

Linux基础 - 环境变量

目录

01

环境变量介绍

02

常见环境变量操作

目录

01

环境变量介绍

02

常见环境变量操作

环境变量介绍

- Linux环境变量是在Linux系统中定义的用于控制系统和程序运行的变量
- Linux环境变量包括系统路径、用户账号等信息，决定了系统的资源访问位置和程序的执行路径。
- Linux环境变量包括系统级和用户级的
 - 系统级的环境变量是每个登录到系统的用户都要读取的系统变量
 - 用户级的环境变量则是该用户使用系统时加载的环境变量。

环境变量介绍

按照作用生命周期和作用域可以做如下划分：

分类类型	分类性质	分类定义
生命周期	永久性	需要用户修改相关的配置文件，变量永久生效
	临时性	用户利用export命令，在当前终端下声明环境变量，关闭Shell终端失效
作用域	系统级	对系统中的所有用户都有效
	用户级	只对特定的用户有效

环境变量介绍

- Linux系统有一些常用的系统默认的环境变量，具体如下：

环境变量	说明
PATH	指定系统可执行程序或者命令的路径列表
HOME	指定用户主目录的路径
SHELL	指定用户默认使用的Shell环境
LOGNAME	指定用户的登录名
LANG	指定系统默认编码类型
PWD	指定当前工作目录的路径
MAIL	指定用户默认的邮件文件路径
PS1	指定默认的终端命令提示符

环境变量介绍

- Linux系统主要的环境变量配置文件，如下表所列：

配置文件	配置作用
/etc/profile	设置系统环境参数，对系统内所有用户生效
~/.bash_profile	设置用户环境参数，只针对当前用户生效
/etc/bashrc	设置系统环境参数，只针对运行bash的shell生效
~/.bashrc	作用类似于/etc/bashrc，只针对当前用户生效

环境变量介绍

```
# It's NOT a good idea to change this file unless you know what you  
# are doing. It's much better to create a custom.sh shell script in  
# /etc/profile.d/ to make custom changes to your environment. This script  
# will prevent the need for merging in future updates.
```

```
pathmunge () {  
    case "${PATH}" in  
        *:~:*)  
            ;;  
        *)  
            if [ "$2" = "after" ] ; then  
                PATH=$PATH:$1  
            else  
                PATH=$1:$PATH  
            fi  
        esac  
    }
```

```
# are we an interactive shell?  
if [ "$PS1" ]; then  
    if [ -z "$PROMPT_COMMAND" ]; then  
        case $TERM in  
            xterm*|vte*)  
                if [ -e /etc/sysconfig/bash-prompt-xterm ]; then  
                    PROMPT_COMMAND=/etc/sysconfig/bash-prompt-xterm  
                elif [ "${VTE_VERSION:-0}" -ge 3405 ]; then  
                    PROMPT_COMMAND="__vte_prompt_command"  
                else  
                    PROMPT_COMMAND='printf "\033]0;%s@%s:%s\007"  
}' "${PWD/#$HOME/~}"'  
                fi  
            fi  
        fi  
    fi  
fi
```

目录

01

环境变量介绍

02

常见环境变量操作

常见环境变量操作

■ Linux环境变量常见操作示例

- 显示当前系统所有环境变量

```
env
```

```
[root@rocky ~]# env
XDG_SESSION_ID=31
HOSTNAME=rocky
SELINUX_ROLE_REQUESTED=
TERM=xterm
SHELL=/bin/bash
HISTSIZE=1000
SSH_CLIENT=192.168.110.1 55788 22
SELINUX_USE_CURRENT_RANGE=
SSH_TTY=/dev/pts/1
USER=root
LS_COLORS=rs=0:di=01;34:ln=01;36:mh=00:pi=40;33:so=01;35:do=01;35:bd=40;33
x=01;32:*.tar=01;31:*.tgz=01;31:*.arc=01;31:*.arj=01;31:*.taz=01;31:*.lha=
```

常见环境变量操作

■ Linux环境变量常见操作示例

- 显示本地定义的环境变量

```
set
```

```
_yu_yumdownloader ()
{
    local cur prev words=() split=false;
    _yu_init_completion "$2" "$3";
    _yum_complete_baseopts "$cur" "$prev" 2> /dev/null && return 0;
    case $prev in
        --destdir)
            COMPREPLY=( $( compgen -d -- "$cur" ) );
            return 0
            ;;
        --archlist)
            return 0
            ;;
    esac;
    $split && return 0;
    if [[ $cur == -* ]]; then
        COMPREPLY=( $( compgen -W ' $( _yum_baseopts 2>/dev/null ) --destdir
            --urls --resolve --source --archlist' -- "$cur" ) );
        return 0;
    fi;
    _yum_list all "$cur"
}
```

常见环境变量操作

■ Linux环境变量常见操作示例

● 显示某个环境变量的值

```
echo <某个待查询的环境变量>
```

```
[root@rocky ~]# echo $PATH
/opt/clusconf//bin:/opt/clusconf//interpreter/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/root/bin
[root@rocky ~]# echo $HOME
/root
[root@rocky ~]# █
```

● 设置某个环境变量的值

```
export <key=value>
```

```
[root@rocky ~]# export HELLO="hello"
[root@rocky ~]# echo $HELLO
hello
[root@rocky ~]# █
```

常见环境变量操作

■ Linux环境变量常见操作示例

- export常用参数有以下两个

参数	说明
-n	移除环境变量
-p	打印所有环境变量

```
[root@rocky ~]# export HELLO=hello
[root@rocky ~]# export -p|grep -i hello
declare -x HELLO="hello"
[root@rocky ~]# export -n HELLO
[root@rocky ~]# export -p|grep -i hello
[root@rocky ~]#
```

常见环境变量操作

■ Linux环境变量常见操作示例

- declare命令设置环境变量，该命令语法如下：

```
declare [+/-][选项] [变量名=变量值]
```

选项	说明
-f	仅显示函数
r	将变量设置为只读
x	指定的变量会成为环境变量，可供shell以外的程序来使用
i	设置值可以是数值，字符串或运算式

```
[root@rocky ~]# declare -x mag="hello world"
[root@rocky ~]# export -p|grep mag
declare -x mag="hello world"
[root@rocky ~]# █
```

常见环境变量操作

■ Linux环境变量常见操作示例

● 设置只读环境变量

```
readonly <key=value>
```

```
[root@rocky ~]# readonly TEST=test_path  
[root@rocky ~]# echo $TEST  
test_path  
[root@rocky ~]# export TEST=test_path1  
-bash: TEST: 只读变量  
[root@rocky ~]# █
```

常见环境变量操作

■ Linux环境变量常见操作示例

● 清除环境变量值

```
unset <待清除的环境变量值>
```

```
[root@rocky ~]# unset HELLO  
[root@rocky ~]# echo $HELLO  
  
[root@rocky ~]# █
```

常见环境变量操作

■ Linux环境变量常见操作示例

- 修改/etc/bashrc文件 (对所有用户有效)

```
unset i
unset -f pathmunge
fi
# vim:ts=4:sw=4
export PATH=$PATH:/tmp/test2
```

```
[root@rocky ~]# source /etc/bashrc
[root@rocky ~]# echo $PATH
/opt/clusconf//bin:/opt/clusconf//interpreter/bin:/opt/clusconf//bin:/opt/clusconf//l/bin:/usr/sbin:/usr/bin:/tmp/test2:/tmp/test1:/root/bin:/tmp/test2
[root@rocky ~]#
```

常见环境变量操作

- Linux环境变量常见操作示例
 - 将变量a=2追加到/etc/profile文件
 - 将b=3追加到~/.bashrc文件

```
[root@rocky ~]# echo "a=2" >> /etc/profile
[root@rocky ~]# echo "b=3" >> /etc/bashrc
[root@rocky ~]# source /etc/bashrc
[root@rocky ~]# source /etc/profile
[root@rocky ~]# echo $a
2
[root@rocky ~]# echo $b
3
[root@rocky ~]# █
```

The background of the slide is a dark red color with a subtle, light-colored circuit board pattern. The pattern consists of various lines, curves, and small circular nodes, resembling a printed circuit board (PCB) layout. The lines are thin and light, creating a technical and modern aesthetic.

谢谢