符	十进制	字符	十进制	字符	十进制	字符	十进制	字符	ctrl		
0000	0	0	BLANK NULL	^@	NUL	空	16	▶	^P	DLE	数据链路转意	32		48	0	64	@	80	P	96	`	112	p	
0001	1	1	☺	^A	SOH	头标开始	17	◀	^Q	DC1	设备控制 1	33	!	49	1	65	A	81	Q	97	a	113	q	
0010	2	2	☹	^B	STX	正文开始	18	↕	^R	DC2	设备控制 2	34	"	50	2	66	B	82	R	98	b	114	r	
0011	3	3	♥	^C	ETX	正文结束	19	!!	^S	DC3	设备控制 3	35	#	51	3	67	C	83	S	99	c	115	s	
0100	4	4	♦	^D	EOI	传输结束	20	¶	^T	DC4	设备控制 4	36	\$	52	4	68	D	84	T	100	d	116	t	
0101	5	5	♣	^E	ENQ	查询	21	♫	^U	NAK	反确认	37	%	53	5	69	E	85	U	101	e	117	u	
0110	6	6	♠	^F	ACK	确认	22	■	^V	SYN	同步空闲	38	&	54	6	70	F	86	V	102	f	118	v	
0111	7	7	●	^G	BEL	震铃	23	↑↓	^W	ETB	传输块结束	39	'	55	7	71	G	87	w	103	g	119	w	
1000	8	8	◼	^H	BS	退格	24	↑	^X	CAN	取消	40	(56	8	72	H	88	X	104	h	120	x	
1001	9	9	○	^I	TAB	水平制表符	25	↓	^Y	EM	媒体结束	41)	57	9	73	I	89	Y	105	i	121	y	
1010	A	10	◻	^J	LF	换行/新行	26	→	^Z	SUB	替换	42	*	58	:	74	J	90	Z	106	j	122	z	
1011	B	11	♂	^K	VT	坚直制表符	27	←	^[ESC	转意	43	+	59	;	75	K	91	[107	k	123	{	
1100	C	12	♀	^L	FF	换页/新页	28	└	^\	FS	文件分隔符	44	,	60	<	76	L	92	\	108	l	124		
1101	D	13	♪	^M	CR	回车	29	↔	^]	GS	组分分隔符	45	-	61	=	77	M	93]	109	m	125	}	
1110	E	14	🎵	^N	SO	移出	30	▲	^6	RS	记录分隔符	46	.	62	>	78	N	94	^	110	n	126	~	
1111	F	15	◼	^O	SI	移入	31	▼	^-	US	单元分隔符	47	/	63	?	79	O	95	_	111	o	127	Δ	Back space

注：表中的ASCII字符可以用:ALT + “小键盘上的数字键”输入

命令行扩展、被括起来的集合

❖ 命令行扩展：\$() 或 ``

把一个命令的输出打印给另一个命令的参数

```
echo "This system's name is $(hostname) "
```

```
This system's name is server1.example.com
```

```
echo "i am `whoami` "
```

```
i am root
```

❖ 括号扩展：{ }

打印重复字符串的简化形式

```
echo file{1,3,5}
```

```
file1 file3 file5
```

```
rm -f file{1,3,5}
```

```
echo {1..10}
```

```
echo {a..z}
```

```
echo {000..20..2}
```

❖ 命令补全

内部命令：

外部命令：**bash**根据**PATH**环境变量定义的路径，自左而右在每个路径搜寻以给定命令命名的文件，第一次找到的命令即为要执行的命令

用户给定的字符串只有一条唯一对应的命令，直接补全
否则，再次**Tab**会给出列表

❖ 路径补全

把用户给出的字符串当做路径开头，并在其指定上级目录下搜索以指定的字符串开头的文件名

如果惟一：则直接补全

否则：再次**Tab**给出列表

- ❖ **command 2Tab** 所有子命令或文件补全
- ❖ **string2Tab** 以**string**开头命令
- ❖ **/2Tab** 显示所有根目录下一级目录，包括隐藏目录
- ❖ **./2Tab** 当前目录下子目录，包括隐藏目录
- ❖ ***2Tab** 当前目录下子目录，不包括隐藏目录
- ❖ **~2Tab** 所有用户列表
- ❖ **\$2Tab** 所有变量
- ❖ **@2Tab** **/etc/hosts**记录（centos7不支持）
- ❖ **=2Tab** 相当于**ls -A**（centos7不支持）

- ❖ 保存你输入的命令历史。可以用它来重复执行命令
- ❖ 登录**shell**时，会读取命令历史文件中记录下的命令
`~/.bash_history`
- ❖ 登录进**shell**后新执行的命令只会记录在缓存中；这些命令会用户退出时“追加”至命令历史文件中

马哥教育

www.magedu.com

- ❖ 重复前一个命令，有4种方法
 - ➡ 重复前一个命令使用上方向键，并回车执行
 - ➡ 按 **!!** 并回车执行
 - ➡ 输入 **!-1** 并回车执行
 - ➡ 按 **Ctrl+p** 并回车执行
- ❖ **!:0** 执行前一条命令（去除参数）
- ❖ **Ctrl + n** 显示当前历史中的下一条命令，但不执行
- ❖ **Ctrl + j** 执行当前命令
- ❖ **!n** 执行**history**命令输出对应序号**n**的命令
- ❖ **!-n** 执行**history**历史中倒数第**n**个命令

- ❖ **!string** 重复前一个以“**string**”开头的命令
- ❖ **!?string** 重复前一个包含**string**的命令
- ❖ **!string:p** 仅打印命令历史，而不执行
- ❖ **!\$:p** 打印输出 **!\$**（上一条命令的最后一个参数）的内容
- ❖ **!*:p** 打印输出 **!***（上一条命令的所有参数）的内容
- ❖ **^string** 删除上一条命令中的第一个**string**
- ❖ **^string1^string2** 将上一条命令中的第一个**string1**替换为**string2**
- ❖ **!:gs/string1/string2** 将上一条命令中所有的**string1**都替换为 **string2**

- ❖ 使用**up**（向上）和**down**（向下）键来上下浏览从前输入的命令
- ❖ **ctrl-r**来在命令历史中搜索命令
 - **(reverse-i-search) `**:
- ❖ **Ctrl+g**: 从历史搜索模式退出
- ❖ 要重新调用前一个命令中最后一个参数:
 - **!\$** 表示
 - **Esc, .**（点击**Esc**键后松开，然后点击 **.** 键）
 - **Alt+ .**（按住**Alt**键的同时点击 **.** 键）

www.magedu.com

- ❖ `command !^` : 利用上一个命令的第一个参数做`cmd`的参数
- ❖ `command !$` : 利用上一个命令的最后一个参数做`cmd`的参数
- ❖ `command !*` : 利用上一个命令的全部参数做`cmd`的参数
- ❖ `command !:n` : 利用上一个命令的第`n`个参数做`cmd`的参数

马哥教育

www.magedu.com

- ❖ `command !n:^` 调用第n条命令的第一个参数
- ❖ `command !n:$` 调用第n条命令的最后一个参数
- ❖ `command !n:m` 调用第n条命令的第m个参数
- ❖ `command !n:*` 调用第n条命令的所有参数

马哥教育

www.magedu.com

- ❖ **command !string:^** 从命令历史中搜索以 **string** 开头的命令，并获取它的第一个参数
- ❖ **command !string:\$** 从命令历史中搜索以 **string** 开头的命令，并获取它的最后一个参数
- ❖ **command !string:n** 从命令历史中搜索以 **string** 开头的命令，并获取它的第n个参数
- ❖ **command !string:*** 从命令历史中搜索以 **string** 开头的命令，并获取它的所有参数

马哥教育

www.magedu.com

- ❖ `history [-c] [-d offset] [n]`
- ❖ `history -anrw [filename]`
- ❖ `history -ps arg [arg...]`
 - c: 清空命令历史
 - d **offset**: 删除历史中指定的第**offset**个命令
 - n: 显示最近的n条历史
 - a: 追加本次会话新执行的命令历史列表至历史文件
 - r: 读历史文件附加到历史列表
 - w: 保存历史列表到指定的历史文件
 - n: 读历史文件中未读过的行到历史列表
 - p: 展开历史参数成多行, 但不存在历史列表中
 - s: 展开历史参数成一行, 附加在历史列表后

命令历史相关环境变量

- ❖ **HISTSIZE**: 命令历史记录的条数
- ❖ **HISTFILE**: 指定历史文件, 默认为`~/.bash_history`
- ❖ **HISTFILESIZE**: 命令历史文件记录历史的条数
- ❖ **HISTTIMEFORMAT**="`%F %T`" 显示时间
- ❖ **HISTIGNORE**="`str1:str2*:...`" 忽略`str1`命令, `str2`开头的历史
- ❖ 控制命令历史的记录方式:

环境变量: **HISTCONTROL**

ignoredups 默认, 忽略重复的命令, 连续且相同为“重复”

ignorespace 忽略所有以空白开头的命令

ignoreboth 相当于**ignoredups**, **ignorespace**的组合

erasedups 删除重复命令

- ❖ **export** 变量名="值"
- ❖ 存放在 `/etc/profile` 或 `~/.bash_profile`

- ❖ **Ctrl + l** 清屏，相当于**clear**命令
- ❖ **Ctrl + o** 执行当前命令，并重新显示本命令
- ❖ **Ctrl + s** 阻止屏幕输出，锁定
- ❖ **Ctrl + q** 允许屏幕输出
- ❖ **Ctrl + c** 终止命令
- ❖ **Ctrl + z** 挂起命令

马哥教育

www.magedu.com

- ❖ **Ctrl + a** 光标移到命令行首，相当于**Home**
- ❖ **Ctrl + e** 光标移到命令行尾，相当于**End**
- ❖ **Ctrl + f** 光标向右移动一个字符
- ❖ **Ctrl + b** 光标向左移动一个字符
- ❖ **Alt + f** 光标向右移动一个单词尾
- ❖ **Alt + b** 光标向左移动一个单词首
- ❖ **Ctrl + xx** 光标在命令行首和光标之间移动
- ❖ **Ctrl + u** 从光标处删除至命令行首
- ❖ **Ctrl + k** 从光标处删除至命令行尾
- ❖ **Alt + r** 删除当前整行

- ❖ **Ctrl + w** 从光标处向左删除至单词首
- ❖ **Alt + d** 从光标处向右删除至单词尾
- ❖ **Ctrl + d** 删除光标处的一个字符
- ❖ **Ctrl + h** 删除光标前的一个字符
- ❖ **Ctrl + y** 将删除的字符粘贴至光标后
- ❖ **Alt + c** 从光标处开始向右更改为首字母大写的单词
- ❖ **Alt + u** 从光标处开始，将右边一个单词更改为大写
- ❖ **Alt + l** 从光标处开始，将右边一个单词更改为小写
- ❖ **Ctrl + t** 交换光标处和之前的字符位置
- ❖ **Alt + t** 交换光标处和之前的单词位置
- ❖ **Alt + N** 提示输入指定字符后，重复显示该字符N次
- ❖ 注意：**Alt**组合快捷键经常和其它软件冲突

- ❖ 应用程序->附件->终端
- ❖ 支持多个“标签”**shell**的图形化终端模拟器
 - **Ctrl-Shift-n** 创建新窗口
 - **Ctrl-Shift-q** 关闭当前窗口
 - **Ctrl-Shift-t** 创建一个标签页
 - **Ctrl-Shift-w** 关闭一个标签页
 - **Ctrl-PaUp/PgUn** 切换到下一个/前一个标签页
 - **Ctrl-Shift-c** 复制选中的文本
 - **Ctrl-Shift-v** 把文本粘帖到当前光标处

- ❖ 博客: <http://magedu.blog.51cto.com>
- ❖ 主页: <http://www.magedu.com>
- ❖ QQ: 1661815153, 113228115
- ❖ QQ群: 203585050, 279599283

马哥教育
www.magedu.com



马哥教育
最专业的Linux培训机构

Thank You!