# UNIX高级环境编程学习笔记

目录

[UNIX高级环境编程学习笔记 1](#_Toc516411251)

[第零天 2](#_Toc516411252)

[第一天 2](#_Toc516411253)

[第三天-Linux基本常用命令学习 3](#_Toc516411254)

[搭建LAMP服务器： 10](#_Toc516411255)

[第四天：进行unix高级环境编程的代码练习 14](#_Toc516411256)

[Makefile规则 14](#_Toc516411257)

[配置多个IP方法 19](#_Toc516411258)

[Windows CMD命令积累 21](#_Toc516411259)

[windows使用说明 22](#_Toc516411260)

## 第零天

1.cd ~/Desktop/ //可以进入桌面

用户名是在/home/ 之下的

2. tar -zxvf src.3e.tar.gz //对src.3e.tar.gz进行解压缩

3. yum install //在线安装软件

4.su //进入root权限

5.pwd //打印当前工作目录

6.cd ../ //返回父级

7.cp //拷贝文件

8. apt-get //从网上下载文件 ubantu版本可用

9. ./configure //确认当前环境下依赖存在，生成makefile

10. make //编译一波 存在makefile才可以make 编译

11. make install //安装一波

10. makefile //制定每一条规则

头文件（.h）作用，就是函数声明

12.gedit c.c //创建一个c文件

13.gcc c.c –o lishuai //编译c.c文件 自定义lishuai.out

14. ./a.out //执行a.out文件

做一个小型的http服务器mongoose

## 第一天

7月11日 第一天学习这本书

之前也没有怎么接触过unix系统，所以心里还是有点发慌，不过都已经上车了，就坚持下去咯，好好的搞一波这个东西

1.下载APUE

下载地址：<http://apuebook.com/code3e.html>

解压缩tar -zxvf src.3e.tar.gz

2.下载libbsd-devel，这个无法利用yum安装，直接在网上下载压缩包，解压缩之后是一个.h文件。

3.将apue进行安装

操作如下：cd apue.3e---./configure----(makeclean //清除目标文件)---make---make install

用root用户或者sudo执行以下命令：

cp ./include/apue.h /usr/include/

cp ./lib/libapue.a /usr/local/lib/

cp ./lib/libapue.a /usr/lib/

4.创建apueerror.h头文件，

下载地址：<http://download.csdn.net/detail/aboboo5200/9766770>

cp apueerror.h /usr/include/

在要编译运行的代码中#include<apue.h>的下一行增加一行：#include<apueerror.h>

引用到err\_相关的方法的实例代码的头部引入apueerror.h头文件

参考链接：<http://blog.csdn.net/aboboo5200/article/details/58716172>

gcc –c p9.c //生成p9.o

gcc –o p9 p9.o //生成p9s

./p9 //执行p9

一般情况

gcc p9.c

./a.out //执行

## 第三天-Linux基本常用命令学习

关机命令：shutdown -h now

重启命令：shutdown -r now

reboot

man tar / tar –help 可以查看命令的相关解释

帮助命令:

eg:

1. man hexo
2. hexo -h

文件的学习：

1．cd / 与 cd ~=cd的区别：cd / 进入根目录 cd ~ 进入用户主目录/root

cd - //可以返回刚才的目录

2．文件权限：

改变文件权限：chmod 其中rwx，r=4，w=2，x=1，

chmod 777 文件名

chmod 777 alarmConfig.json

chown nobody alarmConfig.json

chgrp nobody alarmConfig.json

chown nobody:nobody alarmConfig.json

3mkdir 文件目录名 //新建目录 创建文件夹

touch 文件名 //新建文件

cat c.c可以查看文件内容

cat > c.c 可以创建文件 向文件中输入内容

grep 文本搜索

cat主要有三大功能：

1.一次显示整个文件。

$ cat filename

2.从键盘创建一个文件。

$ cat > filename

只能创建新文件,不能编辑已有文件.

3.将几个文件合并为一个文件。

rm –rf 删除文件 命令和参数之间需要有空格

rm –i //删除时提示

cp -rf alarm\_config /test

cp –a dir1 dir2 复制一个目录

alias //起别名

tmux //窗口管理程序

ll thanks\* //将thanks各种后缀名的文件显示出来

ls –a //可以将隐藏文件显示出来

ls –al ~/.ssh //将~下的所有的.ssh显示出来

pwd //打印当前工作目录

file filename //可以查看文件的类型和一些详细的信息

mv //移动文件

clear //清除屏幕的命令行

yum –y //不再提示yes/no 回车也可以代替

service iptables restart //服务重启

service iptables status

ifconfig //查看ip

systemctl stop firewalld.service 关闭服务

systemctl disable firewalld.service 关闭开机自启动

4.压缩文件后缀名：.tar/.tar.gz //这种是打包压缩文件

.rpm //是经过编译的可安装文件

5.linux 目录配置：

/etc 配置文件

/bin 可行性文件

/dev 设备

/lib 库函数

/boot 启动有关的文件

/usr Unix Softwore resource

6. ./代表当前目录，在执行一些文件的时候需要用./

7.virtual filesystem switch 虚拟文件系统 内核（需要去重点学习）

8.第八章Linux磁盘与文件系统管理

扇区sector：为最小的物理存储单位，每个扇区为512byte

柱面cylinder：柱面是分区的最小的单位

第一个扇区最重要，硬盘主引导记录MBR及分区表partition table，

df -h 查看磁盘使用量

du –sh \* 显示当前

mount //磁盘挂载

umount //磁盘卸载

fdisk –l //查看磁盘的情况

df 查看分区使用情况

du 查看文件占用空间情况

9.第九章 文件与文件系统的压缩与打包

文件的压缩就是将连续的1或0，进行索引表示

gzip压缩文件

gzip –v 文件名 显示压缩比

gzip –d 文件名 删除文件

tar 打包命令

tar –zpcv –f lishuai.tar.gz lishuai

tar –zxvf lishuai.tar.gz //解压缩

tar –zxvf lishuai.tar.gz –C /tmp //指定路径进行解压缩

10.vim程序编辑器

一般模式：

dd //删除当前一行

yy //复制当前行

p //粘贴

nyy //复制n行

u //撤销

可以利用hjkl进行光标的移动

编辑模式：

I 插入模式 从光标所在处插入

命令行模式：

:w 将文件写入

:q 不保存退出

:q! 强制退出

:wq 保存并推出

:n //跳转到第n行

:u //撤销操作

/ //查找

locate //快速查找文件

11.认识学习bash

history //可以查看之前操作过的命令

12.第十三章 学习shell script

1. awk是一个强大的文本分析工具，相对于grep的查找，sed的编辑，awk在其对数据分析并生成报告时，显得尤为强大。简单来说awk就是把文件逐行的读入，以空格为默认分隔符将每行切片，切开的部分再进行各种分析处理。

2.

#!/bin/bash

sh sh01.sh //调用shell命令

13.第十四章 linux账户管理

增加用户:

useradd ls

passwd ls

删除用户:

userdel peter

groupdel peter

ps –l //查看进程的情况

cat /etc/services //查看服务

14.软件安装

config | configure //生成makefile 编译脚本

make //根据makefile 编译源文件

make install //安装软件

makeclean //在生成makefile之前，先清除一下中间文件

利用rpm进行安装（.rpm）

rpm –ivh package\_name

rpm –qc 文件名 //查询文件所处的位置

rpm –qa | grep 文件名 //查看依赖

rpm –e 文件名 //删除文件

也可以利用yum进行安装

yum list //列出相关列表

yum install 文件名 //利用服务器在线安装

15.操作学习过程中的经验

1.拷贝文件夹到另一个文件夹的时候，cp –r /var/.. /etc

2.mv 拷贝后会将原文件进行删除

3. find /etc -name "yum.repos.\*" –print 在/etc 文件夹下寻找yum.repos不限后缀的文件并打印。

find –name //带了名字的话，可以在当前目录下寻找 如果不带的话，只在本文件目录下寻找

eg:find / -name FBRs.php

进程命令：

1. ps aux

第三天 第一章UNIX基础知识

1.安装VMware tool：

cp vm…… /tmp

cd /cmp

tar –zxvf VM…

./ vm…

2.安装sublime

cd /opt

wget <http://c758482.r82.cf2.rackcdn.com/sublime_text_3_build_3083_x64.tar.bz2>

//wget 文件下载命令

tar –jxvf sublime\_text\_3\_build\_3059\_x64.tar.bz2

配置桌面快捷方式：

# cp /opt/sublime\_text\_3/sublime\_text.desktop /usr/share/applications

修改配置文件

application中设置一下

sudo ln –s /opt/sublime\_text\_3/sublime\_text /usr/bin/sublime //增加链接

将sublime打开后即可将sublime固定在启动器上

快捷键:/usr/share/applications 在这个文件夹下

3.基础知识

EOF：end of file 相当于ctrl+D

4.更改yum变成是阿里yum

学习资料：<http://www.cnblogs.com/comexchan/p/5815869.html>

云源就是rpm包，位置：/etc/yum.repos.d

Repo就是源

## 搭建LAMP服务器：

1.配置静态IP

1. vi /etc/sysconfig/network-scripts/ifcfg-ens33(ens33可以是其他的网卡设备名)

将ip与一些相关的配置修改后即可。

将DNS设置后，即可连入外网。

查看ip设置：ifconfig

参考<http://blog.csdn.net/johnnycode/article/details/50184073>

2.service network restart //重启网卡服务

ping通代表成功

更改阿里yum

yum install wget –y

wget –O （更改输入）

2.安装apache

1. yum install httpd //安装httpd
2. systemctl start httpd.service //开启httpd服务
3. systemctl enable httpd.service //设置为开机启动
4. systemctl status httpd.service //查看服务的状态

/var/www/html :这里是 CentOS 默认的“首页”目录；

service httpd status //查看服务的状态

1. 开放80端口（利用防火墙）
2. 输入http ://192.168.0.134/出现测试的apache页面

3.安装PHP

1. yum install php
2. php –version

4.上传网站

1.利用xftp将网站上传到服务器上

2.cp –r /applePay /var/www/html(apache默认的根目录) 文件夹拷贝到另一个文件夹下 需要-r

3. <http://192.168.0.134/applePay/> 可以输入这个网址检测是否成功

5.安装MySQL

1. wget <http://repo.mysql.com/mysql-community-release-el7-5.noarch.rpm>

cd 就是cd /root

1. yum install mysql-community-server
2. systemctl start mysqld
3. systemctl enable mysqld
4. systemctl status mysqld

6.安装php-mysql

1. yum install php-mysql
2. systemctl restart httpd

以上主要参考<http://techsnail.blog.51cto.com/9807313/1631842>

7.安装phpMyadmin

1. 安装EPEL源
2. wget <http://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm>
3. rpm -ivh epel-release-latest-7.noarch.rpm
4. yum repolist //检测是否已添加至源列表
5. yum install phpmyadmin php-mcrypt
6. 用文本编辑器打开phpMyAdmin的配置文件(路径：/etc/httpd/conf.d/phpMyAdmin.conf)，找出并注释掉带有"Require ip XXXX"字样的代码行。会有四处这样的代码行，用"Require all granted"取而代之。重新改动过的配置文件如下所示。
7. systemctl restart httpd //重启apache

参考<http://www.jb51.net/article/68026.htm>

1. 输入http://192.168.0.134/phpmyadmin即可检验是否，但是不可以设置为空密码，所以需要修改mysql的密码：

mysqladmin -u root -p password

修改成功后可以利用mysql –u root –p 检测是否成功

1. 之后将数据库导入即可。

按照以上步骤完成后，一个动态的网站就可以运行了！！！

## 第四天：进行unix高级环境编程的代码练习

1. 硬连接：inode，同样的
2. 软连接：只保存了一个绝对路径
3. ln mylife hardlink //硬连接
4. ln –s mylife softlink //软连接
5. ls –li //可以看出结点
6. date –s //可以更改时间
7. cp –r //可以拷贝目录
8. cp –help //可以查看帮助
9. touch //创建文件
10. mkdir //创建文件夹
11. ./a.out > temp.txt //重定向
12. ps //显示进程
13. tty //代表设备

关于父子进程

<http://blog.csdn.net/xulingxin/article/details/53439551>

这个链接解释的比较清楚

## Makefile规则

1.#代表注释

2.-lm -l //连接库的意思 这句话是连接math库的意思

3.-L //显示库的位置

默认的位置是/lib /usr/lib /usr/local/lib

4.-I //用i显示制定头文件

5.运用变量时，以${变量}使用

6.gcc –c //仅限编译，不进行连接操作

7.gcc –o //指定生成的输出文件，可执行

<http://blog.csdn.net/stpeace/article/details/49408665>

示例一:

1.gcc –c p9.c //会输出p9.o

2.gcc –o p9 p9.o //会生成p9可执行文件

示例二：

#################################################

#

# Copyright: (c) 2001-2012 Great ChinArk, Inc. All rights reserved.

#

#################################################

#

# ƽ̨������Makefile

#

# @author: sky huang

# @date: 2012-10-29

# @current version: 0.1.0

# @change list:(date|action<create,modify>|programer|version|[change log])

# 2012-10-25|create|sky huang|0.0.1|

# 2012-10-29|modify|sky huang|0.1.0|add tape controller src

#################################################

Target=drc\_server #变量

DVR\_SERVER=dvr\_server

CC=g++

CCXXFLAG=-I. -I/usr/include/drcplatform -I/usr/include -I/usr/include/mysql -I./dvr -DDRC\_DEBUG –DLINUX #头文件

LD=-lIce -lIceUtil -L/usr/lib64/mysql -lmysqlclient -lcurl -L/lib/ /lib/DRCPlatform.a #库文件目录

DVR\_LIBS=-lhcnetsdk -lpthread

SRC=DRCServer.cpp service.cpp TaskControlI.cpp DVRControlI.cpp dvr/DataTrans.cpp dvr/DvrClient.cpp \

lib/ServerUtil.cpp Archive/ArchiveController.cpp Archive/DVRAlarmArchiveController.cpp Archive/DVRFullArchiveController.cpp \

Archive/DVRPackageArchiveController.cpp Archive/RecyclebinArchiveController.cpp dvr/AXIS\_CurlUTIL.cpp dvr/AXIS\_Task.cpp \

tape/TapeController.cpp tape/TapeTaskDispatcher.cpp tape/TapeControlI.cpp fs/FileTapeBKRSController.cpp \

log/JobLogUpdater.cpp log/SystemLogUpdater.cpp ClientCleaner.cpp

INC=service.h TaskControlI.h DVRControlI.h include/public.h dvr/DvrClient.h dvr/DataTrans.h \

lib/ServerUtil.h Archive/ArchiveController.h Archive/DVRAlarmArchiveController.h Archive/DVRFullArchiveController.h \

Archive/DVRPackageArchiveController.h Archive/RecyclebinArchiveController.h dvr/AXIS\_CurlUTIL.h \

dvr/AXIS\_Task.h fs/FileTapeBKRSController.h log/JobLogUpdater.h log/SystemLogUpdater.h ClientCleaner.h

OBJ=DRCServer.o service.o TaskControlI.o DVRControlI.o DvrClient.o DataTrans.o ServerUtil.o ArchiveController.o \

DVRAlarmArchiveController.o DVRFullArchiveController.o DVRPackageArchiveController.o RecyclebinArchiveController.o \

AXIS\_CurlUTIL.o AXIS\_Task.o TapeController.o TapeTaskDispatcher.o TapeControlI.o FileTapeBKRSController.o \

JobLogUpdater.o SystemLogUpdater.o ClientCleaner.o

# dvr src

DVR\_SRC=dvr/DeviceCFG.cpp dvr/DeviceOperate.cpp dvr/PlatformLink.cpp dvr/TaskOperate.cpp dvr/DvrServer.cpp

DVR\_INC=dvr/DeviceCFG.h dvr/DeviceOperate.h dvr/PlatformLink.h dvr/TaskOperate.h

DVR\_OBJ=DeviceCFG.o DeviceOperate.o TaskOperate.o PlatformLink.o DvrServer.o

all:$(Target) $(OBJ)

$(CC) -o $(Target) $(OBJ) $(LD) #链接目标文件生成可执行$(Target)

$(OBJ):$(SRC) #对于$(OBJ)是由此得来

$(CC) -c $(SRC) $(CCXXFLAG) #生成各自名称的目标文件

clean:

rm -rf \*.o $(Target)

home/ 是每一个用户名称的文件路径

## 配置多个IP方法

ifup {interface} //启动接口

ifdown {interface} //关闭接口

1. IP信息

1.ip a //打印ip信息 查询

vim /etc/sysconfig/network-scripts/ifcfg-eth0 //网卡信息 修改

1. 路由信息

2.route //打印路由 可以查看默认网关 不能联网需要注意 查询

/etc/sysconfig/network //修改 路由信息

1. DNS信息

3.nslookup [www.baidu.com](http://www.baidu.com) //详细解释dns 查询

/etc/resolv.conf //这个里面是最终的DNS 修改

检验方法：

修改成功后：

service network restart 或者ifup eth0

4.ping 8.8.8.8 谷歌DNS

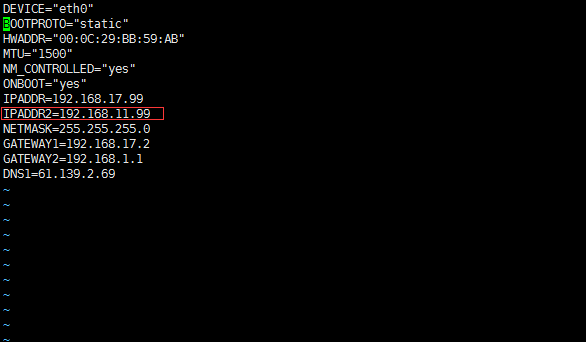
方法一：临时IP方法

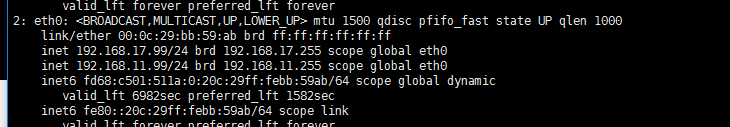
ifconfig eht0:3 192.168.17.66 netmask 255.255.255.0 broadcast 192.168.17.255 up

但是关机后，这个ip会失效

方法二：vim etc/sysconfig/network-scripts/ifcfg-eth0

直接在网卡文件中增加IPADDR2=192.168.11.99,重新启动网卡即可。





方法三：添加eth0:1

cd /etc/sysconfig/network-scripts

ll //查看当前的ip配置文件

cp ifcfg-eth0 ifcfg-eth0:1 //拷贝一个配置文件

vim ifcfg-eth0:1 //修改相关的ip

service network restart //重启网卡 或者./ifup eth0 //重启网卡

ifconfig //能打印出多个网卡信息 或者 ip a

## Windows CMD命令积累

winver //ver //可以查看系统的版本

操作系统的基本命令：

1.文件相关

cd

./--. // 都是代表当前目录的意思

../--.. //代表父目录

dir //打印当前目录

mkdir //新建文件夹

rd /s/q learn //删除learn文件夹

echo > test.txt //创建test.txt文件

del 文件名或者\*.\*删除所有文件

环境变量：

echo %PATH% 打印环境变量

echo %mysqldir% //打印名称为mysqldir的环境变量

echo %CHINARKDRC% //打印名称为CHINARKDRC这个名称的环境变量

系统信息：

systeminfo //操作系统的信息

2.网络相关

ping –n

ping –t //循环ping

Ipconfig //打印当前的ip配置

Ipconfig /all //可以查看网址是否冲突

tracert 192.168.0.1 //检查本地到达目的主机所经过的路由

route print //打印当前路由表的信息

例子：

netsh int ip add address “本地连接” 172.17.7.128 255.255.255.0

route add 192.168.17.18 mask 255.255.0.0 192.168.2.1 //增加一个路由

route -f 清除路由表

net user //可以查看有哪些用户

3.进程相关

net start watchdog 启动watchdog服务

netsh int ipv4 reset 重新启动ipv4服务

tasklist 打印进程

taskkill /pid //杀死进程

taskkill /im //杀死对应名称的进程

start /b //后台运行 这个是cmd命令 启动一个单独的程序在后台运行。

## windows使用说明

Win+x 可以查看所有的基本功能

计算机的基本功能包括：

静态：

计算机->管理 可以实现很多功能

计算机管理：计算机设备管理（驱动的开关）、磁盘管理、服务和应用、事件查看器

运行态：

网络管理、文件管理、任务管理、运行（cmd/mstsc/regedit）

环境变量的设置：

计算机->属性->高级系统设置->高级->环境变量

计算机的磁盘存储大小：

Win+I->系统->存储-> 可以查看具体的细节

管理计算机的入口：

Win+r control //控制面板

Win+x //基础设置