

### 3\_Services01 扩展应用

#### 一 构建新的教学环境

##### 1.1 构建两台虚拟机,操作系统为 CentOS7.5

[命令地址:/usr/local/bin/clone-vm7]

[真机上虚拟网卡地址:192.168.4.254]

```
[student@room9pc01 ~]$ clone-vm7
```

#克隆两台虚拟机,分别命名为 A 与 B

##### 1.2 利用 root 进入两台虚拟机,密码为:123456

##### 1.3 配置永久主机名

虚拟机 A:svr7.tedu.cn    虚拟机 B:pc207.tedu.cn

##### 1.4 配置 IP 地址

虚拟机 A 为 eth0 配置 192.168.4.7/24

虚拟机 B 为 eth0 配置 192.168.4.207/24

##### 1.5 利用真机进行远程管理两台虚拟机

```
[student@room9pc01 ~]$ ping 192.168.4.7
```

```
[student@room9pc01 ~]$ ping 192.168.4.207
```

```
[student@room9pc01 ~]$ ssh -X root@192.168.4.7
```

```
[student@room9pc01 ~]$ ssh -X root@192.168.4.207
```

##### 1.6 为真机上设置永久别名

```
[student@room9pc01 ~]$ vim /home/student/.bashrc
```

```
alias goa='ssh -X root@192.168.4.7'
```

```
alias gob='ssh -X root@192.168.4.207'
```

## 1.7 构建 Yum 仓库

### 1.7.1 服务端:真机构建了 FTP 服务器,共享光盘所有内容

真机检测软件包 vsftpd 的安装

```
[student@room9pc01 ~]$ rpm -q vsftpd
```

```
vsftpd-3.0.2-22.el7.x86_64
```

```
[student@room9pc01 ~]$ systemctl status vsftpd
```

### 1.7.2 共享光盘所有内容

默认共享的路径:/var/ftp

```
]$ mkdir /var/ftp/centos-1804/
```

```
]$ mount /iso/CentOS7-1804.iso /var/ftp/centos-1804/
```

```
]$ ls /var/ftp/centos-1804/
```

```
]$ vim /etc/fstab #设置自动挂载
```

```
/var/lib/libvirt/images/iso/CentOS7-1804.iso
```

```
/var/ftp/centos-1804 iso9660 defaults 0 0
```

```
]$ firefox ftp://192.168.4.254/centos-1804 #浏览器验证
```

客户端:两台虚拟机 yum 配置文件编写

```
[root@svr7 ~]# vim /etc/yum.repos.d/local.repo
```

```
[local_repo]
```

```
name=CentOS-$releasever - Base  
baseurl="ftp:#192.168.4.254/centos-1804"  
enabled=1  
gpgcheck=0  
[root@svr7 ~]# yum -y install httpd
```

虚拟机 C\D

- 1.配置主机名为:svr10.tedu.cn\pc20.tedu.cn
- 2.配置 eth0 的 IP 地址:192.168.4.10/24\192.168.4.20/24
- 3.配置 Yum 仓库,指向服务端为真机的 FTP 服务

## 二 扩展的几个应用

### 2.1 目录结构

认识 Linux 的目录层次:

`man hier`[显示根目录下目录说明]

常见一级目录的用途

`/boot` 存放系统引导必需的文件,包括内核、启动配置

`/bin`、`/sbin` 存放各种命令程序

`/dev` 存放硬盘、键盘、鼠标、光驱等各种设备文件

`/etc` 存放 Linux 系统及各种程序的配置文件

`/root`、`/home/用户名` 分别是管理员 `root`、普通用户的默认家目录

`/var` 存放日志文件、邮箱目录等经常变化的文件

/proc 存放内存中的映射数据,不占用磁盘

/tmp 存放系统运行过程中使用的一些临时文件

```
[root@svr7 ~]# ls -ld /tmp          #此目录自带 t 权限
```

```
drwxrwxrwt. 8 root root 172 7月 15 10:34 /tmp
```

## 2.2 权限的数值表示

权限的数值化

基本权限: $r = 4, w = 2, x = 1$

附加权限:SUID = 4,SGID = 2,Sticky Bit = 1

## 2.3 历史命令

### 2.3.1 管理/调用曾经执行过的命令

history:查看历史命令列表

history -c:清空历史命令

!n:执行命令历史中的第 n 条命令

!str:执行最近一次以 str 开头的历史命令

### 2.3.2 调整历史命令的数量

```
[root@svr7 ~]# vim /etc/profile
```

```
HISTSIZE=1000    #默认记录 1000 条
```

## 2.4 实用小命令工具

du, 统计文件的占用空间

du [选项]... [目录或文件]...

-s: 只统计每个参数所占用的总空间大小

-h: 提供易读容量单位(K、M等)

**常用格式** `du -sh [目录或文件]...`

## 2.5 date, 查看/调整系统日期时间

`date +%F`、`date +%R`

`date +"%Y-%m-%d %H:%M:%S"`

`date -s "yyyy-mm-dd HH:MM:SS"`

`~]# date +%F`      #显示 年-月-日

2019-07-15

`~]# date +%Y`      #显示 年

`~]# date +%m`      #显示 月

`~]# date +%d`      #显示 日期

`~]# date +%H`      #显示 时

`~]# date +%M`      #显示 分

`~]# date +%S`      #显示 秒

## 2.6 制作快捷方式      制作连接(链接)文件

### 2.6.1 软连接特点:

若原始文件或目录被删除,连接文件将失效

软连接可存放在不同分区/文件系统

### 2.6.2 硬连接特点:

若原始文件被删除,连接文件仍可用

硬连接与原始文件必须在同一分区/文件系统

制作软连接

**格式: ln -s /路径/源文件 /路径/快捷方式名 #必须是绝对路径**

制作硬连接

**格式: ln /路径/源文件 /路径/快捷方式名 #必须是绝对路径**

## 2.7 获取命令帮助

[root@svr7 ~]# man 5 passwd #5 代表配置文件类型帮助信息

## 2.8 zip 归档工具,跨平台的压缩格式

### 2.8.1 归档+压缩操作

**zip [-r] 备份文件.zip 被归档的文档...**

~]# yum -y install zip #需要安装

~]# zip -r /opt/file.zip /root/ /etc/passwd

~]# ls /opt

~]# zip -r /opt/nsd.zip /boot/

~]# ls /opt

### 2.8.2 释放归档+解压操作

**unzip 备份文件.zip [-d 目标文件夹]**

~]# yum -y install unzip #需要安装

~]# unzip /opt/nsd.zip -d /mnt/

```
~]# ls /mnt/
```

```
~]# ls /mnt/boot/
```

## 2.9 自定义 Yum 源

### 2.9.1 具备从互联网下载一些软件包

```
~]$ ls /linux-soft/01/
```

### 2.9.2 将 tools.tar.gz 传递到虚拟机 A

```
~]$ scp /linux-soft/01/tools.tar.gz root@192.168.4.7:/root/
```

**[scp = ssh + cp ,前提是真机能 ssh 虚拟机]**

### 2.9.3 在虚拟机 A 查看

```
]# ls /root
```

```
]# tar -xf /root/tools.tar.gz -C /
```

```
]# ls /tools/other/
```

### 2.9.4 在虚拟机 A 生成仓库数据文件

```
~]# yum -y install createrepo
```

```
~]# createrepo /tools/other/
```

```
~]# ls /tools/other/repodata      #新生成的仓库软件包文件
```

### 2.9.5 书写客户端 yum 配置文件

```
~]# vim /etc/yum.repos.d/local.repo
```

```
[myrpm]
```

```
name=other rpm
```

```
baseurl=file:///tools/other      #指定为本机为 Yum 服务端
enabled=1
gpgcheck=0
~]# yum repolist                  #检测 yum 配置文件
```

### 2.9.5 安装软件

```
[root@svr7 ~]# yum -y install cmatrix
[root@svr7 ~]# cmatrix
[root@svr7 ~]# cmatrix -C red
[root@svr7 ~]# yum -y install sl
[root@svr7 ~]# sl
[root@svr7 ~]# yum -y install oneko
[root@svr7 ~]# oneko
[root@svr7 ~]# oneko &           #将程序放入后台
```

## 三 vim 编辑技巧

### 3.1 命令模式操作

```
[root@svr7 ~]# cp /etc/passwd /opt/a.txt
[root@svr7 ~]# vim /opt/a.txt
```

### 3.2 光标跳转

Home 键 或 ^、数字 0	跳转到行首
End 键 或 "\$"键	跳转到行尾



PgUp 键、PgDn 键	向上翻页、向下翻页
1G 或 gg	跳转到文件的首行
G	跳转到文件的末尾行

### 3.3 复制/粘贴/删除

复制 yy、3yy	复制光标处的一行、3 行
粘贴 p、P	粘贴到光标处之后、之前
x 或 Delete 键	删除光标处的单个字符
dd、#dd	删除光标处的一行、#行
d^	从光标处之前删除至行首
d\$或 D(大写)	从光标处删除到行尾
C(大写)	从光标处删除到行尾,进入插入模式

### 3.4 查找/撤销/保存

/word	向后查找字符串“word”
n、N	跳至后/前一个结果
u	撤销最近的一次操作
U(大写)	撤销对当前行的所有修改
Ctrl + r	取消前一次撤销操作
ZZ(大写)	保存修改并退出

### 3.5 末行模式操作

:r /etc/filesystems                      #读入其他文件内容

```
[root@svr7 ~]# echo 123456 > /opt/2.txt
[root@svr7 ~]# echo hahaxixi > /opt/3.txt
[root@svr7 ~]# vim /opt/3.txt

:r /opt/2.txt      #读入/opt/2.txt 到当前文件

:r /etc/passwd    #读入/etc/passwd 到当前文件
```

### 3.6 字符串替换

```
:s/root/admin/      替换当前行第一个 root 为 admin
:s/root/admin/g      替换当前行所有的 root 为 admin
:1,10s/root/admin/g  替换第 1-10 行所有的 root 为 admin
:%s/root/admin/g     替换文件内所有的 root 为 admin
```

### 3.7 开关参数的控制

```
:set nu 或 nonu    显示/不显示行号
:set ai 或 noai    启用/关闭自动缩进
```

## 四 源码编译安装

RPM 包: rpm -ivh yum -y install

源码包 -> gcc make-> 可以执行的程-> 运行安装

主要优点

获得软件的最新版,及时修复 bug

软件功能可按需选择/定制,有更多软件可供选择

源码包适用各种平台

## 4.1 安装 gcc 与 make 开发工具

```
[root@svr7 ~]# yum -y install gcc
```

```
[root@svr7 ~]# yum -y install make
```

## 4.2 进行 tar 解包

```
]# tar -xf /tools/inotify-tools-3.13.tar.gz -C /
```

```
]# ls /
```

```
]# cd /inotify-tools-3.13/
```

```
]# ls
```

## 4.3 运行 configure 脚本

作用 1:检测是否安装 gcc

作用 2:可以选择软件的功能,也可以指定软件安装的路径

--prefix=指定安装的位置

```
]# cd /inotify-tools-3.13/
```

```
]# ./configure --prefix=/opt/myrpm
```

## 4.4 make 编译生成可以执行的程序

```
]# cd /inotify-tools-3.13/
```

```
]# make
```

## 4.5 make install 将可以执行的程序运行安装

```
]# cd /inotify-tools-3.13/
```

```
]# make install
```